Methods for Optimal Transmission over MIMO Wiretap Channels

Kaiya Li

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements
for the Master of Applied Science in
Electrical and Computer Engineering.

Ottawa-Carleton Institute for Electrical and Computer Engineering
Faculty of Engineering
University of Ottawa

Acknowledgments

I would like to show my appreciation to my research supervisor Dr. S. Loyka for his guidance and patient. Without his support, this thesis would not be finished. I would also like to thank my cousins Susan Shao, Arabelle Hou and Amanda Hou for their special help and advices. My deep thankfulness is shown to my family for their unconditional support and love from ten thousands kilometers away.

Abstract

We discuss the architecture of wireless multiple-input multiple-output (MIMO) channels with additive white Gaussian noise. The transmission optimization problem over covariance matrix for achieving MIMO channel capacity is reviewed under the assumption that the channel state information (CSI) is available from both the transmitter and the receiver. We also discuss several transmission strategies such as beamforming and water-filling algorithm.

The information-theoretic approach to physical-layer security for wireless communications system is discussed. Physical-layer security which is different from cryptographic approach security is achieved by exploiting the physical-layer properties of wireless channel. The principal results and concepts in this field such as equivocation rate and secrecy capacity are reviewed. The wiretap channel model is considered. We mainly discussed the Gaussian MIMO wiretap channel and the methods for obtaining its secrecy capacity.

The optimization problem for the secrecy capacity of Gaussian MIMO wiretap channel under total transmit power constraint is discussed. CVX, which is a modeling toolbox for solving convex optimization problems, is reviewed. The precision variable of CVX is also considered. The performances of CVX and Monte Carlo optimization are compared. Differential Evolution algorithm is used to compute the secrecy capacity of Gaussian MIMO wiretap channel. The case of weak eavesdropper is considered and the closed form solution based on the necessary KKT conditions is validated. The algorithm for general cases based on the linear approximation and reformulation of the original optimization problem is also investigated and validated. It has been verified that the lower bound and the upper bound of secrecy capacity can be obtained by such algorithm.

Table of Content

List of Acronym and Notations

Acronym	Meaning
AWGN	additive white Gaussian noise
CSI	channel state information
CDI	channel distribution information
KKT	Karush-Kuhn-Tucker conditions
MIMO	multiple-input multiple-output
MISO	multiple-input single-output
MMSE	minimum mean-square error
Rx	receiver
SIMO	single-input multiple-output
SISO	single-input single-output
SNR	signal to noise ratio
Tx	transmitter
WF	waterfilling

List of Symbols and Notations

Notations	Meaning
T	transpose
+	conjugate transposition
Tr(**A**)	trace of matrix **A**
a_{ij}	entry of i^{th} row, j^{th} column of matrix **A**
log|**A**|	logarithm of determinant of matrix **A**
$(x, 0)_+$	max(x,0)

Symbol	Meaning	First apperance
m	number of transmit antennas	(3.1)
n	number of receive antennas of legitimate receiver	Section 3.1
n_e	number of receive antennas of eavesdropper	Section 3.2
M	confidential message	Section 1.1
$\mathcal{M}$	confidential message set	Section 1.1
X^k	codeword	Section 1.1
Y^k	output at legitimate receiver	Section 1.1
Z^k	output at eavesdropper	Section 1.1
$\widehat{M}$	message decoded by legitimate receiver	Section 1.1
k	length of codeword	Section 1.1
R_e	equivocation rate	(1.1)
$\mathbb{H}(A\|B)$	conditional entropy	(1.1)

C_s	secrecy capacity of MIMO wiretap channel	Section 1.1
C	capacity of MIMO channel	(3.2)
$C(\mathbf{R})$	transmit rate of MIMO channel with respect to $\mathbf{R}$	(3.2)
$C_s(\mathbf{R})$	secrecy rate of MIMO wiretap channel with respect to $\mathbf{R}$	
$\mathbf{H}$=$[\mathbf{h}_{1,}\mathbf{h}_2, \dots, \mathbf{h}_m]$	MIMO channel matrix (without eavesdropper)	(3.1)
$\mathbf{x} = [x_1, x_2, \dots, x_m]^{\mathrm{T}}$	Tx vector	(3.1)
$\mathbf{y} = [y_1, y_2, \dots, y_n]^{\mathrm{T}}$	Rx vector	(3.1)
$\mathbf{R} = \overline{\mathbf{x}\mathbf{x}^{\mathrm{T}}}$	transmit covariance matrix	Section 1.1
$\mathbf{R}^*$	optimal transmit covariance matrix	Section 1.1
P_T	Total transmit power	(3.1)
$\mathbf{U}$	unitary matrix if the eigenvectors representing the data streams	(3.3)
diag$\{P_1, P_2, \cdots, P_m\}$	diagonal matrix whose entries represent the power allocated to different streams	(3.3)
$\mathbf{n} = \mathcal{CN}(0, \sigma_0^2\mathbf{I})$	circularly symmetric additive white Gaussian noise vector with i.i.d. entries	(3.1)
σ_0^2	noise power	Section 3.1
$\lambda_i(\mathbf{A})$	i^{th} eigenvalue of $\mathbf{A}$	(3.4)
$\mathbf{W}_1 = \mathbf{H}_1^{+}\mathbf{H}_1$	legitimate receiver channel matrix	(3.9)
$\mathbf{W}_2 = \mathbf{H}_2^{+}\mathbf{H}_2$	eavesdropper channel matrix	(3.9)
$\mathbf{I}_{m\times m}$	$m \times m$ identical matrix	Case 4-1
$\mathbf{E}_{m\times m}$	$m \times m$ all ones matrix	Case 5-1
SD	Standard diviation	(5.3)
$r_+(\mathbf{A})$	number of positive eigenvalues of $\mathbf{A}$	Section 5.1
$\mathbf{D}_{i,G}$	i^{th} target matrix in G^{th} generation	(5.14)
$\mathbf{V}_{i,G}$	i^{th} mutant matrix in G^{th} generation	(5.15)
F	scale factor of differential variation	(5.15)
$\mathbf{U}_{i,G}$	i^{th} trial matrix in G^{th} generation	(5.16)
$\mathbf{W}_{+(-)}$	positive (negative and zero) eigenmodes of $\mathbf{W}_1 - \mathbf{W}_2$	(5.19)
$\boldsymbol{v}_i(\mathbf{A})$	i^{th} eigenvector of $\mathbf{A}$	Section 6.1

P_{active}	active power contributing to the secrecy capacity	(6.7)
$\Delta\mathbf{R}$	step size of the reformulated optimization problem	(7.2)
t	step size of backtracking line search	(7.12)
$\mathbf{K}$	auxiliary matrix	(7.17)
C_s^-	lower bound of secrecy capacity	(7.18)
C_s^+	upper bound of secrecy capacity	(7.19)

1.Introduction

1.1 Motivation

Multi-input Multi-output (MIMO) system is a core architecture utilized in modern wireless communication standards such as IEEE 802.11n (Wi-Fi), Long Term Evolution (LTE) and 4G (LTE Advanced). MIMO system is one of the main LTE technologies. By using MIMO, rather than providing interference in previous telecommunications systems, multiple signals paths can be used to increase throughput. In 1995-1996, it was first proposed by Foschini [12] and Telatar [4] to improve the channel throughput effectively. The introduction of MIMO architecture has brought significant progress in the field of wireless communication systems since it improves the spectral efficiency significantly when compared to conventional systems [4]. MIMO architecture offers spectrum effectiveness in communications by utilizing the degrees of spatial freedom supplied by multiple transmit and receive antennas, such that the transmission rate and quality of communications can be improved. It has attracted worldwide attention in recent years since it improves systems throughput significantly without transmit power or bandwidth increase.

The information-theoretic physical-layer security in wireless communications has been exploited to face new challenges on conventional security measures such as cryptography and improve the overall security for wireless communications [14]. Unlike the conventional security techniques, the physical-layer security utilizes the physical-layer characteristics of wireless channel such as fading or noise for concealing legitimate communications without using the encryption key. Such characteristics provide structural randomness to prevent third parties from intercepting. The legitimate receiver can also benefit by exploiting the difference between the channels to legitimate receiver and eavesdropper [14].

The wiretap channel is a basic model representing the physical-layer security for communications [16]. As shown in Figure 1.1, a transmitter tries to transmit the confidential message M (assumed to be randomly and uniformly distributed over a message set $\mathcal{M}$) to a legitimate receiver, meanwhile preventing the message to be obtained by the eavesdropper by stochastically encoding M into a codeword X^k consisting of k symbols. Y^k and Z^k are output sequences for the legitimate receiver and the eavesdropper respectively, the legitimate receiver obtains estimated message $\widehat{M}$ by decoding Y^k.

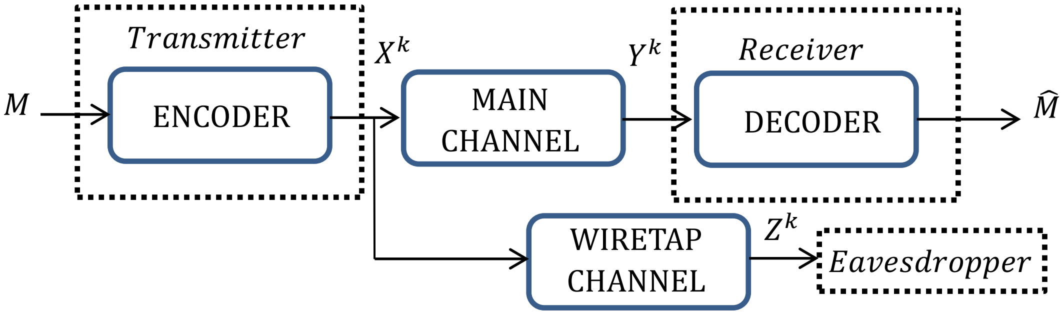


Figure 1.1. Wiretap channel model.

Figure 1.2 shows the wiretap channel model with physically degraded assumption, where $\widehat{M}$ can be obtained by the receiver by decoding Y^k whereas Z^k is a noisy version of Y^k observed by the eavesdropper [14].

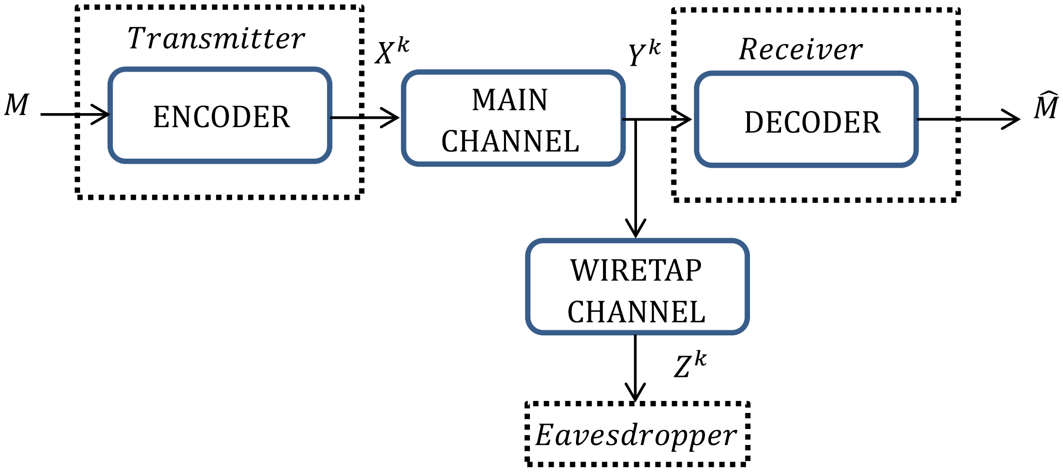


Figure 1.2. Wiretap channel model (degraded assumption).

The equivocation rate [14] is an important concept which quantifies how unlikely the eavesdropper to intercept valuable information in information-theoretic physical-layer security. This concept is defined as

$$R_e = \frac{1}{k}\mathbb{H}(M|Z^k) \tag{1.1}$$

where $\mathbb{H}(A|B)$ denotes the conditional entropy of random variable A given random variable B; R_e represents the normalized uncertainty of message M for a given Z^k. The secrecy capacity C_s is the maximum transmission rate achievable when the equivocation rate equals to the transmission rate as k goes to infinity [14].

We focus on the secrecy issue for MIMO system. A Gaussian MIMO wiretap channel is considered where the transmitter, the legitimate receiver and the eavesdropper have multiple antennas; the signals received by the legitimate receiver and the eavesdropper are corrupted by additive white Gaussian noise (AWGN). The secrecy capacity of the Gaussian wiretap channel is lower-bounded by the difference between the capacity of the channel to the legitimate receiver and the capacity of the channel to the eavesdropper [14]. The secrecy capacity for Gaussian MIMO wiretap channel is formulated as an optimization problem over transmit covariance matrix $\mathbf{R}$ under the total transmit power constraint [23]-[28]. Unfortunately, explicit solutions are only available for certain cases such as weak eavesdropper or full-rank covariance matrix [28, 30]. The analytical solution for general Gaussian MIMO wiretap channel is still an open problem.

This thesis discusses the application of CVX for solving the optimization problem for secrecy capacity of Gaussian MIMO wiretap channel. CVX is a popular modeling system for solving convex optimization problems and introduced in [63]. Since it is found that CVX is not able to solve the optimization problem for secrecy capacity properly, the stochastic optimization methods, such as the Monte Carlo optimization [64] and the Differential Evolution algorithm [67] are investigated for obtaining the numerical solution of the problem. An analytical solution for the weak eavesdropper case in [32] is validated, and a method for obtaining the optimal covariance matrix $\mathbf{R}^*$ for a general Gaussian MIMO wiretap channel is proposed and investigated.

1.2 Contribution of Thesis

The major contributions of the thesis are:

- Investigating of the CVX precision variable's impact on the optimization problem for MIMO channel capacity;
- Developing random optimization methods for obtaining the secrecy capacity of the Gaussian MIMO wiretap channel;
- Discussing the limitations of CVX for dealing with the optimization problem of secrecy capacity of Gaussian MIMO wiretap channel over input covariance matrix;
- Validating the analytical solution introduced in [30] for the weak eavesdropper case;
- Accessing and validating the reformulation and approximation of the original optimization problem for secrecy capacity of general Gaussian MIMO wiretap channel.

1.3 Organization of Thesis

This thesis is organized as follows. Chapter 2 reviews recent results on MIMO systems and MIMO channel capacity, security issues of communications, wiretap channel and recent explicit solutions for the optimization over Gaussian MIMO wiretap channels; convex optimization is reviewed as well. Stochastic optimization methods such as Monte Carlo optimization and Differential Evolution algorithm are discussed. Chapter 3 presents the model of Gaussian MIMO wiretap channel. Chapter 4 discusses the CVX, which is main modeling toolbox for handling convex optimization problems throughout this thesis. The precision variable of CVX is considered in this chapter. Chapter 5 gives the numerical results obtained by Monte Carlo optimization and Differential Evolution algorithm. It also gives the performances of these two algorithms including efficiency and accuracy. Chapter 6 discusses the difficulty of CVX for solving optimization problem for secrecy capacity of Gaussian MIMO wiretap channel over covariance matrix. It validates the analytical solution for the case of weak eavesdropper given in [30]. It discusses another modeling toolbox, YALMIP. It proves that both CVX and YALMIP are not able to solve our optimization problem correctly in general cases. Chapter 7 analyses the reformation and approximation of the optimization problem. Backtracking is discussed for establishing the connection between the original problem and the approximated problem. Chapter 8 concludes this thesis.

2. Literature Review

2.1 MIMO System and MIMO Channel Capacity

To solve issues presented by the rapid development of high-speed wireless communication systems, multiple transmit and multiple receive antenna (MIMO) system is employed in space-time coding and signal processing techniques which improve the quality and spectral efficiency of wireless communications. Such a MIMO system provides a powerful paradigm in wireless communications. An overview of this MIMO communication systems is discussed in [1], [4] and [12] in terms of MIMO channel capacity based on information theoretic results. It has been observed that the channel capacity in rich scattering environments can be improved by employing the MIMO systems.

The relationship of MIMO system and practical wireless communications standards is addressed in [3]. It emphasizes some techniques and algorithms such as spatial multiplexing and space-time coding schemes for realizing the benefits of MIMO systems

The capacity formula of single user MIMO channel with and without fading is derived in [4] and later proves that the potential gains of such a MIMO system is much greater than SISO system when the noise and fades are assumed to be independent at different receiving antennas. A derivation of the capacity is given by maximizing the mutual information between input and output of the channel. The ergodic capacity of a Gaussian channel with Rayleigh fading is introduced. Each entry of this channel matrix has uniformly distributed phase and Rayleigh distributed magnitude, the capacity of such channel is achieved when the input signal is a circularly symmetric complex Gaussian variable.

An iterative waterfilling (WF) numerical algorithm is presented in [5] to compute the optimal distribution that maximizes the Gaussian MIMO channel capacity. It corresponds to finding an optimal transmitter covariance matrix. The capacity computation is formulated as a convex optimization problem. It showed that for a single user Gaussian MIMO channel, the solution can be easily obtained by the WF algorithm. The channel can be decomposed into a set of parallel independent scalar sub-channels, and the power gains of sub-channels correspond to the eigenvalues of the product of channel matrix and its conjugate transpose ($\mathbf{HH}^{+}$) [2]. Therefore the optimal power allocation strategy is a WF power allocation based on the SNR in each sub-channel. The optimal power allocation for a fading channel over time can also be computed efficiently by such an algorithm.

The pioneering work [4] has proved that MIMO could provide a large increase channel capacity compared to traditional SISO systems under the assumption that fades are independent and identical distributed (i.i.d) in each sub-channel (a link between each transmitter and each receiver). Under such assumption, the channel capacity of MIMO system is the sum of the capacities of the sub-channels. Reference [8] investigates the effects of fading correlations, which is typical for real propagation environments. The distributions of the channel gains of sub-channels can be affected by fading correlation effect such that the capacity is lower than what have been achieved under i.i.d fading assumption.

Since it is difficult to evaluate the exact ergodic capacity of MIMO correlated fading channel, reference [9] focuses on evaluating the bounding techniques of MIMO capacity. The upper and lower bounds on the ergodic capacity of spatially correlated Rician MIMO channels are considered in [10], the outage capacity of such channels at high signal-to-noise ratio (SNR) is also discussed. It has been found that the upper bounds of ergodic capacity are tight at high SNR. Both ergodic and outage capacities can be affected by the antenna configuration. For the single-user system, the predicted capacity gain obtained from MIMO is based on sometimes unpractical assumptions such as the channel state information (CSI) is both known at transmitter and receiver. Reference [11] shows that the capacity gain is significantly affected by the available channel information at either transmitter or receiver, SNR and the correlation between the sub-channels. Unlike single-user channels, the capacity of multiuser MIMO system is difficult to obtain. For single-user systems, the capacity is found to grow linearly with the number of antennas under the assumption of perfect CSI at both transmitter and receiver. Without such an assumption, the capacity computation is more difficult and depends on the CSI of channel distribution information (CDI) and value of SNR.

Reference [12] addresses the single user MIMO architecture where the transmitter and receiver employ the same number of antennas. The channel is assumed to be Rayleigh fading and CSI is known by the transmitter but unknown by the receiver. It proves that the capacity gain is significant and linear with the number of antennas, which is inline with the observation in [11]. The MIMO systems have been attracted enormous attentions in the world. Figure 1.1 shows the number of publications in this area since it was first discovered.

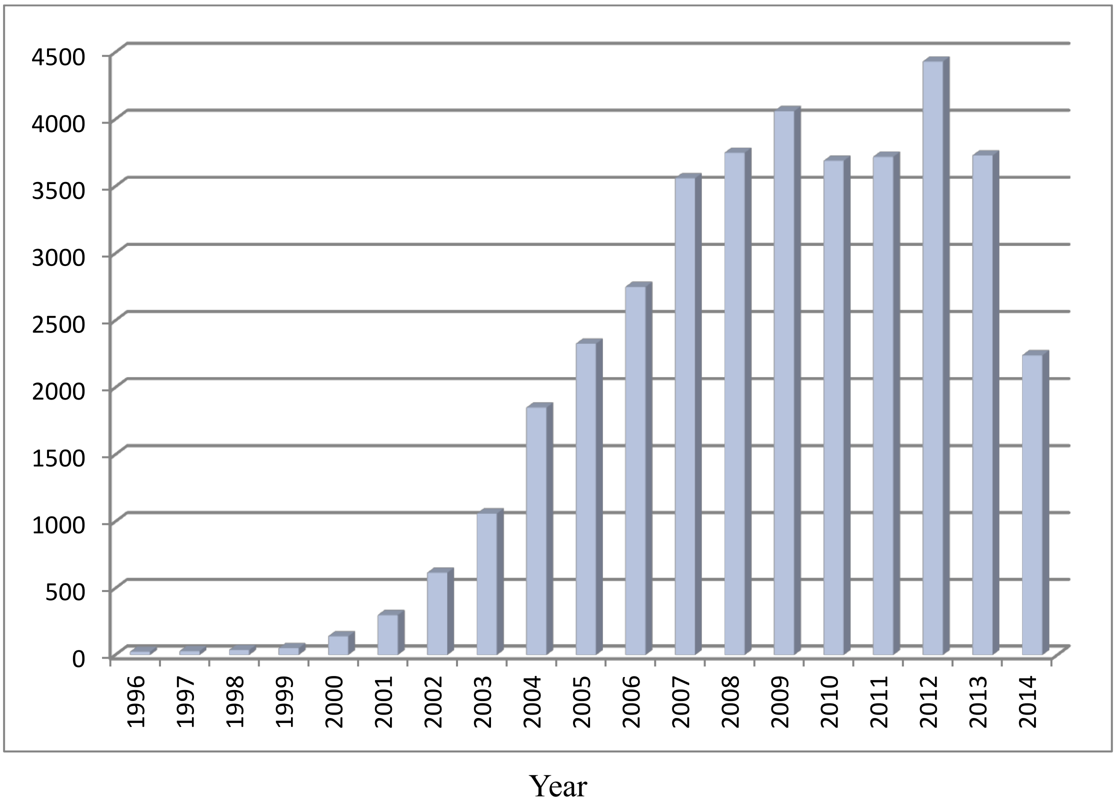


Figure 1.1. Number of publications of MIMO since 1996[1].

[1] This data includes any published paper containing all of the keywords “MIMO”, “wireless”, “channel”, “space-time”, “communications” returned by the Google Scholar search engine.

2.2 Information Theoretic Security

Security, including confidentiality, integrity, authentication and nonrepudiation, is becoming an extremely important issue in the communication systems [14]. The confidentially, to guarantee that the legitimate receiver is able to obtain the intended information while preventing the eavesdropper accessing that information, is achieved via cryptographic encryption. The original source information is encrypted and converted by a key, including secret-key encryption algorithm and public-key encryption algorithm, from plaintext to ciphertext. The eavesdropper is able to access the ciphertext while being unable to get the decryption key to recover the original information. Using cryptography to provide the security over wireless communication networks meets several significant challenges because of the characteristics of the wireless network. Therefore, a new research direction based on information theory has been proposed to solve the security issues in wireless network [20].

To enhance the security without sacrificing efficiency, the hybrid cryptosystems are employed in practice [15, 16]. The security based on information theoretic analysis was introduced by Shannon in [19]. An important concept of information theoretic security system is equivocation which is the measure of security level and studied in [17], [18] and [20]. It is developed without applying encryption keys while utilizing the difference between the legitimate channel and the eavesdropper's channel based on the randomness of the physical medium to guarantee the legitimate receiver to obtain the information and prevent the eavesdropper from accessing the useful information [14]. It can be thought as the entropy of the confidential information conditioned on channel outputs for the eavesdropper. There are two performance need to be considered which are *reliability*[2] and *security*[3]. A confidential message M (assumed to be randomly and uniformly distributed over a message set $\mathcal{M}$) is stochastically encoded (mapped) by the encoder into the codeword X^k consisting of

[2] Reliability is measured in term of probability of error, reliability condition for a length k message: $\lim_{k\to\infty} \Pr\left(\widehat{M} \neq M\right) = 1/|\mathcal{M}| \Pr \sum_{|\mathcal{M}|}\{\widehat{M} \neq M\} = 0$ where $\widehat{M}$ is the estimation of message M at receiver side. [3] Strong security condition is measured in terms of the normalized mutual information between the message M and Eve's observation Z^k: $\lim_{k\to\infty} I(M; Z^k)/k = H(M) - H(M|Z^k) = 0$, i.e. the leakage of information to the eavesdropper is zero when message k trends to infinity.

k symbols. The transmitter transmits X^k to a legitimate receiver, meanwhile prevents the message to be obtained by the eavesdropper. Y^k and Z^k are output sequences for the legitimate receiver and the eavesdropper respectively, the legitimate receiver obtains estimated message $\widehat{M}$ by decoding Y^k. The reliability is measured in term of the probability of error and the security is measured by the equivocation rate of the eavesdropper which is the normalized mutual information between the message and the eavesdropper's observation. The equivocation rate is used for measuring the secrecy level of the confidential message and defined as $R_e = \frac{1}{k}\mathbb{H}(M|Z^k)$. It estimates the uncertainty of eavesdropper about message M based on the observation on Z^k. The secrecy capacity C_s is the maximum transmission rate when both reliability and security are satisfied. The wiretap channel is the channel model associated with the information theoretic secrecy which was introduced in [17] and [20]. It can be expanded to several different types such as Gaussian, multi-input multi-output (MIMO), compound, and feedback wiretap channels [14].

Several code schemes for achieving the secrecy capacity of wiretap channels are introduced in [21] to [23]. Reference [21] introduced a coding scheme for the Gaussian wiretap channel based on low-density parity-check (LDPC) codes. This coding scheme is effective when it yields the bit-error-rate (BER) of an eavesdropper whose SNR is lower than a threshold SNR$_E$ that is close to 0.5. Meanwhile, it provides high reliability for the communications between the transmitter and the legitimate receiver when they have SNR above another threshold SNR$_B$. This paper designed the optimal LDPC code with differential evolution to realize the reliability of the communications in the main channel and try to keep SNR$_B$ /SNR$_E$ as low as possible. The LDCP codes can be combined with conventional cryptographic to improve security and take advantages of the physical-layer characteristic of communication channels such as its stochastic nature. Polar code was first introduced by Arikan and considered in [22] for constructing a coding scheme that achieves the secrecy capacity of the wiretap channel. It was shown that the coding scheme works for any wiretap channels when both legitimate channel $\mathbf{W}_1$ and eavesdropper $\mathbf{W}_2$

are symmetric, binary-input and $\mathbf{W}_1 \geq \mathbf{W}_2$ (degraded assumption). A modified construction for strong security is provided with the guarantee that the rate approaches to the secrecy capacity. While the reliability of the communications may not be satisfied under this coding scheme unless $\mathbf{W}_1$ is noiseless. Reference [23] introduced an extension of Bellare-Tessaro coding scheme for discrete, degraded, symmetric wiretap channel to Gaussian wiretap channel. It simplified constructing the code for achieving the secrecy capacity for a Gaussian wiretap channel to constructing the code scheme for achieving capacity of Gaussian channel. The proposed coding scheme consists of two layers; one is for reliability of communications and another is for security.

The Gaussian wiretap channel is introduced in [24] and the latter provides an important conclusion that the secrecy capacity C_s of such a channel is the difference between the capacities of the main channel and the eavesdropper's channel. Based on the discussions in [20], [25] and [26], the Gaussian signaling is proved to be the optimal over Gaussian wiretap channel. The pioneering studies from [24] to [31] investigate the physical-layer security of MIMO channel. Reference [27] proved that the proper utilization of space-time diversity by employing MIMO could improve the information security. The secrecy capacity of a MIMO wiretap channel computation is based on the optimization of the transmit covariance matrix with the assumption that the main channel information and the eavesdropper's channel information are both available to the transmitter. It has been proved that the secrecy capacity here is the difference between the maximum mutual information of the legitimate receiver and the eavesdropper and that it is not affected by the number of antennas. The optimal covariance matrices for special cases have been obtained in [26], [28] and [29]. It is shown in [28] that the secrecy capacity optimization problem is not always convex for general Gaussian MIMO channel. For the case where only transmitter employs multiple antennas and legitimate receiver and eavesdropper employ single antenna (MISO), the analytical solution of the optimization problem can be obtained. Similarly, the secrecy capacity of SIMO channel under Gaussian noise is investigated in [29] by transforming the SIMO channel into a scalar Gaussian wiretap channel based on communication theory. It is proved that the impact of the slow fading on the secrecy capacity can be mitigated by applying multiple receive antennas. In addition, the optimal covariance matrix under the total power constraint of a 2-2-1 wiretap Gaussian MIMO channel is obtained in [26]. In such a channel model, the transmitter and legitimate receiver employ two antennas while the eavesdropper has only one antenna. The beamforming is proved to be optimal in this case. The secrecy capacity and optimal signaling under the total transmit power constrain of general Gaussian MIMO wiretap channel, in which the transmitter, legitimate receiver and eavesdropper are equipped with multiple antennas, is investigated in [30, 31]. Since the optimization of this objective is not always convex, analytical solutions are not available except for some special cases. But it has been proved that the necessary condition of the optimality, which is transmitting the information into the positive direction of the difference channel between the legitimate receiver and the eavesdropper, can be obtained from the KKT conditions. The analytical solution has been obtained when the optimal covariance matrix has full-rank [30]. For some cases, the optimal signalling is similar to the conventional WF. However, in high SNR regime, the isotropic signaling may not be optimal. For the general cases, it has been shown that the rank of optimal covariance matrices does not exceed the rank of the difference channel. For example, if the difference channel has only one positive eigenmode, then the optimal covariance matrix has rank-one as well and the closed form solution of latter can be achieved. Reference [32] provided the closed form solution of optimal covariance matrix for the case where the eavesdropper is weak. In addition, [33] discussed the secrecy capacity of compound MIMO Gaussian channel where the channel information of main channel is known by the transmitter but only partial eavesdropper’s CSI is known by the receiver. Reference [34] investigated the high SNR regime and revealed that radiating power isotropically in all directions could achieve the near-optimal performance in such a regime. Reference [35] provided the upper bound and the lower bound of secrecy capacity of general MIMO Gaussian channel. An isotropic eavesdropper model is studied by [36] which provides the upper and lower bound of secrecy capacity for the non-isotropic eavesdropper case. It is also proved to be the worst case.

Furthermore, the security of communications over fading channels is discussed from [37] to [41]. The secrecy capacity of fading channels is studied in [37] under the assumption of asymptotically long coherent intervals. It has been proved that the secrecy capacity attained under the full CSI condition where the transmitter knows both main channel and eavesdropper's channel is the upper bound of the secrecy capacity when the transmitter only knows the eavesdropper's channel. Reference [37] also revealed the constructive impact of fading on the secrecy capacity. The fading wiretap broadcast channel with confidential messages (BCC) is studied in [38] where the full CSI is available for both the transmitter and the receiver. The boundary of secrecy capacity region of the parallel BCC is achieved by optimizing the source power allocation. The results can be applied to study the fading BCC. Reference [39] investigated the case where the legitimate receiver's channel is an AWGM channel while the eavesdropper's is Rayleigh fading with additive Gaussian noise. It shows that even when the eavesdropper's channel is arbitrarily stronger than the main channel, the secrecy capacity can also be achieved by injecting artificial noise. The further discussion of artificial noise for improving the secrecy transmission is given in [40]. It shows that the secrecy capacity can be achieved by adding artificial noise on the information signal. In addition, the artificial noise can be generated intelligently in MIMO scenario such that the noise only degrades the eavesdropper's channel but does not degrade the intended receiver's channel. The noise can be transmitted in the null space of the legitimate channel so that it does not affect the legitimate communication. Reference [41] considered the system model which has multiple colluding eavesdroppers that can be modeled as an eavesdropper equipped with multiple antennas. Meanwhile, the transmitter and legitimate receiver both employ multiple antennas. Following the observation in [40], the secrecy transmission can be achieved by selectively degrading the eavesdropper's channel by adding artificial noise even if the main channel is weaker than the eavesdropper's channel. The degrading of eavesdropper's channel can be made by transmitting the information signal and the artificial noise separately on the positive direction and null space of the legitimate channel respectively.

2.3 Convex Optimization

The optimization of transmitter in this thesis is based on convex optimization theory, which is introduced in [42] to [44]. It shows that the numerical results of a special class of mathematical optimization problems such as linear and least-squares problems can be solved efficiently since a reasonably complete theory for this class of problems has been found. In the last decades, new methods for solving new classes of optimization problems including semidefinite problems were developed. Within a few years, numerous applications of convex optimization have been discovered. Formulating a problem as a convex optimization problem brings a lot of advantages. It allows us to efficiently and reliably solve the original problem. Reference [45] studied the robust convex optimization and [46] introduced four kinds of convexity which are weaker than strict convexity but stronger than quasiconvexity. Reference [47] discussed quasiconvex programmings. References [48, 49] investigated the central cutting plane algorithm for the convex problems. This algorithm approaches the optimum by building up a cutting plane through the center of a polyhedral approximation to the optimum to generate a series of points satisfying the KKT conditions of the problems.

References [50, 51] discussed the interior-point method for solving semidefinite problems and also provided the convergence rate, stability and numerical results. Reference [52] studied robust convex problems with quadratically constraints which can be thought as a subset of robust convex programs studied in [43]. It investigated an uncertainty that allows this kind of problems to be formulated as second-order cone programs (SOCP) which is studied in [53]. It described a family of problems that can be reformulated as SOCP and the applications of SOCP in engineering industry. Some other kinds of convex optimization problems such as least-square, conic and linear problems are introduced in [54] to [57].

Convex optimization is widely used in communication systems. Reference [58] introduced the applications of convex optimization in signal processing and digital communication. It focuses on how to exploit the hidden convexity. Reference [59] shows that the antenna array pattern synthesis problems can be solved efficiently when expressed as convex optimization problems. References [60, 61] investigated the problems of achieving capacity of different types of MIMO channels via convex optimization.

2.4 Stochastic Optimization Methods

Reference [78] discusses the stochastic methods for solving optimization problems that have randomness such as random search, stochastic approximation and evolutionary computation. It focuses on two general problems which are finding the vector-valued variable $\boldsymbol{\theta} \in \Theta$ (Θ is the domain of allowed values of $\boldsymbol{\theta}$) that minimizes a scalar-valued objective function $L(\boldsymbol{\theta})$ and finding the vector-valued variable $\boldsymbol{\theta} \in \Theta$ that realizes the equation $\boldsymbol{g}(\boldsymbol{\theta}) = \mathbf{0}$ for some vector-valued function $\boldsymbol{g}(\boldsymbol{\theta})$ where $\boldsymbol{g}(\boldsymbol{\theta}) = \partial L(\boldsymbol{\theta}) / \partial \boldsymbol{\theta}$**.** These two kinds of problems are closely related and can be converted into each other. The stochastic methods introduced are for solving problems where the closed-form analytical solution of $\arg\min_{\boldsymbol{\theta}} L(\boldsymbol{\theta})$ is hard to obtain. In some cases, only local optimum can be guaranteed to be obtained by some algorithms while the global solution can be found among multiple local solutions. The algorithm discussed in [78] is called stochastic search and optimization since there is a random choice made as the algorithm searches toward the solution. The (pseudo) random number generators in MATLAB are reviewed such as '*randn*' and '*rand'* which are popular in stochastic search and optimization. The efficiency of algorithms is a key point which is needed to be considered including computer run time, number of algorithm iterations, and the number of objective function evaluations. The efficiency of algorithms can be affected by the dimensionality of objective functions and there is a trade-off between the algorithm efficiency and the algorithm robustness. The direct search methods for solving the optimization problem of minimizing objective function $L = L(\boldsymbol{\theta})$ subject to $\boldsymbol{\theta} \in \Theta$ are introduced. Such methods are effective when the gradient of L does not exist. A series of values of $\boldsymbol{\theta}$ in Θ are generated randomly such that the values of $L(\boldsymbol{\theta})$ can be tested sufficiently. This process will be repeated until an appropriate stopping criterion is satisfied. The values of $\boldsymbol{\theta}$ can be generated with prior information. The optimum of L can be obtained only if the probability of $\boldsymbol{\theta} = \boldsymbol{\theta}^*$(the optimal $\boldsymbol{\theta}$) is nonzero.

Monte Carlo (MC) optimization is a popular stochastic simulation method introduced in [64]. It is an efficient and general technique for solving mathematical programming problems when the analytical solution is difficult to obtain, especially for nonlinear systems. As shown by Conley [65, 66], MC approaches the optimum by generating a sample of a large number of feasible solutions and selecting the best one. Furthermore, the multistage Monte Carlo optimization technique is also discussed in [64] which is an improvement on the initial MC. It focuses on searching for the optimum in a narrow region which is based on incumbent optimum until a new incumbent optimum is found. This process is repeated a number of times until no better solution is found. Our rank-adaptive Monte Carlo is based on this method and will be discussed in Chapter 5.

Differential Evolution Algorithm (DE), introduced in [67], is a powerful technique for the optimization of possibly nonlinear and non-differential functions over continuous spaces, (note that [67] is popular and has been cited 7987 times). It has been demonstrated that DE is more efficient than many other popular global optimization algorithms. Reference [68] compared the performance of DE with other recently investigated methods by testing the convergence speeds of the optimizations of 15 carefully-selected functions. It found that DE was the fastest for 11 of them and competitive on the remaining 4 functions. References [69] and [70] investigated the choice of DE’s control parameters including dimensionality, population size, scale factor of the differential variation and crossover constant. Reference [71] focuses on the effects of crossover and the scaling parameter of differential vector in mutation. In general cases, the Differential Evolution algorithm is population-based. Reference [70] studied opposition-based DE to improve efficiency. Several studies from [72] to [76] introduced the self-adaptive Differential Evolution Algorithm. Since it has been demonstrated that the success of DE in optimizing an objective depends on proper choosing trial vector generation strategy and associated parameters, the adaptive DE is developed. Differing from the conventional DE, the control parameters in adaptive DE are not pre-specified but self-adapting according to the previous experiences during the evolutionary process. Therefore, more appropriate parameters setting can be determined adaptively for different generations. It has been concluded that self-adaptive DE is more effective and able to obtain better quality results. Reference [77] discussed DE for multi-objective optimization.

2.5 Summary

With investigation and utilization of MIMO which is one of the most significant advances in modern communications, the security issue of MIMO system is becoming more and more important. This chapter reviews the recent advances of this field and discusses associated methods. The main issues are the optimization of covariance matrix and the computation of secrecy capacity of wiretap MIMO channel. Several papers from [24] to [31] have provided the analytical solutions of the covariance matrices for some special wiretap channels, i.e. the solutions are limited, while the solutions for general cases are still open problems. This thesis considers the methods of obtaining the optimal covariance matrix of general Gaussian MIMO wiretap channel by utilizing convex optimization [42], Differential Evolution algorithm [67] and Monte Carlo optimization [64].

3. System Model

3.1 Capacity of Regular MIMO Channel

The standard discrete-time MIMO system model is given by

$$\mathbf{y} = \mathbf{Hx} + \mathbf{n} = \sum_{i=1}^{m} \mathbf{h}_i x_i + \mathbf{n} \tag{3.1}$$

where $\mathbf{H}$=$\left[\mathbf{h}_{1,}\mathbf{h}_2, \dots, \mathbf{h}_m\right]$ is the $n \times m$ matrix consisting of channel gains between each transmit (Tx) antenna and each receive (Rx) antenna; $\mathbf{h}_i$ denotes the i^{th} column of $\mathbf{H}$; n and m are the number of Rx and Tx antennas; $\mathbf{y} = [y_1, y_2, \dots, y_n]^{\mathrm{T}}$ and $\mathbf{x} = [x_1, x_2, \dots, x_m]^{\mathrm{T}}$ are the vectors representing the received symbols and transmitted symbols; $\mathbf{n}$ is the vector of circularly-symmetric additive white Gaussian noise (AWGN) assumed to be $\mathcal{CN}(0, \sigma_0^2\mathbf{I})$, i.e. independent and identically distributed in each receiver. The channel matrix $\mathbf{H}$ is assumed to be known by both the transmitter and receiver. Figure 3.1 depicts this system model.

The channel capacity of such channel is given by [1]:

$$C = \max_{\mathbf{R}} C(\mathbf{R}) = \max_{\mathbf{R}} \left\{ \ln \det\left(\mathbf{I} + \frac{1}{\sigma_0^2} \mathbf{WR} \right) \right\} ; \text{s.t.} \geq \mathbf{0}; \mathrm{Tr}(\mathbf{R}) \leq P_T \tag{3.2}$$

where $\mathbf{W} = \mathbf{HH}^{+}$; $\mathbf{R} = \overline{\mathbf{xx}^{\mathrm{T}}}$ is the input covariance matrix; $C(\mathbf{R})$ stands for the transmit rate for the given channel which is a function of $\mathbf{R}$, i.e. C is the maximum rate of information that can be transmitted reliably; Tr(·) denotes the trace operator; P_T denotes the total transmit power; $\mathbf{R} \geq \mathbf{0}$ means that $\mathbf{R}$ is a positive semi-definite matrix and has the form [1]

$$\mathbf{R} = \mathbf{U}\, \mathrm{diag}\{P_1, P_2, \cdots, P_m\}\, \mathbf{U}^{+} \tag{3.3}$$

where $\mathbf{U}$ is an unitary matrix of eigenvectors representing the data streams; $P_i (i = 1,2, \cdots, m)$ stands for the power allocated to the i^{th} eigen direction (stream). For convenience, throughout this thesis the noise power is unity, i.e. $\sigma_0^2 = 1$, so that the value of P_T is equivalent to the signal-to-noise ratio (SNR).

MIMO systems have been utilized for improving the capacity and reliability of wireless communications for a long time [4]. The transmit antenna array can be used to implement two important techniques: space-time coding and beamforming [6]. Space-time coding is used for providing diversity in the fading channel while beamforming is optimal when the information is intended for a single receiver. According to [7], beamforming is defined as the strategy where information is transmitted in a single direction so the optimal covariance matrix $\mathbf{R}$ has rank-one.

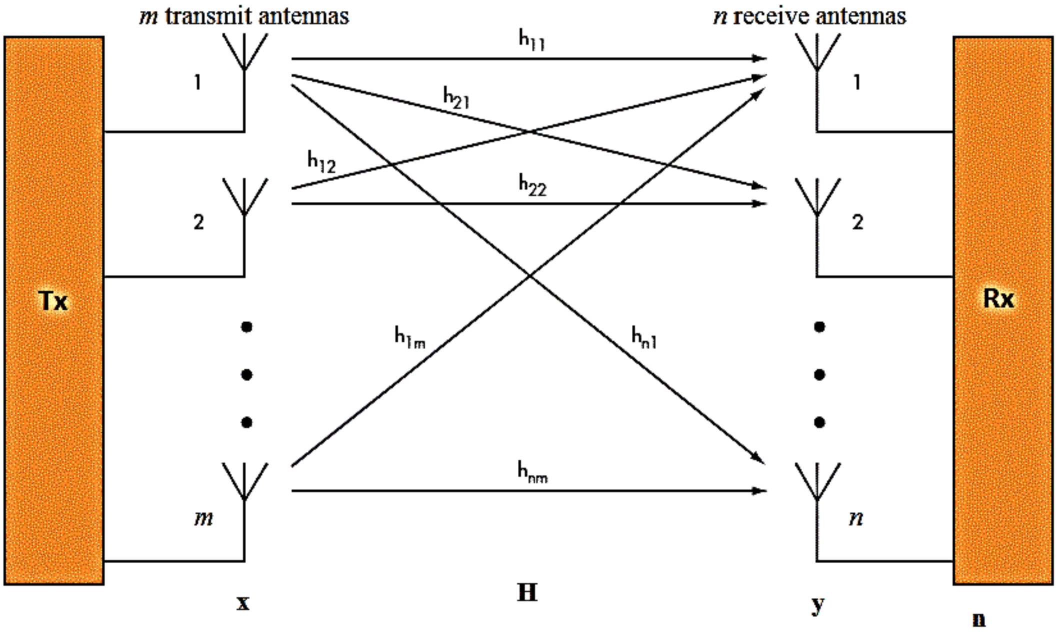


Figure 3.1. A system model of regular MIMO system (no secrecy/eavesdropper).

While the transmission is being performed in *d* directions, if the covariance matrix $\mathbf{R}$ has rank-*d*, then it provides *d*-fold diversity. Full-rank $\mathbf{R}$ provides the *m*-fold diversity. The power allocation is usually based on the waterfilling algorithm (WF) [13]. Based on the assumption that the full channel state information (CSI) is known to the transmitter, we decompose the channel matrix $\mathbf{W}$ as $\mathbf{W} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{+}$, where the columns of $\mathbf{U}$ are the eigenvectors of $\mathbf{W}$, and $\mathbf{\Lambda}$ is the diagonal matrix whose diagonal entries are the eigenvalues of $\mathbf{W}$. Assume that $\lambda_1(\mathbf{W}) \geq \lambda_2(\mathbf{W}) \geq \cdots \geq \lambda_m(\mathbf{W})$ are the ordered eigenvalues of $\mathbf{W}$ which denote the power gain of each corresponding eigenmode, $\lambda_i(\mathbf{A})$ denotes the *i*th eigenvalue of $\mathbf{A}$, then the power is allocated to the eigenmodes based on their strength according to the WF algorithm [1]:

$$P_i^* = \left(\mu - \frac{\sigma^2}{\lambda_i(\mathbf{W})}\right)_+ \tag{3.4}$$

where P_i^* is the power allocated to the i^{th} eigenmode, $(x)_+ = \max(0, x)$;

μ is chosen to satisfy the total power constraint:

$$\sum_{i=1}^{m} \left(\mu - \frac{\sigma_0^2}{\lambda_i(\mathbf{W})}\right)_+ = P_T \tag{3.5}$$

This WF algorithm is based on special dimensions and results in the capacity being

$$C = \sum_{i=1}^{m} \ln\left(1 + \frac{P_i^*(\lambda_i(\mathbf{W}))\lambda_i(\mathbf{W})}{\sigma_0^2}\right) \tag{3.6}$$

In the case of beamforming, there is only one nonzero eigenmode such that the capacity has the form

$$C = \ln\left(1 + \frac{P_T\lambda}{\sigma_0^2}\right) \tag{3.7}$$

where λ is the power gain in the nonzero eigenmode.

3.2 Wiretap MIMO Channel and Secrecy Capacity

The wiretap channel, which was introduced in [17] by Wyner, is a broadcast channel where one of the receivers, the eavesdropper, tries to access information illegally. In this thesis, we are focused on passive eavesdroppers that listen to source information without modifying or injecting information [14]. Figure 3.2 shows the wiretap channel model.

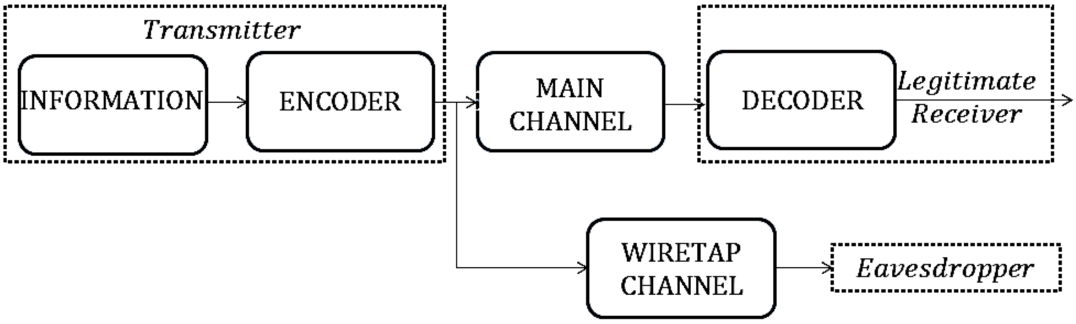


Figure 3.2. Wiretap Channel Model.

In the Gaussian wiretap channel, the additive white Gaussian noise (AWGN) corrupts the outputs at both legitimate receiver and eavesdropper. The secrecy capacity of the Gaussian wiretap channel is given by [24]

$$C_s = \left[\ln\left(1+\frac{P_T}{\sigma^2}\right) - \ln\left(1+\frac{P_T}{\upsilon^2}\right)\right]_+ \tag{3.8}$$

where σ^2 and υ^2 are noise powers of the legitimate channel and eavesdropper's channel respectively. Formula (3.8) is given by the difference between the capacity of the legitimate channel and the eavesdropper's channel which can be considered as a property of some special cases of memoryless discrete wiretap channels [20].

For the MIMO Gaussian wiretap channel, n_e is the number of antennas employed by the eavesdropper. The secrecy capacity of the MIMO wiretap channel is defined as the maximized secrecy rate $C_S(\mathbf{R})$ of information can be reliably and securely transmitted subject to total transmit power and given by [14]:

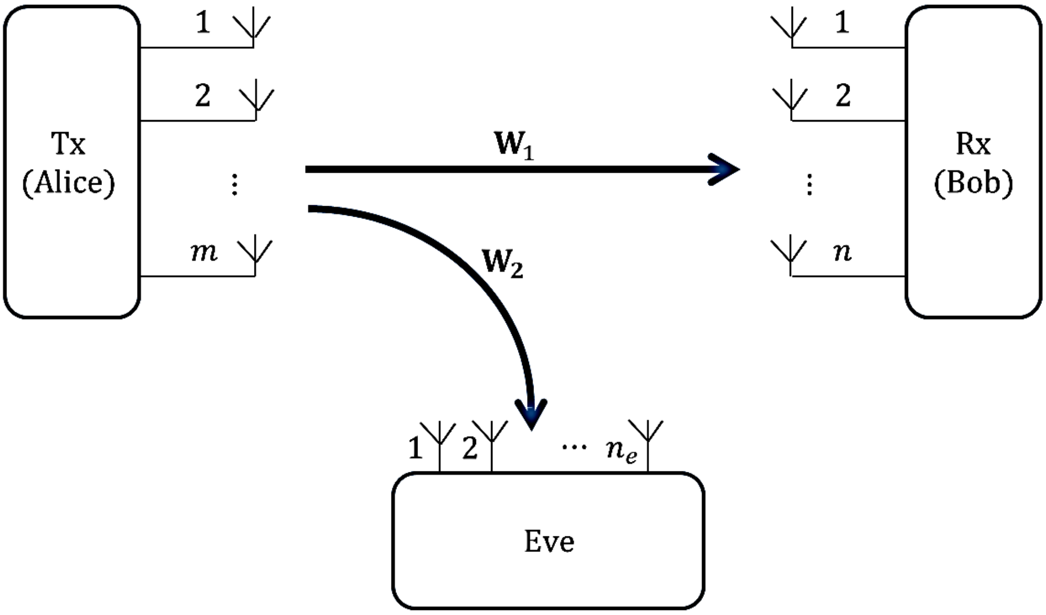


Figure 3.3. System model of MIMO wiretap channel.

$$C_s = \max_{\mathbf{R}} C_s(\mathbf{R}) = \max_{\mathbf{R}}\left\{\ln\frac{|\mathbf{I}+\mathbf{W}_1\mathbf{R}|}{|\mathbf{I}+\mathbf{W}_2\mathbf{R}|}\right\};\ \text{s.t.}\,\mathbf{R}\geq\mathbf{0}, \text{Tr}(\mathbf{R})\leq P_T \tag{3.9}$$

where **|A|** denotes the determinant of matrix **A**; $\mathbf{H}_1$ is the $n\times m$ channel matrix of the legitimate receiver; $\mathbf{H}_2$ is the $n_e\times m$ channel matrix of the eavesdropper,

$\mathbf{W}_1 = \mathbf{H}_1^+\mathbf{H}_1$, $\mathbf{W}_2 = \mathbf{H}_2^+\mathbf{H}_2$. Note that for the given channel matrices, secrecy rate $C_s(\mathbf{R})$ (the rate of information that can be transmitted safely and reliably) for the Gaussian MIMO wiretap channel is a function of the covariance matrix. It has been demonstrated that the Gaussian signaling is optimal over the wiretap Gaussian MIMO channels [25] to [30]. The optimal transmit covariance matrix of the 2-1-1 Gaussian MIMO wiretap channel, where the transmitter and the receiver have two antennas while the eavesdropper has only one antenna, has been obtained in [26]. When multiple transmit antennas are employed while the intended receiver and the eavesdropper use single receive antenna (MISO), the optimization problem can be solved easily. This simple case has been demonstrated in [28]. The more general MIMO case has been studied in [30] based on the minimum mean-square error (MMSE) approach. Based on the necessary KKT conditions of optimality, the explicit, closed-form solution has been given for the case where the optimal covariance matrix has full-rank, and the case of weak eavesdropper and high-SNR regime are considered in [30] to [32].

However, for general cases, since the optimization problem in (3.8) is not convex with respect to $\mathbf{R}$ (unless $\mathbf{W}_1 \geq \mathbf{W}_2$), a closed-form solution has not been found under the total transmit power constraint.

3.3 Summary

With the extensive application of MIMO technology in wireless communications in this decade, the security issues of MIMO systems became a new challenge for the industry. Based on information-theoretic secrecy, the general expression of secrecy capacity of Gaussian MIMO wiretap channels has been obtained while an explicit, closed-form optimal solution is still problematic, except for some special cases. The purpose of this thesis is to discuss numerical methods for achieving the secrecy capacity of general Gaussian MIMO wiretap channels and their corresponding transmit covariance matrices.

4. An Introduction to CVX

4.1 Why CVX?

CVX is a popular modeling system for solving convex optimization problems [63]. This modeling system is implemented in MATLAB which allows convex programs to be constructed by common MATLAB functions and operators. It is convenient to use CVX to formulate and solve convex problems such as constrained entropy maximization and determinant maximization [63]. CVX is an important toolbox for solving the problem of optimization the secrecy capacity of Gaussian MIMO wiretap channel throughout this thesis.

4.2 CVX Precision

The numerical results of convex optimization problems obtained by CVX are not exact; they are computed within a predefined numerical precision or tolerance [63]. We will not interpret this variable thoroughly since it might be different in different applications and heavily depends on how does the CVX transform problems into its solvers. While the setting of CVX precision affects the accuracy of results and processing time, we will try to find the proper CVX precision so that the accuracy and processing time are both acceptable.

As mentioned in Chapter 3, it has been proved that the MIMO channel transmit rate $C(\mathbf{R})$, consisting the logarithm of a matrix determinant, is a concave function with respect to the transmit covariance matrix $\mathbf{R}$ [42], and the associated constraints of its maximization ($C = \max_{\mathbf{R}} C(\mathbf{R})$; s.t. $\mathbf{R} \geq \mathbf{0}$, $\mathrm{Tr}(\mathbf{R}) \leq P_T$) are two linear functions which are also concave. Hence the optimization problem for the capacity of MIMO channel is concave which can be conveniently formulated and solved by CVX. This is also considered a special case of the MIMO wiretap channel ($\mathbf{H}_2 = \mathbf{0}$). The optimization problem (3.2) to be processed and solved by CVX is given as

$$C = \max_{\mathbf{R}} C(\mathbf{R}) = \max_{\mathbf{R}} \{\ln |\, \mathbf{I} + \mathbf{WR} \,|\} \,;\mathrm{s.t.} \geq \mathbf{0}; \mathrm{Tr}(\mathbf{R}) \leq P_T \tag{4.1}$$

Covariance matrix $\mathbf{R}$ is the variable of this problem and has to be positive semi-definite. Based on the total power constraint, $\mathrm{Tr}(\mathbf{R})$ representing the total input power cannot be greater than P_T. Note that as mentioned in the last chapter, the noise power σ^2 is set to be 1 so that SNR is equivalent to P_T. For the given $\mathbf{W}$, the capacity and corresponding optimal covariance matrix $\mathbf{R}^*$ can be obtained by CVX. The accuracy of the result is controlled by the CVX precision variable. We chose some channel matrixes which have different characteristics to verify the impact of CVX precision variable.

The MATLAB code of this problem is given as follows,

```
rho=10^(SNRdB/10); % SNR in linear domain
  cvx_begin; % CVX begins
   cvx_precision(ε); % set CVX precision, ε can be 10^-1, 10^-4, 10-^16
   variable R(m,m) symmetric; % define variable (covariance matrix)
   R == semidefinite(m); % R has to be positive semidefinite
   C= log_det(eye(m)+W*R);
 maximize C;
  0<=trace(R)<= m*rho; % Tr(R) is less than total transmit power
cvx_end; % CVX ends
```

Case 4-1: $\mathbf{W} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ **(I)**, i.e. the channel matrix is of full-rank and has identical eigenvalues (power gains of each eigen direction). This channel matrix represents for isotropic antennas and propagation, the optimal covariance has been known to be scaled identity matrix [1].

Table 4.1. The results of Case 4-1 returned by CVX.

	SNR=-10 dB		SNR=20 dB		
CVX precision	Capacity [nats/s/Hz]	$\mathbf{R}^*$	Capacity [nats/s/Hz]	$\mathbf{R}^*$	Time [s]
10^{-1}	0.0743	$\begin{bmatrix} 0.049 & 0 \\ 0 & 0.049 \end{bmatrix}$	7.7677	$\begin{bmatrix} 49.25 & 0 \\ 0 & 49.25 \end{bmatrix}$	1.22
10^{-4}	0.0976	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	7.8637	$\begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}$	1.25
10^{-16}	0.0976	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	7.8637	$\begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}$	2.64
Default	0.0976	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	7.8637	$\begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}$	2.03

By observing Table 4.1, we see that regardless of the CVX precision, all of the optimal input covariance matrixes are full-rank and scaled identity matrixes. It can be concluded that the power is allocated equally into each direction, as expected. Lower precision setting (10^{-1}) allows the system to return results faster but it sacrifices accuracy. This can be observed from $\mathrm{Tr}(\mathbf{R}^{*})$ which is less than P_T. A reasonable explanation is that the precision has been reached before the power being used totally.

Proposition 4.1: In low SNR regime, the channel capacity is achieved by any covariance matrix when $\mathbf{W}$ is a scaled identity matrix, i.e. $\alpha\mathbf{I}$

Proof:

At low SNR, the channel transmission rate formula has the following approximation:

$$C(\mathbf{R}) = \ln|\mathbf{I} + \mathbf{WR}| \approx \sum_{i=1}^{m} \ln|1 + \lambda_i(\mathbf{W}_1\mathbf{R})| \approx \sum_{i=1}^{m} \lambda_i(\mathbf{WR}) \qquad (4.2)$$

$$= \sum_{i=1}^{m} \lambda_i(\mathbf{WR}) = \sum_{i=1}^{m} \alpha \times \lambda_i(\mathbf{R}) = \alpha\mathrm{Tr}(\mathbf{R}) = \alpha P_T$$

For any $\mathbf{R}$, including optimal one, so that any $\mathbf{R}$ is optimal and able to achieve the channel capacity $C = \max_{\mathbf{R}} C(\mathbf{R})$.

Case 4-2:

Table 4.2. Channel matrix of Case 4-2 and its eigenmodes.

W	$\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$	
Eigenvectors	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
Eigenvalues	2	1

Note that this channel matrix has two eigenmodes with different power gains. The power allocation should be based on the WF. We can observe from Table 4.3 that, based on the WF, in low SNR regime (SNR $= -10$ dB), $\mathbf{R}^{*}$ allocated total power to the strongest eigenmode. However, at high SNR regime (SNR $= 20$ dB), $\mathbf{R}^{*}$ allocated power to each eigenmode almost equally, as expected from WF.

Table 4.3. The results of Case 4-2 returned by CVX.

	SNR=−10 dB		SNR=20 dB		
CVX precision	Capacity [nats/s/Hz]	$\mathbf{R}^*$	Capacity [nats/s/Hz]	$\mathbf{R}^*$	Time [s]
10^{-1}	0.1774	$\begin{bmatrix}0.0975 & 0\\ 0 & 0.0019\end{bmatrix}$	8.5258	$\begin{bmatrix}50.1985 & 0\\ 0 & 49.2530\end{bmatrix}$	0.75
10^{-4}	0.1823	$\begin{bmatrix}0.1 & 0\\ 0 & 0\end{bmatrix}$	8.5470	$\begin{bmatrix}50.2749 & 0\\ 0 & 49.7253\end{bmatrix}$	1.22
10^{-16}	0.1823	$\begin{bmatrix}0.1 & 0\\ 0 & 0\end{bmatrix}$	8.5470	$\begin{bmatrix}50.2504 & 0\\ 0 & 49.7496\end{bmatrix}$	2.51
Default	0.1823	$\begin{bmatrix}0.1 & 0\\ 0 & 0\end{bmatrix}$	8.5470	$\begin{bmatrix}50.2504 & 0\\ 0 & 49.7496\end{bmatrix}$	2.07

Case 4-3:

Table 4.4. Channel matrix of Case 4-3 and its eigenmodes.

$\mathbf{W}$	$\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}$	
Eigenvectors	$\begin{bmatrix}-0.7071\\ 0.7071\end{bmatrix}$	$\begin{bmatrix}0.7071\\ 0.7071\end{bmatrix}$
Eigenvalues	0	2

In this case, the channel matrix has rank-one, which means that there is only one non-zero eigenmode so that all of the power should be allocated to that direction with non-zero power gain, i.e. $\mathbf{R}^*$ is beamforming.

Table 4.5. The results of Case 4-3 returned by CVX.

	SNR=-10 dB		SNR=20 dB		
CVX precision	Capacity [nats/s/Hz]	$\mathbf{R}^*$	Capacity [nats/s/Hz]	$\mathbf{R}^*$	Time [s]
10^{-1}	0.1507	$\begin{bmatrix}0.0473 & 0.0432\\ 0.0432 & 0.0473\end{bmatrix}$	5.2099	$\begin{bmatrix}48.8058 & 46.5481\\ 46.5481 & 48.8058\end{bmatrix}$	0.73
10^{-4}	0.1823	$\begin{bmatrix}0.05 & 0.05\\ 0.05 & 0.05\end{bmatrix}$	5.3033	$\begin{bmatrix}50.0000 & 49.9999\\ 49.9999 & 50.0000\end{bmatrix}$	1.13
10^{-16}	0.1823	$\begin{bmatrix}0.05 & 0.05\\ 0.05 & 0.05\end{bmatrix}$	5.3033	$\begin{bmatrix}50 & 50\\ 50 & 50\end{bmatrix}$	2.35
Default	0.1823	$\begin{bmatrix}0.05 & 0.05\\ 0.05 & 0.05\end{bmatrix}$	5.3033	$\begin{bmatrix}50 & 50\\ 50 & 50\end{bmatrix}$	2.05

Table 4.5 indicates that regardless of whether the SNR is low or high, all of the optimal covariance matrices is of rank-one (approximate rank-one when precision is low), i.e. beamforming as expected [3].

Comparing these results, we can conclude that better CVX precision achieves more accurate results but also requires more processing time. However, when we double the precision variable, the processing time does not increase at the same rate. In other words, the processing time is acceptable if we require a better precision for more accurate results.

We can also observe that precision of 10^{-1} is not a proper setting, even if it returns results faster. Once we improve the precision, we find that a precision of 10^{-4} is able to achieve sufficiently accurate results without sacrificing efficiency. Any higher precision will not yield significant improvement but will increase processing time. Hence in the subsequent parts, we set the CVX precision to 10^{-4}. As mentioned at the beginning of this chapter, CVX precision variable is a tolerance level of numerical computation but is not the accuracy of the results [63].

4.3 Summary

As we have observed in this chapter, CVX is a good numerical toolbox to compute the MIMO channel capacity and corresponding optimal transmit covariance matrix for a given channel matrix. Since the optimization problem for secrecy capacity of the Gaussian MIMO wiretap channel introduced in Chapter 3 is not concave unless $\mathbf{W}_1 - \mathbf{W}_2 \geq \mathbf{0}$, (i.e. degraded channel). Even if $\mathbf{W}_1 - \mathbf{W}_2$, CVX cannot process such objective directly neither. In the next chapter, other numerical methods for handling this non-convex optimization problem will be discussed.

5 The Methods of Random Optimization

Except for some specific channels, the analytical solution for the optimal transmit covariance matrix of the Gaussian MIMO wiretap channel is unknown. However, since we still need to find a numerical solution of this optimization problem, the Monte Carlo optimization and Differential Evolution algorithm will be discussed in this chapter.

5.1 Monte Carlo Optimization

Monte Carlo (MC) optimization is an efficient random optimization method for solving mathematical programming problems whose closed-form solutions are difficult to obtain [78]. The problem is associated with a certain probability model and simulated by computer in order to obtain an approximate solution. Based on a given probability distribution of the probability model, MC approaches an optimum by randomly (or pseudo-randomly) drawing a set of a large number of feasible solutions and selecting the optimal one as a numerical solution. In other words, it uses statistical methods to estimate the numerical characteristics of the probability model thereby obtaining a numerical solution of practical problems.

In this section, MC will be used to compute the secrecy capacity of Gaussian MIMO wiretap channel. In some cases where *m* is large, the convergence time will be long as shown in Figures 5.2 and 5.3. In this section, we will investigate the impacts introduced by different types of channels and *m* on the speed of convergence and accuracy of results. Special considered cases here are selected carefully where the analytical solutions have been available to provide us confidence for different algorithms. We begin with cases of regular MIMO channels without eavesdropper, where the results returned by CVX can be thought as accurate results.

Performance of Monte Carlo for the Regular MIMO channel Capacity

The optimization problem for the channel capacity for a given Gaussian MIMO channel with covariance matrix $\mathbf{R}$ is formulated as

$$C = \max_{\mathbf{R}} C(\mathbf{R}) = \max_{\mathbf{R}} \left\{ \ln\det\left(\mathbf{I} + \frac{1}{\sigma_0^2}\mathbf{WR} \right) \right\}; \mathrm{s.t.}\, \mathbf{R} \geq \mathbf{0}; \mathrm{Tr}(\mathbf{R}) \leq P_T \qquad (5.1)$$

where $C(\mathbf{R})$ denotes the channel transmit rate which is a function of $\mathbf{R}$. (5.1) can be solved approximately by the following method. A set of transmit covariance matrices are randomly generated as follows,

$$\mathbf{A} = randn(m,m)^3 \rightarrow \mathbf{R} = P_T \frac{\mathbf{A}^+\mathbf{A}}{\mathrm{Tr}(\mathbf{A}^+\mathbf{A})} \qquad (5.2)$$

where $\mathbf{A}$ is an $m \times m$ matrix whose entries are normally distributed with zero mean and unity variance. $\mathbf{A}^+\mathbf{A}$ is a positive semidefinite (PSD) matrix such that $\mathbf{R}$ is a PSD matrix whose sum of diagonal entries is P_T.

Therefore, the transmit covariance matrix is built randomly and the corresponding transmit rate $C(\mathbf{R})$ is computed. The $\mathbf{R}$ yielding largest rate will be selected as the optimal covariance matrix $\mathbf{R}^*$. The more the transmit covariance matrices are tested, the higher the probability that the result approaches true capacity (see Figures 5.2 – 5.4).

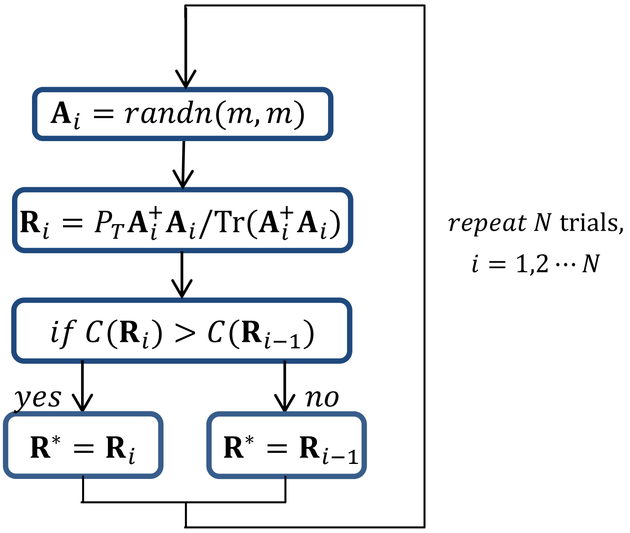

$\mathbf{A}_i = randn(m,m)$
$\mathbf{R}_i = P_T\mathbf{A}_i^+\mathbf{A}_i/\mathrm{Tr}(\mathbf{A}_i^+\mathbf{A}_i)$
$if\ C(\mathbf{R}_i) > C(\mathbf{R}_{i-1})$
yes
no
$\mathbf{R}^* = \mathbf{R}_i$
$\mathbf{R}^* = \mathbf{R}_{i-1}$
$repeat\ N$ trials,
$i = 1,2 \cdots N$

Figure 5.1. Flow chart of Monte Carlo Optimization for (5.1).

In the following cases, the MC's performance (efficiency and accuracy) for computing the transmit capacity of the given channel matrices that have different features will be validated. We will begin with the case where the optimal transmitting is beamforming ($\text{rank}(\mathbf{R}^*) = 1$).

Case 5-1: $\mathbf{W}_1 = \frac{1}{m}\begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}_{m\times m} = \frac{1}{m}\mathbf{E}_{m\times m}$,

i.e. $\mathbf{W}_1$ has rank-one which means there is only one positive eigen direction. The power should be transmitted into the non-zero eigenmode only (beamforming) [13], hence that direction needs to be found. The capacities for different SNR and m obtained by MC vs N (the number of trials of MC) are shown in Figures $5.2 - 5.4$. To make the results consistent, each computation process is repeated M iterations. Therefore, the results shown in Figures $5.2 - 5.4$ are the average values over the $M = 50$ iterations. To show the convergence of the results, the standard deviation (SD) in each trial is computed as follows

$$SD = \sqrt{\frac{1}{M}\sum_{j=1}^{M}(C_j - \bar{C})^2}, \qquad (5.3)$$

is also given, where C_j is the capacity returned in j^{th} iteration, $\bar{C}$ is the average of the results for current trial, M (equal to 50 in this example) is the number of iterations.

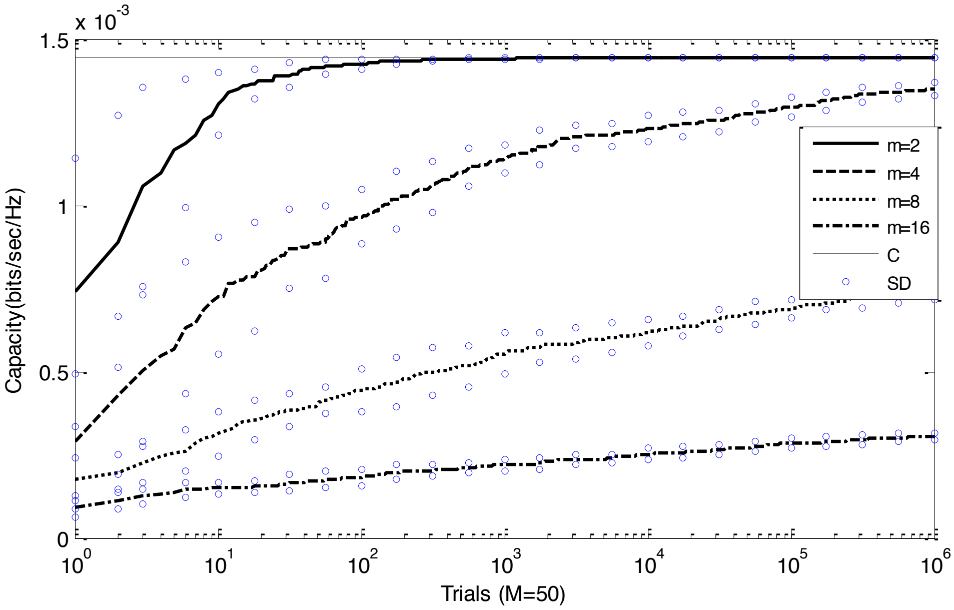


Figure 5.2. The capacity of Case 5-1 obtained by MC vs Trials (SNR $= -30$ dB).

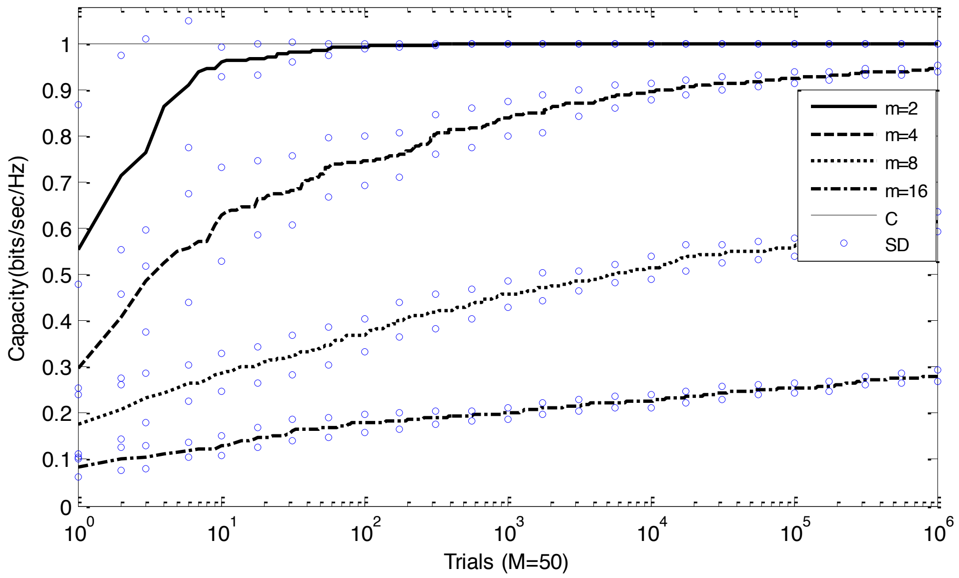


Figure 5.3. The capacity of Case 5-1 obtained by MC vs Trials (SNR $= 0$ dB).

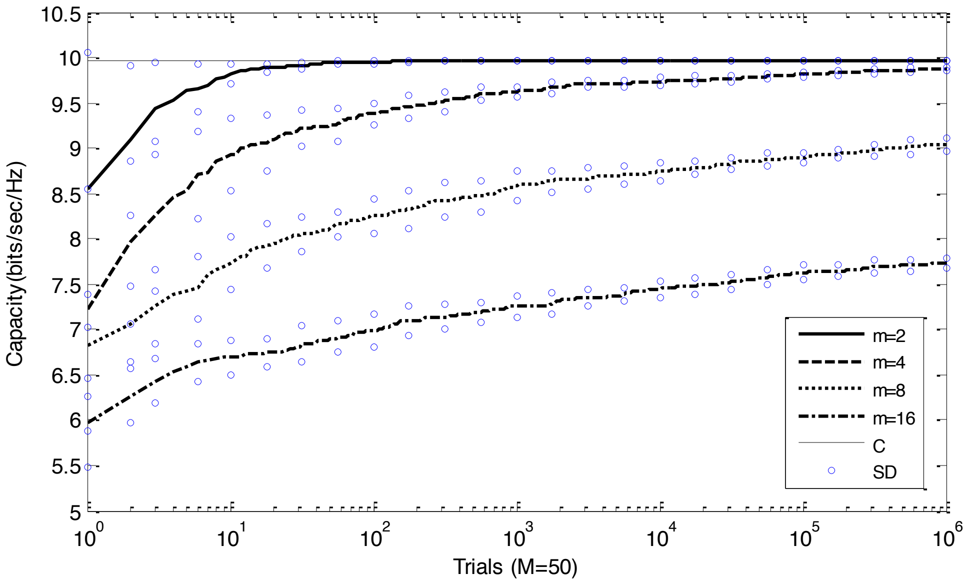


Figure 5.4. The capacity of Case 5-1 obtained by MC vs Trials (SNR $= 30$ dB).

Observing from Figures 5.2 $-$ 5.4, it can be concluded that regardless of the values of SNR, the convergence of the algorithm is fast when m is small ($m = 2$), and an accurate enough result can be obtained before the 500^{th} trial. A comparison of the results obtained by MC and CVX is shown in Tables 5.1 – 5.2.

Table 5.1. The results of Case 5-1 returned by CVX and MC $(N = 500,\ m = 2)$.

SNR[dB]	CVX (precision:10^{-4})		MC	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	1.20	0.0014	0.005	0.0014
0	1.58	1	0.005	0.99
30	1.87	9.97	0.006	9.96

As shown in Table 5.1, when $m = 2$, the capacity obtained by MC is much close to the capacity obtained by CVX, and the processing time of MC is much shorter than the processing time of CVX. Therefore, it can be concluded that MC optimization is the better method for computing the capacity of a MIMO channel when the number of antennas is small.

Table 5.2. The results of Case 5-1 resulted by CVX and MC $(N = 10^6,\ m = 16)$.

SNR [dB]	CVX (precision:10^{-4})		MC	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	1.66	0.0014	31.37	0.0003
0	1.79	1	25.77	0.27
30	6.04	9.97	25.36	7.76

However, in Table 5.2 where m is large, even if we increase N to 10^6, the performance of MC is noticeably worse than that of CVX. A possible explanation of this phenomenon is that when $m = 16$, the randomly generated transmit covariance matrix $\mathbf{R}$ is a 16×16 matrix whose rank has 16 possibilities; hence the probability of $\text{rank}(\mathbf{R}) = 1$ is much lower than that in the case where $m = 2$. In other word, it is difficult for MC to find the optimal eigen direction and allocate all of the power to the optimal direction when the size of the channel matrix is large. Table 5.3 shows that regardless of m, the probability of $\text{rank}(\mathbf{R}) = 1$ is zero which means that the approximate optimal transmit covariance matrix obtained by MC based on (5.2) can only approximate a rank-one matrix (approximate beamforming) instead of exact rank-one matrix even if m is small. The method for generating the results in

Table 5.3 is as follows: in $N \cdot M$ trials, we count the number of randomly generated covariance matrices whose rank is 1 as r_1 and count the number of covariance matrices whose rank is m as r_2, so that the empirical probability of $\text{rank}(\mathbf{R}) = 1$ i.e. $P(\text{rank}(\mathbf{R}) = 1) = r_1/(N \cdot M)$ and the empirical probability of $\text{rank}(\mathbf{R}) = m$ i.e. $P(\text{rank}(\mathbf{R}) = m) = r_2/(N \cdot M)$.

Table 5.3. $P(\text{rank}(\mathbf{R}) = 1)$ and $P(\text{rank}(\mathbf{R}) = m)$
(the number of samples are $N \cdot M = 10^6 \times 50$).

	$m = 2$	$m = 4$	$m = 8$	$m = 16$
$P(\text{rank}(\mathbf{R}) = 1)$	0	0	0	0
$P(\text{rank}(\mathbf{R}) = m)$	1	1	0.999	1

If we investigate a weaker condition of beamforming which is most of the power is allocated to one direction, i.e. the largest eigenvalue of $\mathbf{R}$ ($\lambda_{\max}(\mathbf{R})$) is close to P_T. Table 5.4 gives the empirical probabilities of $\lambda_{\max}(\mathbf{R}) \geq 0.95P_T$, ($P(\lambda_{\max}(\mathbf{R}) \geq 0.95 \cdot P_T)$) for different m. In $N \cdot M$ trials, we count the number of randomly generated covariance matrices whose largest eigenvalue is greater than $0.95P_T$ as r_p, so then $P(\lambda_{\max}(\mathbf{R}) \geq 0.95 \cdot P_T) = r_p/(N \cdot M)$.

Table 5.4. $P(\lambda_{\max}(\mathbf{R}) \geq 0.95 \cdot P_T)$
(the number of samples are $N \cdot M = 10^6 \times 50$).

	$m = 2$	$m = 4$	$m = 8$	$m = 16$
$P(\lambda_{max}(\mathbf{R}) \geq 0.95 \cdot P_T)$	0.44	2×10^{-4}	0	0

Table 5.4 shows that the probability of $\lambda_{\max}(\mathbf{R}) \approx P_T$ when $m = 2$ is much higher than that probabilities for other m. This phenomenon reveals the reason that the results obtained by MC converge to the capacity fast when m is small but slow when m is large. Furthermore, the larger the m is, the narrower the beamwidth of antenna pattern is, so that the system has a good directivity when m is large. However, when we use MC to obtain the capacity, the impact introduced by the gap between the main beam of the antenna pattern and the direction of the propagation of

power provided by **R** returned by MC will be enormous when m is large. In other words, even if there is only partial power allocated to the direction of main beam, considerable transmit rate can be achieved when the beamwidth is wide. When the beamwidth becomes marrow, the considerable transmit rate is difficult to be achieved when the power is allocated to a suboptimal eigen direction (see Figure 5.5).

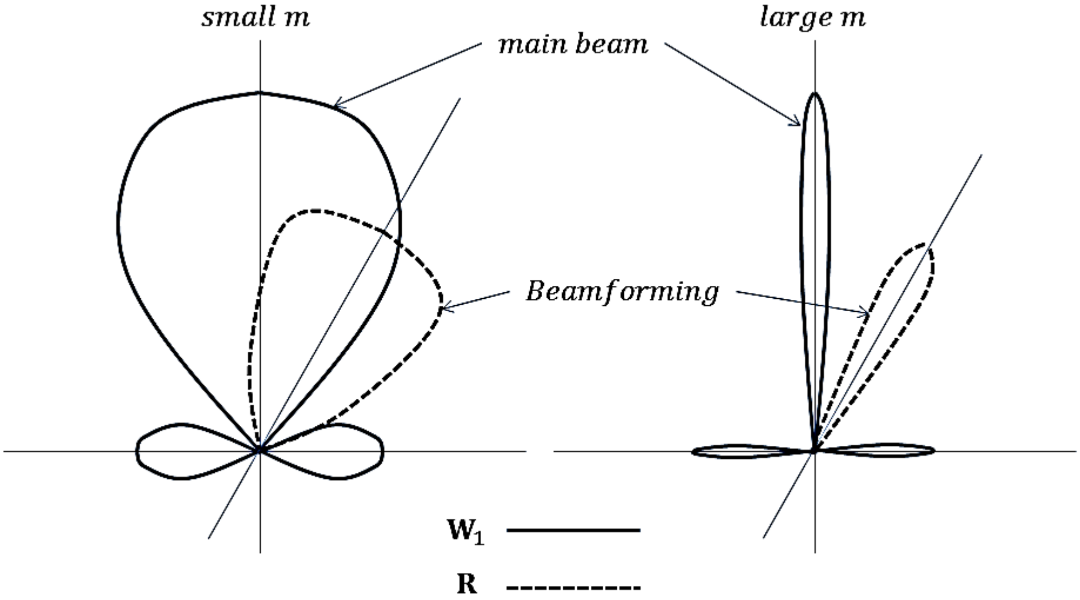


Figure 5.5. Schematic diagram of antenna pattern and beamforming.

Case 5-2: $\mathbf{W}_1 = (1/\sum_{i=1}^{m} i) \cdot \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & m \end{bmatrix}_{m\times m}$,

i.e. $\mathbf{W}_1$ is a diagonal matrix and has full-rank. This channel has m eigenmodes associated with different power gains. The optimal power allocation strategy can be solved by the WF algorithm [1]. The capacities for different SNR and m obtained by MC vs *N* are shown in Figures $5.6 - 5.8$.

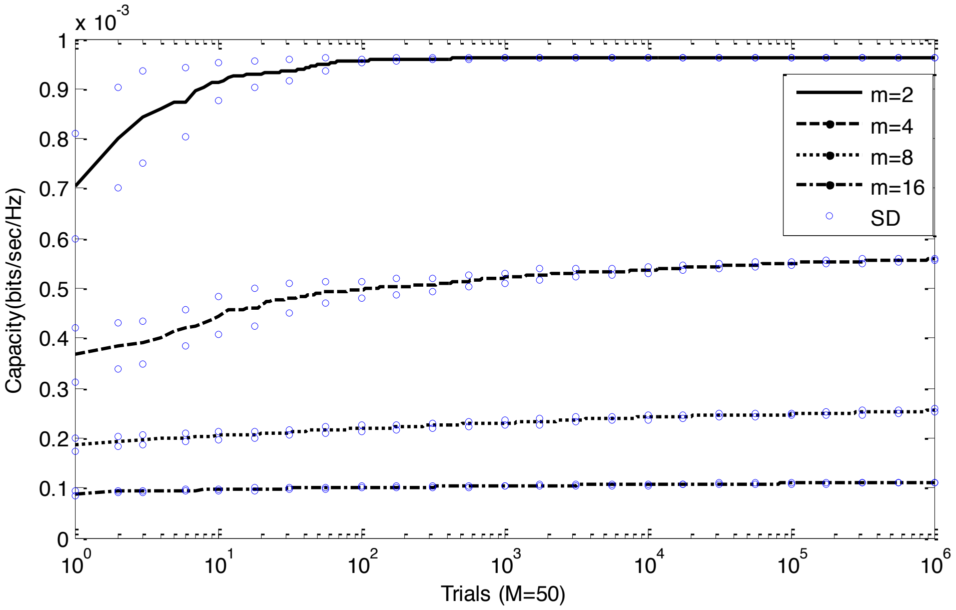


Figure 5.6. The capacity of Case 5-2 obtained by MC vs Trials (SNR $= -30$ dB).

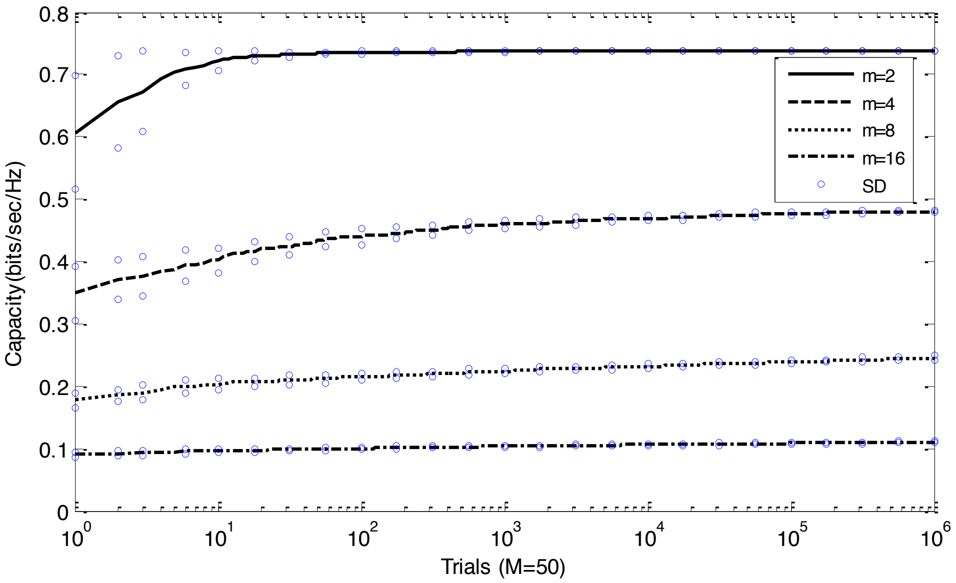


Figure 5.7. The capacity of Case 5-2 obtained by MC vs Trials (SNR $= 0$ dB).

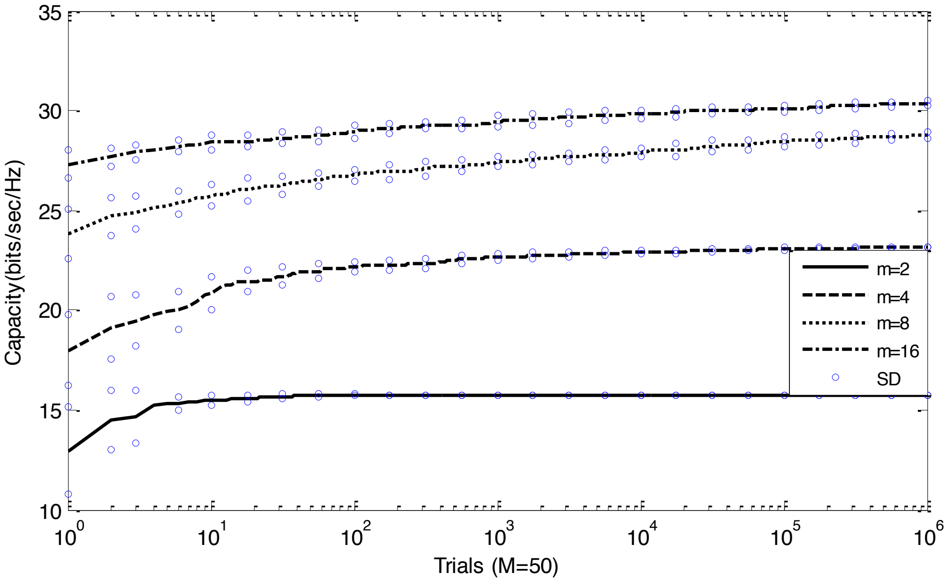


Figure 5.8. The capacity of Case 5-2 obtained by MC vs Trials (SNR = 30 dB).

Table 5.5. The results of Case 5-2 returned by CVX and MC ($N = 500,\ m = 2$).

SNR [dB]	CVX (precision:10^{-4})		MC	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	1.11	9.6×10^{-4}	0.0047	9.6×10^{-4}
0	1.2856	0.7370	0.0069	0.7370
30	1.7451	15.7746	0.0132	15.7746

Table 5.6. The results of Case 5-2 returned by CVX and MC ($N = 10^6,\ m = 8$).

SNR [dB]	CVX (precision:10^{-4})		MC	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	0.4223	3.17×10^{-4}	16.3284	2.53×10^{-4}
0	0.4230	0.2916	15.5230	0.2485
30	0.9271	30.7436	15.5911	28.8305

Table 5.7. The results of Case 5-2 returned by CVX and MC ($N = 10^6,\ m = 16$).

SNR [dB]	CVX (precision:10^{-4})		MC	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	0.5543	1.3×10^{-4}	26.7792	1.1×10^{-4}
0	0.6431	0.1613	25.5704	0.1105
30	1.0872	34.8933	25.9139	30.3155

We can see that regardless of the value of SNR, the speed of convergence is fast when $m = 2$. Accurate enough results can be obtained by MC in 500 trials with short processing time. Comparing Case 5-2 with Case 5-1, Tables 5.6 and 5.7 indicate that MC is still able to obtain relatively accurate results even if m is as large as 8 or 16. Unlike the channel in Case 5-1 that has only one positive eigenmode contributing to the transmit rate, the channel in Case 5-2 has full-rank, i.e. each eigenmode contributes to the transmit rate when the power allocated to that eigenmode is nonzero. Therefore, even if it is difficult for MC to allocate power properly to different eigen directions when m is large, the impact of this issue is not significant in this case. Especially when SNR is large ($\text{SNR} = 30$ dB), the power is approximately equally allocated to different eigenmodes based on WF (Table 5.8), hence the capacities obtained by MC are close to the capacities obtained by CVX. If SNR is small ($\text{SNR} = -30$ dB), most of power should be allocated the strongest eigenmode (Table 5.8), hence the gaps between the capacities obtained by MC and CVX increase (in Tables 5.6 and 5.7, the comparison is based on the percentage but not the absolute difference value) since it is difficult for MC to find that strongest direction.

Table 5.8. Power ($\lambda(\mathbf{R}_{\text{cvx}})$) allocated to eigenmodes with different gains ($\lambda(\mathbf{W}_1)$) of Case 5-2 ($m = 8$).

$\lambda(\mathbf{W}_1)$								
	0.0278	0.0556	0.0833	0.1111	0.1389	0.1667	0.1944	0.2222
$\lambda(\mathbf{R}_{\text{cvx}})$								
$\text{SNR} = 30$ dB	101.3	119.2	125.1	128.2	130	131.2	132.1	132.8
$\lambda(\mathbf{R}_{\text{cvx}}) \cdot 10^3$								
$\text{SNR} = -30$ dB	0.0004	0.0005	0.0006	0.0007	0.0009	0.0013	0.0032	0.992

It is evident that the most difficult part of Monte Carlo is finding the optimal direction(s) but not finding the optimal power allocation strategy. Based on the observation of this section, we can conclude that when we use CVX for the optimization problem of the channel capacity for the regular MIMO channel, the processing time is usually acceptable and increases with the increasing of SNR. Since the capacity formula of the regular channel is concave ($\ln|\mathbf{A}|$ is concave function of $\mathbf{A}$ where $\ln|\mathbf{A}|$ denotes the logarithm of the determinant of $\mathbf{A}$ [42]), the CVX is capable of computing the true capacity.

While using Monte Carlo optimization, SNR value does not affect the processing time much, but the number of transmit antennas does. The gap between the capacity obtained by CVX and the capacity obtained by Monte Carlo becomes larger when the number of transmit antennas is larger (fixed trials). The hardest task for Monte Carlo is finding the optimal direction rather than finding the optimal power allocation strategy. It can be concluded that MC is a stronger method than CVX when m is small (considering the accuracy of results and the processing time).

Monte Carlo versus CVX (Weak Eavesdropper):

Since CVX is not able to solve (3.9) which is

$$C_s = \max_{\mathbf{R}} C_s(\mathbf{R}) = \max_{\mathbf{R}} \left\{ \ln \frac{|\mathbf{I} + \mathbf{W}_1\mathbf{R}|}{|\mathbf{I} + \mathbf{W}_2\mathbf{R}|} \right\};\ \mathrm{s.t.}\, \mathbf{R} \geq \mathbf{0}, \mathrm{Tr}(\mathbf{R}) \leq P_T \qquad (5.4)$$

even if $\mathbf{W}_1 \geq \mathbf{W}_2$, we consider the scenario where the eavesdropper is weak, i.e. $\mathbf{W}_2\mathbf{R} \ll \mathbf{I}$. Based on the following formula,

$$\ln|\mathbf{I} + \mathbf{A}| \approx \mathrm{Tr}(\mathbf{A}) \ \text{ for } \ \mathbf{A} \ll \mathbf{I} \qquad (5.5)$$

the secrecy capacity C_s given in (5.4) can be approximated as

$$C_s \approx C_a = \max_{\mathbf{R}}\{\ln|\mathbf{I} + \mathbf{W}_1\mathbf{R}| - \mathrm{Tr}(\mathbf{W}_2\mathbf{R})\};\ \mathrm{s.t.}\, \mathbf{R} \geq \mathbf{0}, \mathrm{Tr}(\mathbf{R}) \leq P_T \qquad (5.6)$$

when $\mathbf{W}_2\mathbf{R} \ll \mathbf{I}$, i.e. $\lambda_i(\mathbf{W}_2\mathbf{R}) \ll 1$; C_a denotes the approximated secrecy capacity.

The first term of (5.6) is concave and the second term of (5.6) is an affine function which is both concave and convex [42], so that (5.6) is concave for any $\mathbf{R}$ and can be processed by CVX. In this part, we compare the results of (5.5) solved by CVX and the results of (5.4) approximately solved by Monte Carlo.

Proposition 5.1 C_a in (5.6) is the lower bound of the secrecy capacity for any transmit covariance matrix, .i.e.

$$\ln\frac{|\mathbf{I}+\mathbf{W}_1\mathbf{R}|}{|\mathbf{I}+\mathbf{W}_2\mathbf{R}|}-[\ln|\mathbf{I}+\mathbf{W}_1\mathbf{R}|-\mathrm{Tr}(\mathbf{W}_2\mathbf{R})]\geq 0\,;\forall\mathbf{R} \tag{5.7}$$

Proof:

(5.7) is equivalent to

$$\mathrm{Tr}(\mathbf{W}_2\mathbf{R})-\ln|\mathbf{I}+\mathbf{W}_2\mathbf{R}|\geq 0;\forall\mathbf{R} \tag{5.8}$$

that can be reformulated as

$$\sum[\lambda_i(\mathbf{W}_2\mathbf{R})]-\sum\ln[1+\lambda_i(\mathbf{W}_2\mathbf{R})]\geq 0 \tag{5.9}$$

where $[\lambda_i(\mathbf{W}_2\mathbf{R})$ denotes the i^{th} eigenvalue of $\mathbf{W}_2\mathbf{R}$, so that we only need to prove

$$\lambda_i(\mathbf{W}_2\mathbf{R})-\ln[1+\lambda_i(\mathbf{W}_2\mathbf{R})]\geq 0;\ \forall\mathbf{R},i \tag{5.10}$$

The derivative of the left side of (5.9) with respect to $\lambda_i(\mathbf{W}_2\mathbf{R})$ is

$$1-\frac{1}{1+\lambda_i(\mathbf{W}_2\mathbf{R})}\geq 0 \tag{5.11}$$

Hence the left side of (5.9) is monotonically increasing with $\lambda_i(\mathbf{W}_2\mathbf{R})$. The equality is achieved when $\lambda_i(\mathbf{W}_2\mathbf{R})=0$, so that (5.7) can be achieved for any $\mathbf{W}_2\mathbf{R}\geq\mathbf{0}$. It can be concluded that C_a provides the lower bound of C_s.

Since it is difficult for MC to find the optimal eigen direction as it is concluded in last section, we begin with the case where the optimal solution is beamforming (rank$(\mathbf{R})=1$).

Case 5-3: $\mathbf{W}_1=\frac{1}{m}\mathbf{E}_{m\times m}$, $\mathbf{W}_2=\frac{0.1}{m+1}\begin{bmatrix}2 & 1 & \cdots & 1\\ 1 & 1 & \cdots & 1\\ \vdots & \vdots & \ddots & \vdots\\ 1 & 1 & \cdots & 1\end{bmatrix}_{m\times m}$,

i.e. $r_+(\mathbf{W}_1-\mathbf{W}_2)=1$ where $r_+(\mathbf{A})$ denotes the number of positive eigenvalue of $\mathbf{A}$. Based on the discussion in [30], the entire power should be allocated into one specific direction, i.e. $\mathbf{R}^*$ has rank-one. Figures $5.9-5.11$ show the secrecy capacity can be approximately obtained fast (in about 1000 trials) when $m=2$, MC performs well for the optimization problem in (5.4) when the size of the channel is small.

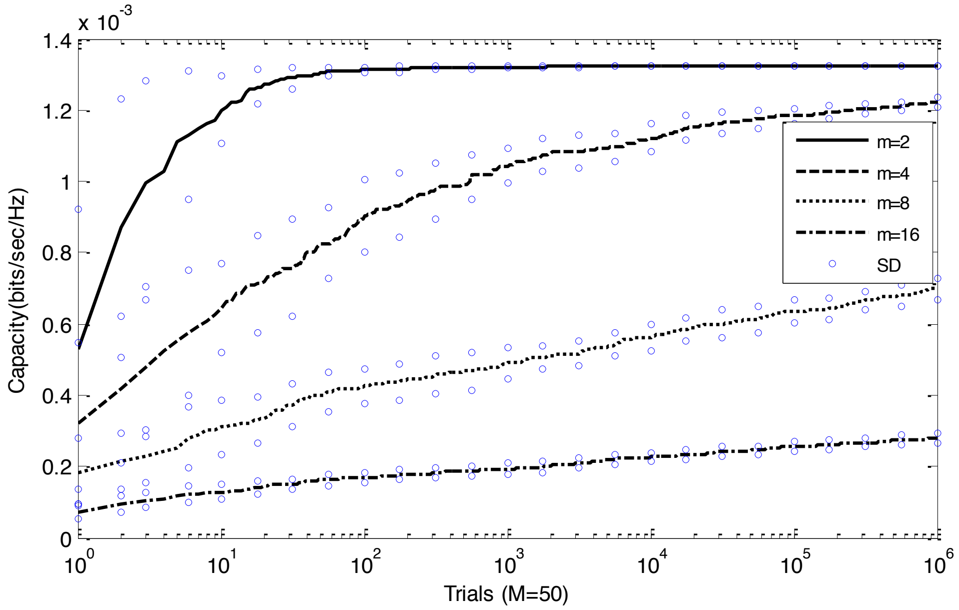


Figure 5.9. The secrecy capacity of Case 5-3 obtained by MC vs Trials (SNR = −30 dB).

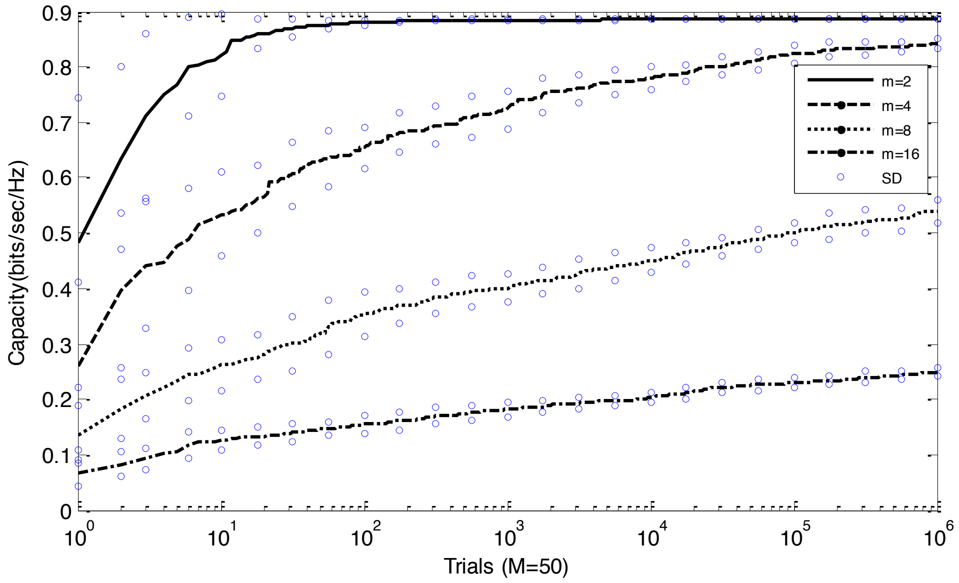


Figure 5.10. The secrecy capacity of Case 5-3 obtained by MC vs Trials (SNR = 0 dB).

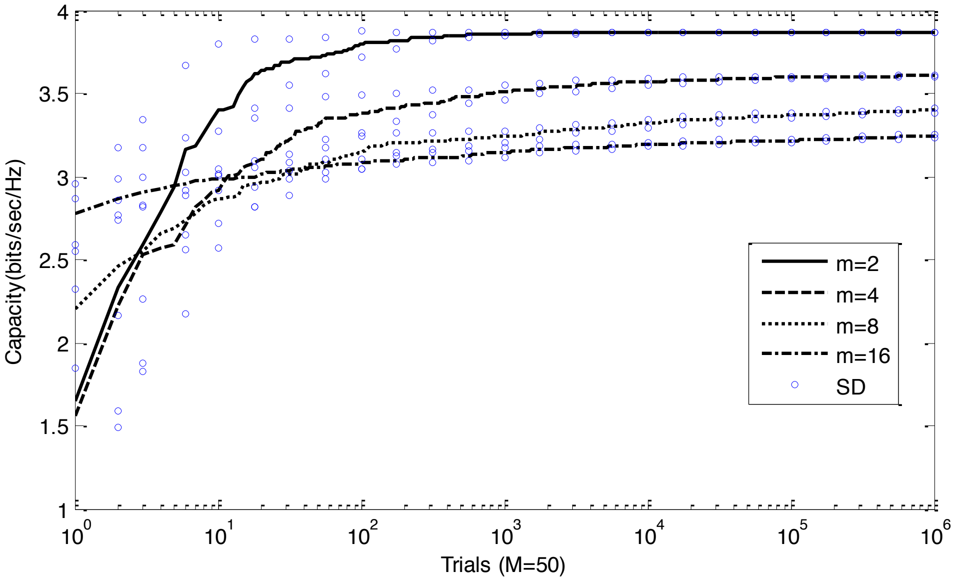


Figure 5.11. The secrecy capacity of Case 5-3 obtained by MC vs Trials (SNR = 30 dB).

An analytical solution of the cases where $r_+(\mathbf{W}_1 - \mathbf{W}_2) = 1$ is introduced in [30] and given as

$$C_s = \ln \lambda_{\max},\ \mathbf{R}^* = P_T \boldsymbol{v}_{\max} \boldsymbol{v}_{\max}^+ \tag{5.12}$$

where $\lambda_{\max}$ and $\boldsymbol{v}_{\max}$ are the largest eigenvalue and corresponding eigenvector of $(\mathbf{I} + P_T\mathbf{W}_2)^{-1}(\mathbf{I} + P_T\mathbf{W}_1)$. Table 5.9 gives the secrecy capacities of the channel in Case 5-3 for different SNR and m based on (5.12).

Table 5.9. The secrecy capacities of Case 5-3.

C_s [bits/s/Hz]	$m = 2$	$m = 4$	$m = 8$	$m = 16$
SNR = −30 dB	0.0013	0.0013	0.0013	0.0013
SNR = 0 dB	0.8853	0.8825	0.8753	0.8698
SNR = 30 dB	3.8682	3.6223	3.4753	3.3948

Observing the results shown in Tables $5.10 - 5.12$, we can conclude that when SNR is low (SNR $= -30$ dB), the secrecy capacity can be approximated obtained by CVX since the condition of (5.6) $\lambda_i(\mathbf{W}_2\mathbf{R}) \ll 1$ can be easily satisfied (unless $\mathbf{W}_2 \gg \mathbf{I}$).

Table 5.10. Secrecy Capacity & Processing Time of Case 5-3 returned by CVX and Monte Carlo (m = 2).

	CVX		Monte Carlo			
SNR [dB]	precision = 10^{-4}		N = 1000		$N = 10^4$	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	1.1144	0.0013	0.0112	0.0013	0.1149	0.0013
0	1.2971	0.8799	0.0119	0.8852	0.1254	0.8853
30	1.7149	2.2625	0.0126	3.8622	0.3304	3.8678

Table 5.11. Secrecy Capacity & Processing Time of Case 5-3 returned by CVX and Monte Carlo (m = 8).

	CVX		Monte Carlo			
SNR [dB]	precision = 10^{-4}		N = 10^6		$N = 10^7$	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	0.4512	0.0013	18.9680	6.8×10^{-4}	188.42	7.7×10^{-4}
0	0.6191	0.8698	19.1579	0.5526	189.74	0.5701
30	0.7208	2.1575	19.0707	3.3847	192.70	3.4058

Table 5.12. Secrecy Capacity & Processing Time of Case 5-3 returned by CVX and Monte Carlo (m = 16).

	CVX		Monte Carlo			
SNR [dB]	precision = 10^{-4}		N = 10^6		$N = 10^7$	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	1.2761	0.0013	32.7387	3.01×10^{-4}	325.62	3.07×10^{-4}
0	1.6216	0.8637	34.2345	0.2558	330.76	0.2881
30	2.4417	2.0977	33.0686	3.2317	330.78	3.2499

MC is more efficient than CVX when m is small ($m = 2$). However, the accuracy of the results obtained by CVX decreases with the increase of SNR since the condition $\lambda_i(\mathbf{W}_2\mathbf{R}) \ll 1$ may not be satisfied when SNR is large. Regardless the value of SNR, MC is able to compute the secrecy capacity efficiently when m is small. However, for the large size channel matrices ($m = 8, m = 16$) in this case, the approximated secrecy capacities can be obtained by MC with large number of trials ($N = 10^7$). The efficiency is sacrificed significantly (processing time is long) but the results are much more accurate than the results obtained by CVX when SNR is high.

Case 5-4: $\mathbf{W}_1$ is full rank matrix;

$$\mathbf{W}_1 = (1/\sum_{i=1}^{m} i)\cdot\begin{bmatrix}1 & 0 & \cdots & 0\\ 0 & 2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & m\end{bmatrix}_{m\times m},\quad \mathbf{W}_2 = 0.1\mathbf{W}_1,$$

i.e. both $\mathbf{W}_1$ and $\mathbf{W}_2$ are full-rank. The different channel is also full-rank and $\mathbf{W}_1 > \mathbf{W}_2 \geq \mathbf{0}$ (degraded channel, $r_+(\mathbf{W}_1 - \mathbf{W}_2) = m$). Hence, the optimal transmit covariance matrix does not have to be a rank-one matrix. The power allocation strategy has some properties which are similar to WF when $\mathbf{R}^*$ is not beamforming.

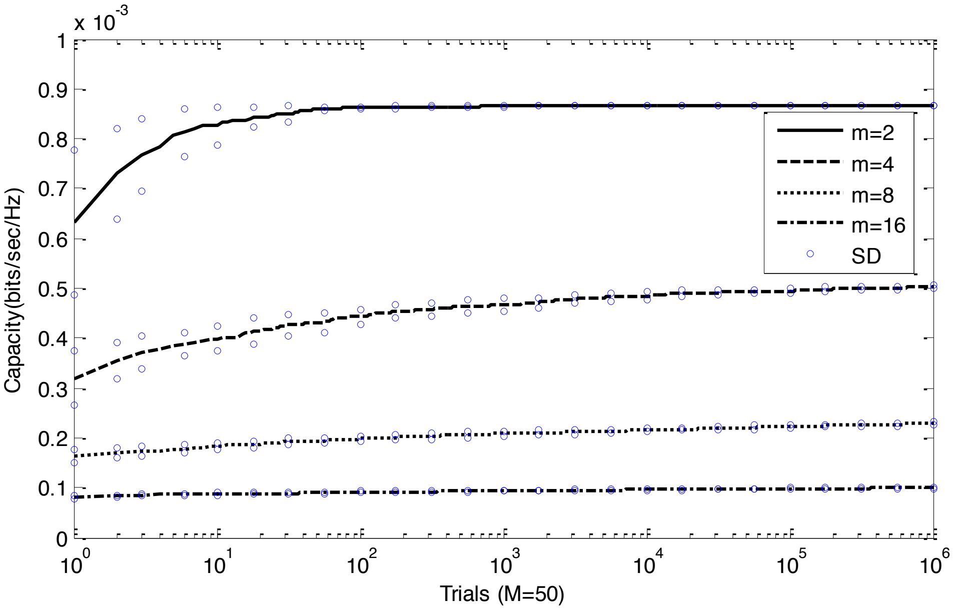


Figure 5.12. The secrecy capacity of Case 5-4 obtained by MC vs Trials (SNR $= -30$ dB).

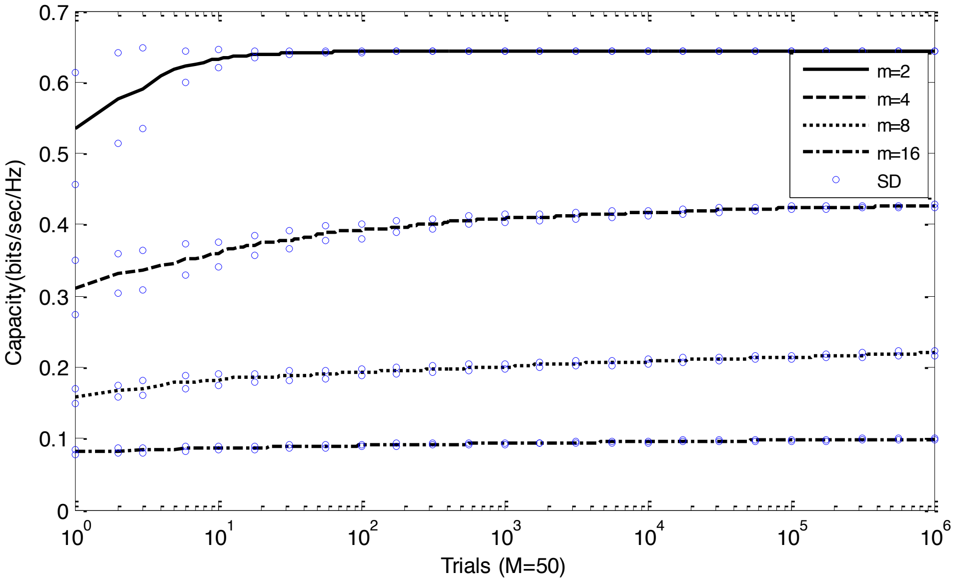


Figure 5.13. The secrecy capacity of Case 5-4 obtained by MC vs Trials (SNR = 0 dB).

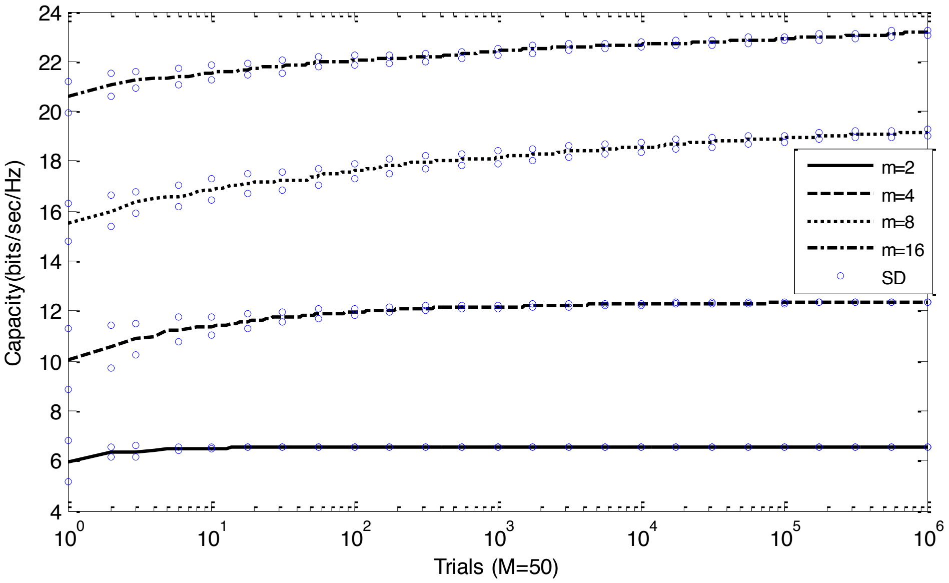


Figure 5.14. The secrecy capacity of Case 5-4 obtained by MC vs Trials (SNR = 30 dB).

Figures $5.12 - 5.14$ demonstrate that regardless of SNR, the secrecy capacity can be obtained fast (in 500 trials) when m is small. Table 5.13 shows that MC performs better than CVX when $m = 2$. However, the results obtained by MC are not accurate when m becomes large even if the optimal solutions may not be beamforming. In Figure 5.14, the relationship of the secrecy capacity and m is different with their relationship shown in Figures 5.12 and 5.13. Higher secrecy capacity corresponds to a larger m in Figure 5.14. In this case, $\mathbf{W}_1$ is proportional with $\mathbf{W}_2$ which means that they have the same eigenvectors and the optimal transmit covariance matrices are same with the optimal covariance matrices in Case 5-2, i.e. $\mathbf{R}^*$ is a full-rank matrix when SNR is high. Based on the discussion in [30],

$$C_s \approx \ln \frac{|\mathbf{W}_1|}{|\mathbf{W}_2|} \tag{5.13}$$

when $\mathbf{R}^*$ has full-rank in high SNR regime. Therefore, when $\mathrm{SNR} = 30\,\mathrm{dB}$, the secrecy capacities in this case are $6.64\ (m = 2), 13.29\ (m = 4), 26.58\ (m = 8)$ and $53.15\ (m = 16)$, i.e. a larger m yields a higher secrecy capacity which follows the results shown in Figure 5.14.

Table 5.13. Secrecy Capacity & Processing Time of Case 5-5 returned by CVX and Monte Carlo (m = 2).

	CVX		Monte Carlo			
SNR [dB]	precision = 10^{-4}		N = 500		N = 10^3	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	1.1770	8.7×10^{-4}	0.0069	8.6×10^{-4}	0.0185	8.6×10^{-4}
0	1.3836	0.6409	0.0066	0.6438	0.0153	0.6438
30	1.8961	4.0470	0.0060	6.5326	0.0123	6.5327

Table 5.14. Secrecy Capacity & Processing Time of Case 5-4 returned by CVX and Monte Carlo (m = 8).

	CVX		Monte Carlo			
SNR [dB]	precision = 10^{-4}		N = 10^6		N = 10^7	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	0.4905	2.8×10^{-4}	18.4318	2.3×10^{-4}	175.3450	2.4×10^{-4}
0	0.4334	0.2603	21.9912	0.2222	205.1646	0.2263
30	1.0652	16.1881	24.8090	19.0056	206.9253	19.2958

Table 5.15. Secrecy Capacity & Processing Time of Case 5-4 returned by CVX and Monte Carlo (m = 16).

	CVX		Monte Carlo			
SNR [dB]	precision = 10^{-4}		$N = 10^6$		$N = 10^7$	
	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]	Time [s]	Capacity [bits/s/Hz]
-30	0.5912	1.17×10^{-4}	30.2734	9.9×10^{-5}	304.9737	1.05×10^{-4}
0	0.6770	0.1446	30.6734	0.0975	304.7918	0.1003
30	1.1525	25.2746	35.3867	23.1709	307.2651	23.2457

In general, when the number of transmit antennas is small (m = 2), the secrecy capacity can be approximately obtained faster and more accurately using MC. We can conclude that Monte Carlo optimization is a good algorithm to approximately compute the secrecy capacity when m is small and that MC is more efficient than CVX. However, MC performs worse as m increases, and increasing the number of trials does not improve the results significant but instead takes enormous amount of processing time. Compared to MC, CVX is able to solve the optimizations approximately in low SNR regime regardless of m. While the results obtained by CVX in high SNR regime can only be considered as the lower bound of the secrecy capacity of a given channel. Neither Monte Carlo nor CVX can handle the optimization problems for secrecy capacity of a Gaussian MIMO wiretap channel properly when m and SNR are both large. We will discuss other methods for obtaining the numerical results in the following sections.

5.2 Differential Evolution

In the last section, we applied Monte Carlo optimization to compute the numerical solutions of the secrecy capacity for a given MIMO wiretap channel. According to the results we found before, MC is able to handle the channel matrices where the number of transmit antennas is small. However, for the cases where the number of transmit antennas is large, the efficiency and the accuracy of the computation are extremely low. Hence, we will apply the differential evolution (DE) algorithm to see if it is able to improve the efficiency and the accuracy of the results.

The method of differential evolution is introduced in [67] and has gained significant popularity (cited by 7987 times). There are three main steps which are *Mutation*, *Crossover* and *Selection*. More specifically, mutation is for generating new parameter matrices, called mutant matrices, by adding the scaled difference between two population matrices to a third population matrix, called target matrix. In the step of crossover, the trial matrix is generated by mixing the parameters of mutant matrix and the parameters of the predetermined target matrix in order to increase diversity. The trial matrix is compared with the target matrix by estimating the values of objective function yielded by them respectively. If the trial matrix yields a better objective function value than target matrix, then the trial matrix is decided to be a member of the following generation, otherwise, the target matrix is retained.

Since our objective is to compute the optimal transmit covariance matrix, we utilize $NP\, m \times m$ matrices $\mathbf{D}_{i,G}$, $i = 1,2,\cdots,NP$ as a population of G^{th} generation, where m is the number of transmit antennas, and *NP* is the number of population that is fixed throughout the optimization process. The methods for generating the NP $m \times m$ matrices $\mathbf{D}_{i,1}, i = 1,2,\cdots NP$ of the first generation and the relationship of $\mathbf{D}_{i,1}$ and $\mathbf{R}_{i,1}$ are given as following

$$\mathbf{D}_{i,1} = randn(m,m) \rightarrow \mathbf{R}_{i,1} = P_T \frac{\mathbf{D}_{i,1}^{+}\mathbf{D}_{i,1}}{\mathrm{Tr}(\mathbf{D}_{i,1}^{+}\mathbf{D}_{i,1})}, i = 1,2,\cdots,NP \qquad (5.14)$$

where $\mathbf{R}_{i,1}$ denotes the i^{th} transmit covariance matrix of the first generation.

Mutation:

For each target matrix $\mathbf{D}_{i,G}$ a mutant matrix is generated according to

$$\mathbf{V}_{i,G+1} = \mathbf{D}_{r_1,G} + F\left(\mathbf{D}_{r_2,G} - \mathbf{D}_{r_3,G}\right) \qquad (5.15)$$

where indexes $r_1, r_2, r_3 \in \{1,2,\cdots,NP\}$ are unique integers, that are chosen randomly from 1 to *NP*. *F*>0 is a constant which controls the amplification of differential variation. Based on the discussion in [67], $F \in [0,2]$.

Crossover:

The i^{th} trial matrix $\mathbf{U}_{i,G+1}$ of the $G+1$ generation has the form

$$\mathbf{U}_{i,G+1} = \begin{bmatrix} u_{11,i,G+1} & \cdots & u_{1m,i,G+1} \\ \vdots & \ddots & \vdots \\ u_{m1,i,G+1} & \cdots & u_{mm,i,G+1} \end{bmatrix}_{m\times m} \tag{5.16}$$

such that the i^{th} trial transmit covariance matrix of the $G+1$ generation has the form

$$\mathbf{R}_{\mathbf{U},i,G+1} = P_T \frac{\mathbf{U}_{i,G+1}^{+}\mathbf{U}_{i,G+1}}{\mathrm{Tr}(\mathbf{U}_{i,G+1}^{+}\mathbf{U}_{i,G+1})}, i = 1,2,\cdots,NP \tag{5.17}$$

where the entries of $\mathbf{U}_{i,G+1}$ are decided as

$$u_{ab,i,G+1} = \begin{cases} v_{ab,i,G+1}, if\big(\text{the } b^{\text{th}}\, rand(1) \le CR\big)\ or\ [a = randi(m), b = randi(m)] \\ d_{ab,i,G}, if\big(\text{the } b^{\text{th}}\, rand(1) > CR\big)\ and\ [a \ne randi(m)\ or\ b \ne randi(m)] \end{cases} \tag{5.18}$$

The variables a and b are integers where $a, b \in [1, m]$; $v(d)_{ab,i,G+1}$ is the entry at a^{th} row, b^{th} column of matrix $\mathbf{V}(\mathbf{D})_{i,G+1}$; the b^{th} *rand*[4](1) ∈ [0,1] is the b^{th} evaluation of a random number generator in the a^{th} row; $CR \in [0,1]$ is the crossover constant which has to be determined by the user; and *randi*[5](*m*) is an integer randomly chosen from 1,2……*m* which ensures that $\mathbf{U}_{i,G+1}$ gets at least one parameter from $\mathbf{V}_{i,G+1}$. The process of Crossover is shown in Figure 5.15.

[4] $r = rand(1)$ returns a pseudorandom scalar drawn from the standard uniform distribution on the open interval (0,1). [5] $r = randi(i_{\max})$ returns a pseudorandom integer between 1 and $i_{\max}$.

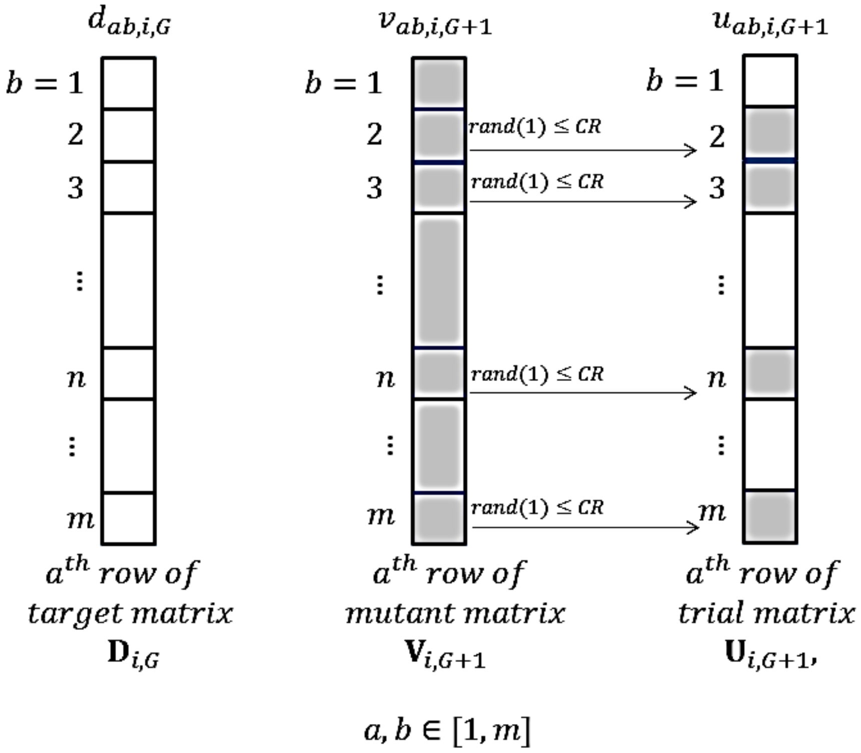

$d_{ab,i,G}$
$v_{ab,i,G+1}$
$u_{ab,i,G+1}$
$b = 1$
2
3
⋮
n
⋮
m
$rand(1) \leq CR$
$rand(1) \leq CR$
$rand(1) \leq CR$
$rand(1) \leq CR$
a^{th} row of target matrix $\mathbf{D}_{i,G}$
a^{th} row of mutant matrix $\mathbf{V}_{i,G+1}$
a^{th} row of trial matrix $\mathbf{U}_{i,G+1}$,
$a, b \in [1, m]$

Figure 5.15. Flow chart of Crossover.

Selection:

To decide whether or not the trial matrix should become a member of next generation G+1, the secrecy capacity given in (5.4) yielded by $\mathbf{U}_{i,G+1}$ ($C_s = C_s(\mathbf{R}_{\mathbf{U},i,G+1})$, where $\mathbf{R}_{\mathbf{U},i,G+1} = P_T\mathbf{U}_{i,G+1}{}^{+}\mathbf{U}_{i,G+1}/\mathrm{Tr}(\mathbf{U}_{i,G+1}{}^{+}\mathbf{U}_{i,G+1})$) is compared to the secrecy capacity yielded by $\mathbf{D}_{i,G}$ ($C_s = C_s(\mathbf{R}_{\mathbf{D},i,G})$, where $\mathbf{R}_{\mathbf{D},i,G} = P_T\mathbf{D}_{i,G}{}^{+}\mathbf{D}_{i,G}/\mathrm{Tr}(\mathbf{D}_{i,G}{}^{+}\mathbf{D}_{i,G})$). If $C_s(\mathbf{R}_{\mathbf{U},i,G+1}) \geq C_s(\mathbf{R}_{\mathbf{D},i,G})$, $\mathbf{D}_{i,G+1} = \mathbf{U}_{i,G+1}$, otherwise $\mathbf{D}_{i,G+1} = \mathbf{D}_{i,G}$. Figure 5.16 shows the flow chart of the Differential Evolution algorithm.

Assuming that there are *N* generations in total, i.e. $G = 1,2 \cdots N$, in the members of the N^{th} generation, $\mathbf{D}_{i,N}$ that yields the largest secrecy capacity is selected as the optimal $\mathbf{D}_{i,N}{}^{*}$ such that the optimal transmit covariance matrix is $\mathbf{R}^{*} = P_T \mathbf{D}_{i,N}{}^{*+} \mathbf{D}_{i,N}{}^{*} / \mathrm{Tr}(\mathbf{D}_{i,N}{}^{*+} \mathbf{D}_{i,N}{}^{*})$.

Based on the discussion in [67], *CR* = 0.9 and *NP* = 200 are proper settings; in the next section, we will discuss the selection of F, $F \in (0,2]$.

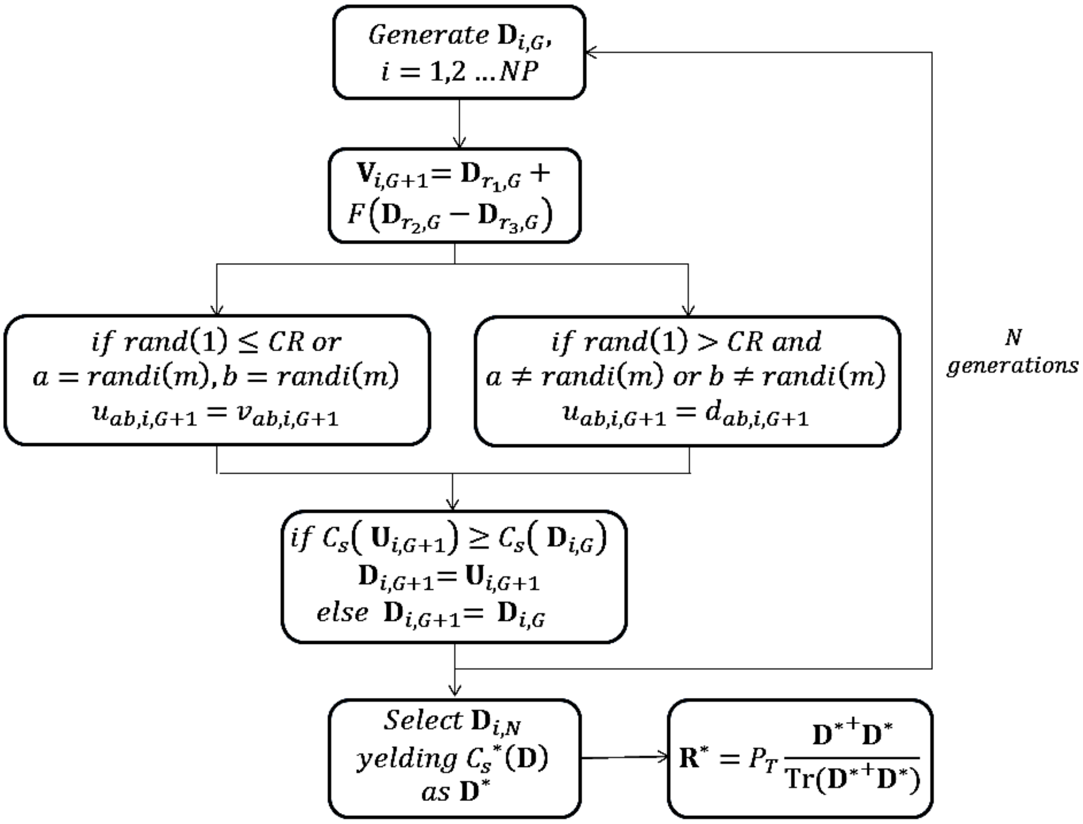

$Generate\ \mathbf{D}_{i,G},\ i = 1,2 \ldots NP$
$\mathbf{V}_{i,G+1} = \mathbf{D}_{r_1,G} + F(\mathbf{D}_{r_2,G} - \mathbf{D}_{r_3,G})$
$if\ rand(1) \leq CR\ or\ a = randi(m), b = randi(m)\ u_{ab,i,G+1} = v_{ab,i,G+1}$
$if\ rand(1) > CR\ and\ a \neq randi(m)\ or\ b \neq randi(m)\ u_{ab,i,G+1} = d_{ab,i,G+1}$
$N\ generations$
$if\ C_s(\mathbf{U}_{i,G+1}) \geq C_s(\mathbf{D}_{i,G})\ \mathbf{D}_{i,G+1} = \mathbf{U}_{i,G+1}\ else\ \mathbf{D}_{i,G+1} = \mathbf{D}_{i,G}$
$Select\ \mathbf{D}_{i,N}\ yelding\ C_s{}^{*}(\mathbf{D})\ as\ \mathbf{D}^{*}$
$\mathbf{R}^{*} = P_T \frac{\mathbf{D}^{*+}\mathbf{D}^{*}}{\mathrm{Tr}(\mathbf{D}^{*+}\mathbf{D}^{*})}$

Figure 5.16. Flow chart of Differential Evolution algorithm.

The Effect of *F*:

In this part, we plot the secrecy capacities obtained by DE with applying different values of *F* vs the number of generations to explore the effect of *F*. Since the disparity between different methods mainly appears in the cases where the number of antennas is large, we will focus on the channel matrices with large *m*.

Case 5-5: $\mathbf{W}_1 = \frac{1}{m}\mathbf{E}_{m\times m}$ $\mathbf{W}_2 = \frac{0.1}{m}\mathbf{E}_{m\times m}$, i.e. both $\mathbf{W}_1$ and $\mathbf{W}_2$ have rank-one. It has been shown that beamforming is the optimal solution in this case. Based on the discussion in [30], $\text{rank}(\mathbf{R}^*) \le r_+(\mathbf{W}_1 - \mathbf{W}_2)$ where $r_+(\mathbf{A})$ denotes the number of positive eigenvalue of $\mathbf{A}$; (5.12) gives the optimal solution of this case where $\mathbf{R}^* = P_T\boldsymbol{v}_{\max}\boldsymbol{v}_{\max}^+ = P_T\mathbf{E}/m$. The secrecy capacity $C_s \approx 3.309$ when $\text{SNR} = 30\text{ dB}$ and $C_s \approx 0.0013$ when $\text{SNR} = -30\text{ dB}$.

Observing from Figures 5.17 and 5.18, the effect of F appears significantly when $m = 8, 16$. We can conclude that $F = 0.5$ and $F = 1$ are not appropriate choices while $F = 1.5$ improves the efficiency significantly. According to (5.15), $F = 0.5$ and $F = 1$ maybe too small to induce the variation between the mutant matrix and the target matrices for some cases. $F \ge 1.5$ can be considered as a proper setting.

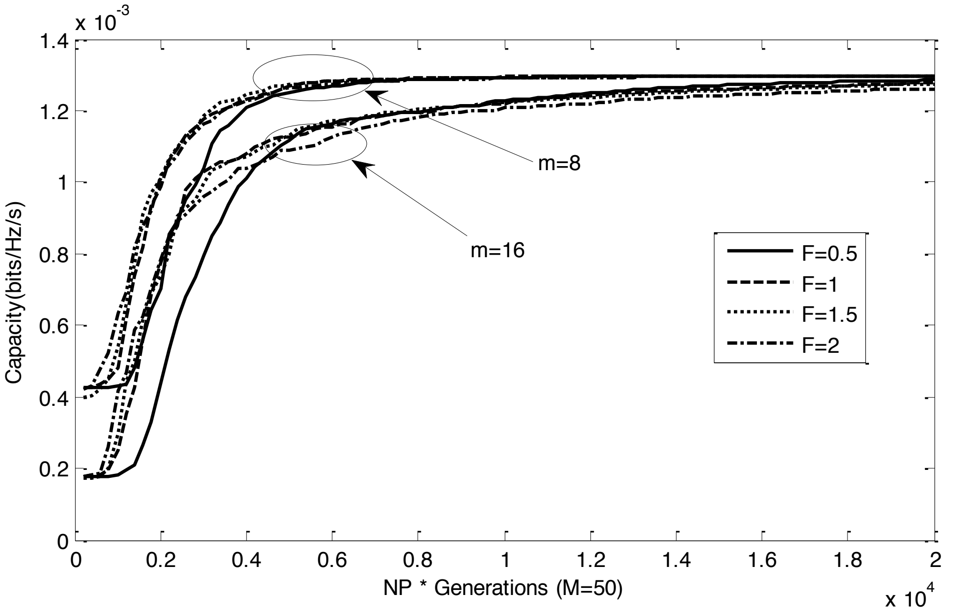


Figure 5.17. The secrecy capacity vs generations of Case 5-5 obtained by DE $m = 8, 16,\ \text{SNR} = -30\text{ dB}\ (NP = 200)$.

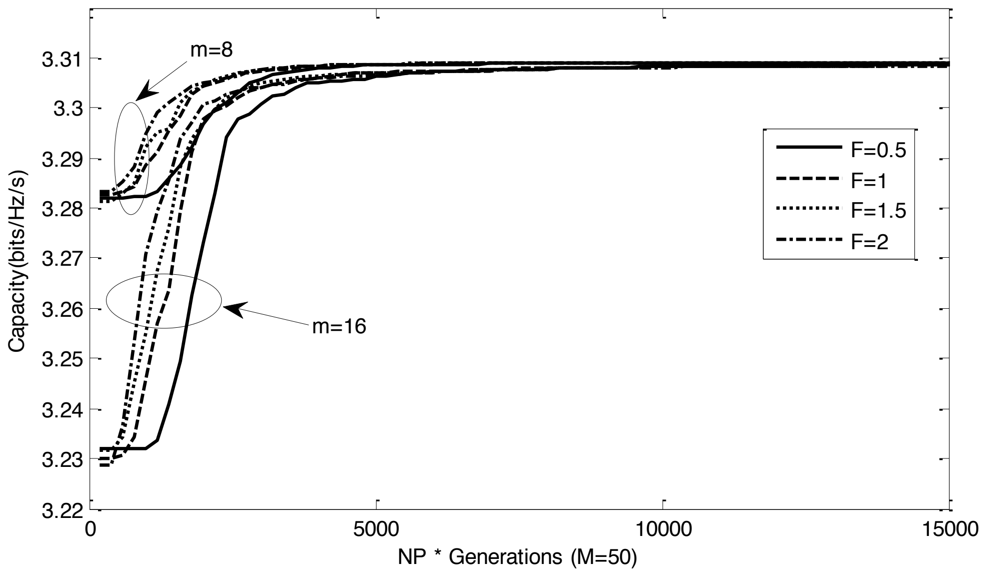


Figure 5.18. The secrecy capacity vs generations of Case 5-5 obtained by DE $m = 8, 16,\ \text{SNR} = 30\text{ dB}\ (NP = 200)$.

5.3 Rank-Adaptive Monte Carlo

To improve the efficiency of Monte Carlo, we developed a novel algorithm to solve the problem. Based on the discussion in [30], the difference between the channel matrices can be represented as

$$\mathbf{W}_1 - \mathbf{W}_2 = \mathbf{W}_+ + \mathbf{W}_- \qquad (5.19)$$

where $\mathbf{W}_{+(-)}$ collects positive (negative and zero) eigenmodes of $\mathbf{W}_1 - \mathbf{W}_2$. $\mathbf{W}_+$ represents the directions where the legitimate channel is stronger than the eavesdropper's channel. It has been proved that [30]

$$\text{rank}(\mathbf{R}^*) \leq \text{rank}(\mathbf{W}_+) \leq m \qquad (5.20)$$

In this part, we improve Monte Carlo optimization by selecting the optimal rank of the transmit covariance matrix. In the regular Monte Carlo optimization, we generate the covariance matrix randomly without any condition and most of the generated covariance matrices are full-rank (see Table 5.3). As we have mentioned, in many cases, the optimal covariance matrix is not full-rank. For example, if $\mathbf{W}_1 - \mathbf{W}_2$ has only one positive eigenvalue, $\mathbf{R}^*$ is a rank-one matrix regardless of the number of transmit antennas. Based on our previous observation, the probability of the randomly generated matrix approximately being of rank-one is almost zero especially when m is large (see Table 5.4). In some cases, the largest eigenvalue of generated matrix may approximately equal to P_T but cannot be exact P_T. Thus we generate the random transmit covariance matrix as follows

$$\mathbf{A}_{m\times r} = randn(m,r) \rightarrow \mathbf{R} = P_T \frac{\mathbf{A}_{m\times r}^{+}\mathbf{A}_{m\times r}}{\mathrm{Tr}(\mathbf{A}_{m\times r}^{+}\mathbf{A}_{m\times r})} \rightarrow \mathbf{R}\text{ has rank } r \qquad (5.21)$$

where $\mathbf{A}_{m\times r}$ is a $m \times r$ matrix of rank-r, and r = 1,2……m. The randomly generated $\mathbf{R}$ has rank-r so that the probability of $\mathbf{R}$ having the same rank as the optimal covariance matrix $\mathbf{R}^*$ is higher than that probability of the covariance matrix generated by (5.2) in regular Monte Carlo, if we consider all possible r. Figure 5.19 shows the flow chart of rank-adaptive Monte Carlo (RAMC) optimization.

Next, we will use three different algorithms which are regular MC, RAMC and Differential Evolution, to solve the optimization problem for secrecy capacity. We will also compare which method is the most efficient way to get the numerical results.

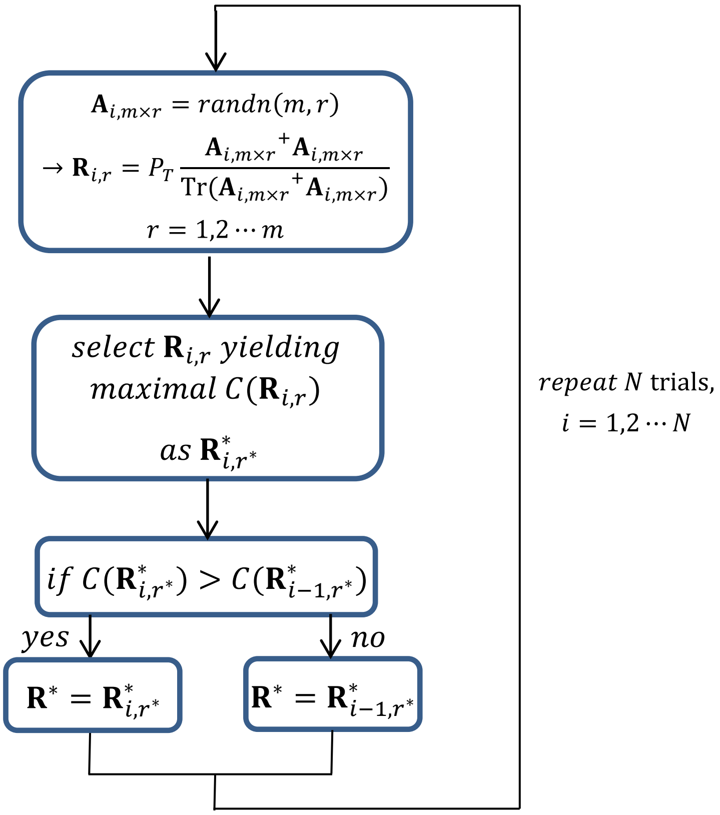

$\mathbf{A}_{i,m\times r} = randn(m,r)$
$\rightarrow \mathbf{R}_{i,r} = P_T \frac{\mathbf{A}_{i,m\times r}{}^{+}\mathbf{A}_{i,m\times r}}{\mathrm{Tr}(\mathbf{A}_{i,m\times r}{}^{+}\mathbf{A}_{i,m\times r})}$
$r = 1,2 \cdots m$
$select\ \mathbf{R}_{i,r}\ yielding\ maximal\ C(\mathbf{R}_{i,r})\ as\ \mathbf{R}^*_{i,r^*}$
$if\ C(\mathbf{R}^*_{i,r^*}) > C(\mathbf{R}^*_{i-1,r^*})$
yes
no
$\mathbf{R}^* = \mathbf{R}^*_{i,r^*}$
$\mathbf{R}^* = \mathbf{R}^*_{i-1,r^*}$
$repeat\ N$ trials,
$i = 1,2 \cdots N$

Figure 5.19. The flow chart of RAMC.

Notations:

DE: The secrecy capacity obtained by DE.

RAMC: The secrecy capacity obtained by rank-adaptive Monte Carlo .

MC: The secrecy capacity obtained by regular Monte Carlo.

C_s: The secrecy capacity returned by the known solution.

We still use the channel matrices in Case 5-5 where $\mathbf{W}_1 = \frac{1}{m}\mathbf{E}_{m\times m}$ and $\mathbf{W}_2 = 0.1\mathbf{W}_1$. In this case, it has been known that the beamforming is the optimal solution and $\mathbf{R}^* = P_T\boldsymbol{v}_{\max}\boldsymbol{v}_{\max}^{+} = P_T\mathbf{E}/m$. The secrecy capacity $C_s \approx 3.1988$ when $\mathrm{SNR} = 20\ \mathrm{dB}$ and $C_s \approx 0.0129$ when $\mathrm{SNR} = -20\ \mathrm{dB}$. Note that we set *NP* = 200, *CR* = 0.9 and *F* = 1.5. The values on the x axis of Figures $5.20 - 5.23$ stand for the total number of random covariance matrices generated and tested for each method with different m. In each trial of regular MC, there is only one covariance matrix generated based on (5.2); in each trial of rank-adaptive MC, there are $m\,(r = 1,2,\cdots,m)$ covariance matrices generated based on (5.21); in each trial of DE, there are *NP* covariance matrices modified and tested. This is the reason that the curves for different methods in Figures $5.20 - 5.23$ start from different points. The results shown in Figures $5.20 - 5.23$ are the average values over $M = 50$ iterations.

It can be observed that when the number of transmit antennas is low ($m = 2$ and 4), the speed of convergence is fast for all three methods. However, as the number of antennas increases, the difference in the speed of convergence becomes apparent, especially when *m*=16. We can conclude that it is difficult for regular Monte Carlo to find the approximate optimal covariance matrix and secrecy capacity. Comparing to regular MC for this case, rank-adaptive MC and DE both improve the accuracy of result significantly. DE converges to the approximate solution faster than rank-adaptive MC (see Table 5.16). Considering that the programming for RAMC optimization is simpler than DE algorithm, it can be concluded that both methods are acceptable when obtaining a numerical solution for the optimization problem for secrecy capacity of Gaussian MIMO wiretap channel.

Table 5.16. Processing time for generating Figures 5.22 and 5.23 (ℕ of G denotes the number of genereations for DE).

	DE ($NP = 200$, ℕ of G $= 1000$)	RAMC ($N = 10^6$)
processing time [s]	5897	18502

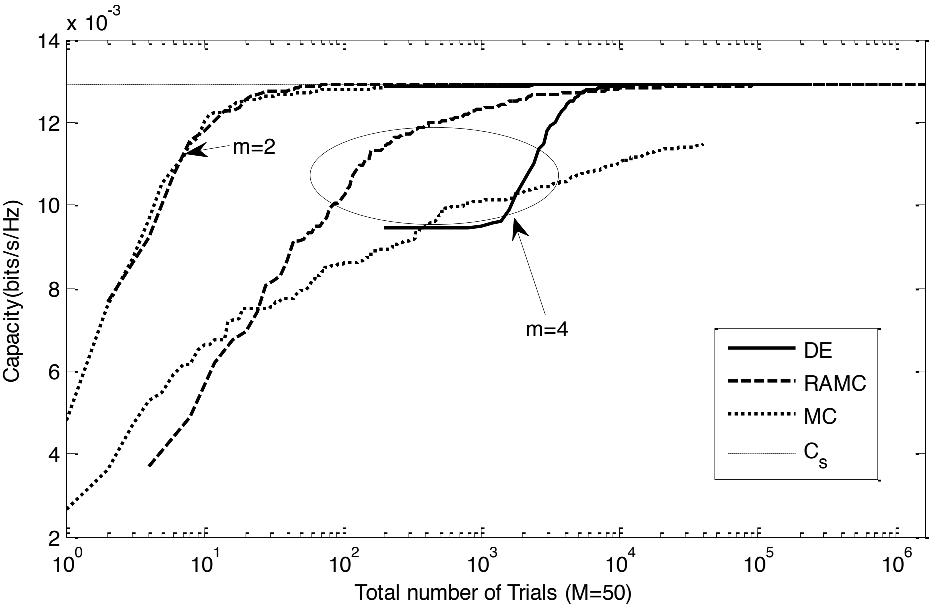


Figure 5.20. The secrecy capacity vs Trials of Case 5-5 obtained by DE, MC and RAMC, $\mathrm{SNR} = -20$ dB, $m = 2, 4$.

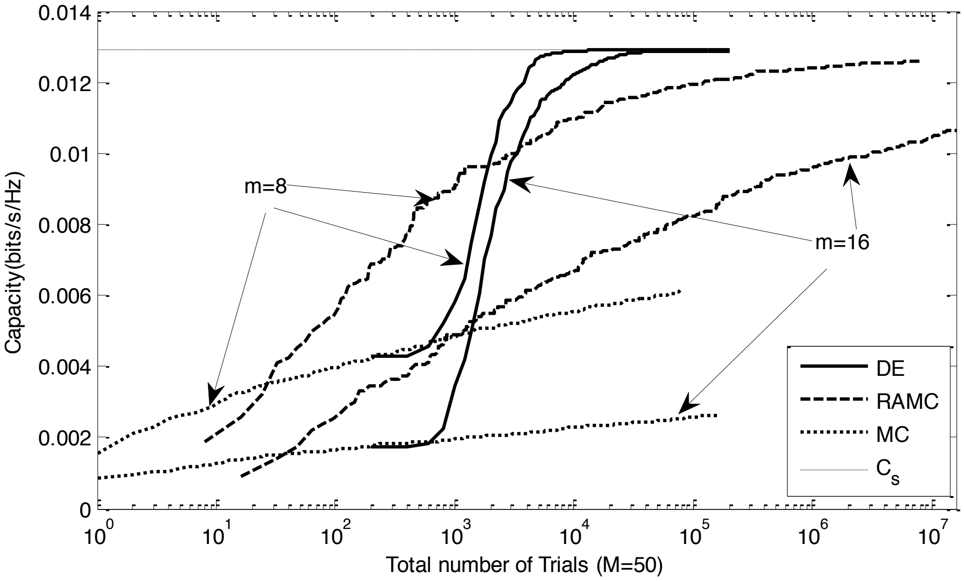


Figure 5.21. The secrecy capacity vs Trials of Case 5-5 obtained by DE, MC and RAMC, $\mathrm{SNR} = -20$ dB, $m = 8, 16$.

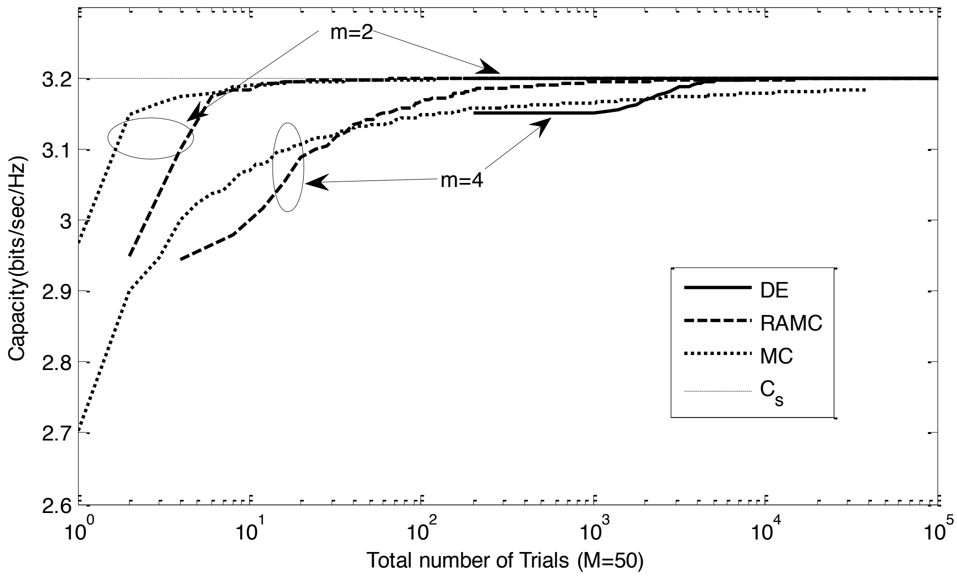


Figure 5.22. The secrecy capacity vs Trials of Case 5-5 obtained by DE, MC and RAMC, $\mathrm{SNR} = 20\,\mathrm{dB},\ m = 2, 4.$

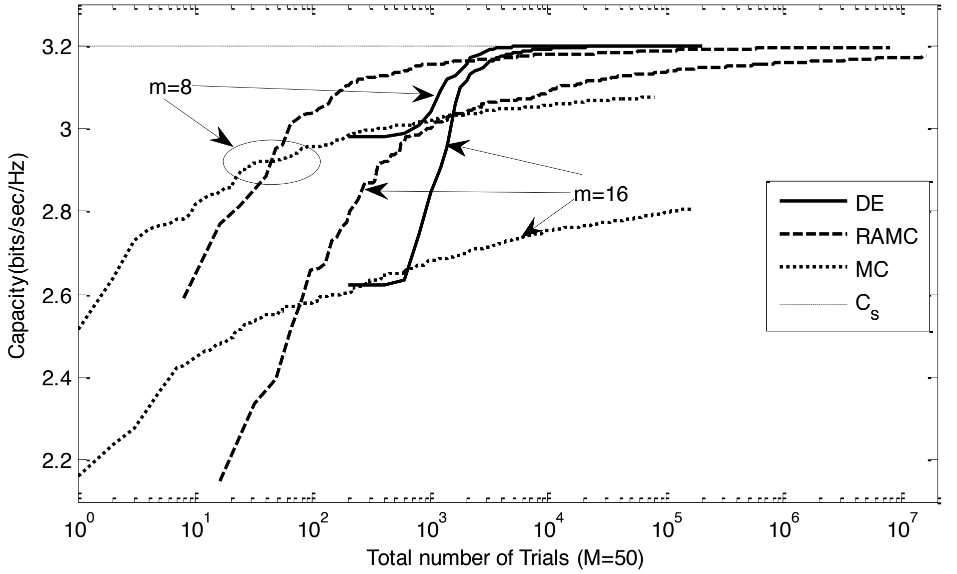


Figure 5.23. The secrecy capacity vs Trials of Case 5-5 obtained by DE, MC and RAMC, $\mathrm{SNR} = 20\,\mathrm{dB},\ m = 8, 16.$

5.4 Summary

In this chapter, we discussed the Monte Carlo optimization for achieving the numerical results of the optimization problem for the secrecy capacity of a given Gaussian MIMO wiretap channel. We found that Monte Carlo is able to approximately obtain the secrecy capacity for the cases where m is small. When m is large, it is difficult for Monte Carlo to converge to the accurate results especially when $\mathbf{R}^*$ has a low rank. The value of SNR does not affect the convergence. We also discussed the approximation of the optimization problem for the MIMO wiretap channel with weak eavesdropper which can be processed by CVX. By comparing the results obtained by Monte Carlo with results obtained by CVX, we found that CVX is able to return relatively accurate results when SNR is low.

To obtian the numerical results of the optimization problem without approximation, we discussed Differential Evolution algorithm and rank-adaptive Monte Carlo. They can both improve the convergence such that the secrecy capacity and the optimal covariance matrix for general channels can be approximately obtained. Comparing the results returned by these two algorithms, we can observe that the convergence of Differential Evolution algorithm is faster than that of rank-adaptive MC. Considering the processing time and the complexity of algorithms, both methods can be used to obtain the numerical results of the optimization problem properly.

6. Weak Eavesdropper Case

6.1 An Approximation for Weak Eavesdropper

As shown before, the disciplined CVX does not process secrecy rate function $C_s(\mathbf{R})$ of the Gaussian MIMO wiretap channel as objective since it is not always concave. In Chapter 5, we considered the scenario where $\mathbf{W}_2\mathbf{R} \ll \mathbf{I}$ such that an approximation for weak eavesdropper can be implemented.

In such scenario, (3.9) which is

$$C_s = \max_{\mathbf{R}} C_s(\mathbf{R}) = \max_{\mathbf{R}}\{\ln \frac{|\mathbf{I} + \mathbf{W}_1\mathbf{R}|}{|\mathbf{I} + \mathbf{W}_2\mathbf{R}|}\} \text{ ; s.t. } \mathbf{R} \geq \mathbf{0}; \mathrm{Tr}(\mathbf{R}) \leq P_T \qquad (6.1)$$

can be approximated by C_a as

$$C_s \approx C_a = \max_{\mathbf{R}} C_a(\mathbf{R}) = \max_{\mathbf{R}}\{\ln|\mathbf{I} + \mathbf{W}_1\mathbf{R}| - \mathrm{Tr}(\mathbf{W}_2\mathbf{R})\}; \qquad (6.2)$$

$$\text{s.t.} \, \mathrm{Tr}(\mathbf{R}) \leq P_T$$

when $\mathbf{W}_2\mathbf{R} \ll \mathbf{I}$, i.e. $\lambda_i(\mathbf{W}_2\mathbf{R}) \ll 1$, where $\lambda_i(\mathbf{A})$ denotes the i^{th} eigenvalue of $\mathbf{A}$.

Scaling Normalized $\mathbf{R}^*$ Returned by CVX:

Based on (6.2) solved by CVX, we will compare its performance with Monte Carlo. Note that the second term of C_a in (6.2) would become the dominant part when the power used is greater than a certain value (saturation point). This means that after that point, P_T cannot be totally consumed, otherwise the secrecy capacity would decrease with the increasing of SNR (noise power is fixed). This is also the reason that the approximation for weak eavesdropper is not proper for high SNR regime. To improve this situation, we normalized the optimal covariance matrix returned by CVX and scaled it by P_T, and then substituted the modified covariance matrix into the accurate secrecy rate $C_s(\mathbf{R})$ to verify if it is able to improve the accuracy. To gain some insights into this problem, we consider below a number of special cases.

Notations:

MC: Secrecy Capacity obtained by Monte Carlo (without approximation).

CVX: Approximated Secrecy Capacity C_a in (6.2) obtained by CVX.

CVX-R: Secrecy Capacity obtained by substituting $\mathbf{R}_{\text{cvx}}$ into $C_s(\mathbf{R})$, where $\mathbf{R}_{\text{cvx}}$ is returned by CVX using (6.2) as objective.

CVX-N: Secrecy Capacity obatined by substituting $\mathbf{R} = P_T \cdot \mathbf{R}_{\text{cvx}}/\text{Tr}(\mathbf{R}_{\text{cvx}})$ into $C_s(\mathbf{R})$.

$\mathbf{E}_m$: $m \times m$ all ones matrix.

$\boldsymbol{v}_i(\mathbf{A})$: i^{th} eigenvector of $\mathbf{A}$.

Case 6-1: $\mathbf{W}_1 = \frac{1}{m}\mathbf{E}_m \;\; \mathbf{W}_2 = 0.1\mathbf{W}_1,\; P_T = m \cdot \text{SNR}$

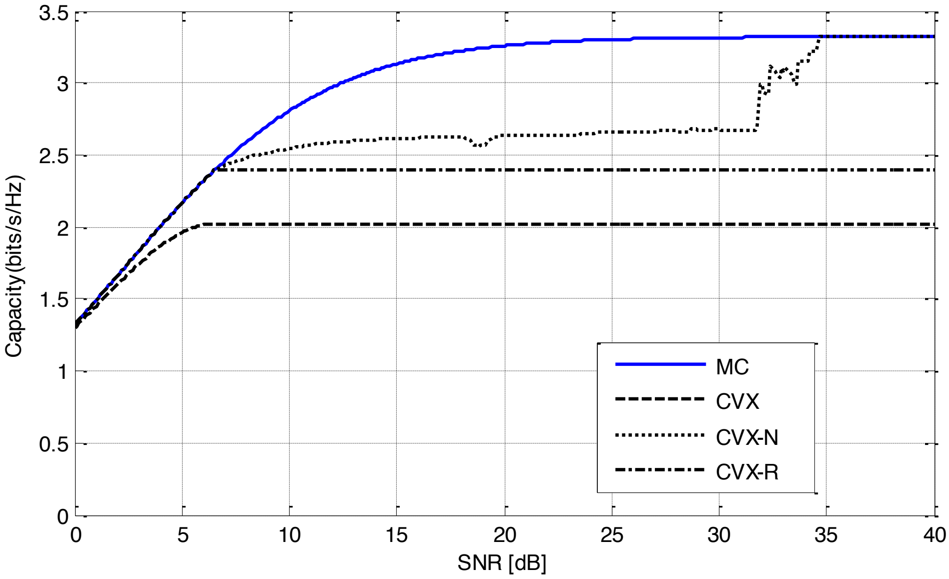


Figure 6.1. Secrecy capacity of Case 6-1 obtained by different methods vs SNR, CVX precision $= 10^{-4}$.

Observing Figures 6.1 and 6.2, even if the CVX-N curve is oscillating, it still has a better performance than using $\mathbf{R}_{\text{cvx}}$ (CVX-R) directly. In Figure 6.2, the CVX precision variable is set as 10^{-16}, which means that the solver of CVX continues as long as it reaches a lower tolerance level [63]. Comparing Figures 6.1 and 6.2, enhancement of CVX precision variable cannot eliminate the oscillation.

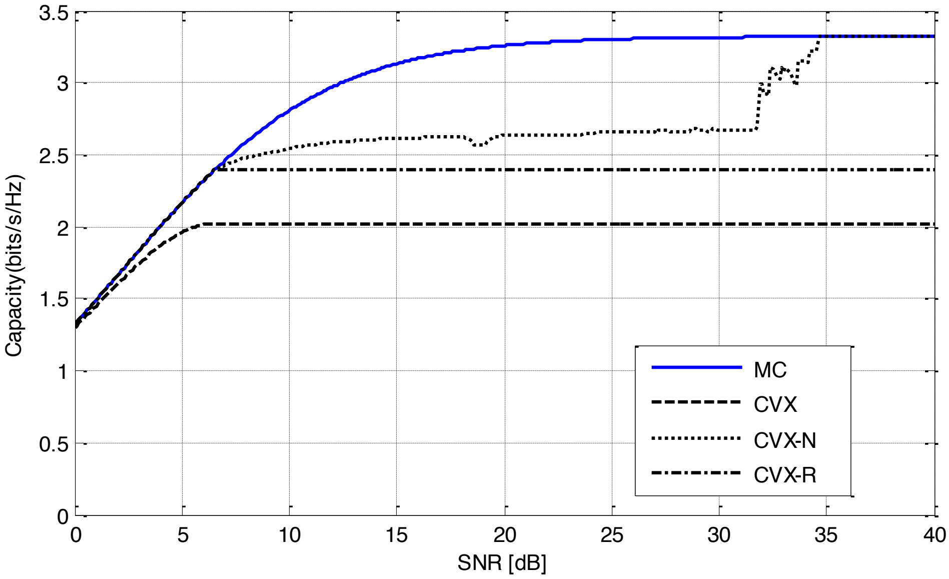


Figure 6.2. Secrecy capacity of Case 6-1 obtained by different methods, $\mathrm{CVX\ precision} = 10^{-16}$.

Adjustment of Approximation for Weak Eavesdropper:

We will use the channel matrices of Case 6-1. For the purpose of convenience, we set $m = 2$. The properties of this channel model are given in Table 6.1. Since $\mathbf{W}_1$ and $\mathbf{W}_2$ both have the same eigenvectors, based on the previous discussion, $r_+(\mathbf{W}_1 - \mathbf{W}_2) = 1 \rightarrow r_+(\mathbf{R}^*) = 1$, $\mathbf{R}^* = P_T \cdot \mathbf{E}/2$, where $r_+(\mathbf{A})$ is the number of positive eigenvalues of $\mathbf{A}$ [30].

Table 6.1. The channel model of Case 6-1 (m = 2) ($\boldsymbol{v}$ denotes the eigenvectors of given matrix; λ denotes the eigenvalues of given matrix).

P_T=2 · SNR	$\mathbf{W}_1 = \frac{1}{2}\mathbf{E}_2$		$\mathbf{W}_2 = \frac{0.1}{2}\mathbf{E}_2$	
$\boldsymbol{v}$	$\begin{bmatrix}0.7071\\0.7071\end{bmatrix}$	$\begin{bmatrix}-0.7071\\0.7071\end{bmatrix}$	$\begin{bmatrix}0.7071\\0.7071\end{bmatrix}$	$\begin{bmatrix}-0.7071\\0.7071\end{bmatrix}$
λ	1	0	0.1	0

Note that the curves of 'CVX-R' and 'MC' coincide at low SNR regime and separate after SNR ≈ 6 dB. In this case, the saturation point of the approximate secrecy capacity in (6.2) is given as follows (in [bits/s/Hz])

$$C_a = \max_{\mathrm{Tr}(\mathbf{R}) \le P_T} \{\log_2|\mathbf{I} + \mathbf{W}_1\mathbf{R}| - \mathrm{Tr}(\mathbf{W}_2\mathbf{R})/\ln 2\} \tag{6.3}$$

$$= \ln(1 + \lambda_1(\mathbf{W}_1\mathbf{R}))/\ln 2 - \lambda_1(\mathbf{W}_2\mathbf{R})/\ln 2$$

$$= [\ln(1 + P_T) - 0.1 \times P_T]/\ln 2$$

$$\frac{\mathrm{d}C_a}{\mathrm{d}P_T} = \frac{1}{(1 + P_T)} - 0.1 = 0$$

i.e. $P_T = 9$, corresponding to $\mathrm{SNR} = 6.53$ dB (from $P_T = 2\,\mathrm{SNR}$). Hence, C_a will not increase once $\mathrm{SNR} > 6.53$ dB. After that point, the power transmitted to the nonzero direction ($\boldsymbol{v}_1(\mathbf{W}_1) = \begin{bmatrix} 0.7071 \\ 0.7071 \end{bmatrix}$) is always 9 (the threshold power of C_a in Case 6-1) such that the upper bound of C_a in this case is given by

$$C_{a(\max)} = \log_2(1 + 9) - 0.1 \times 9/\ln 2 = 2.02\ \mathrm{bits/s/Hz} \qquad (6.4)$$

where $C_{a(\max)}$ denotes the upper bound of C_a in Case 6-1.

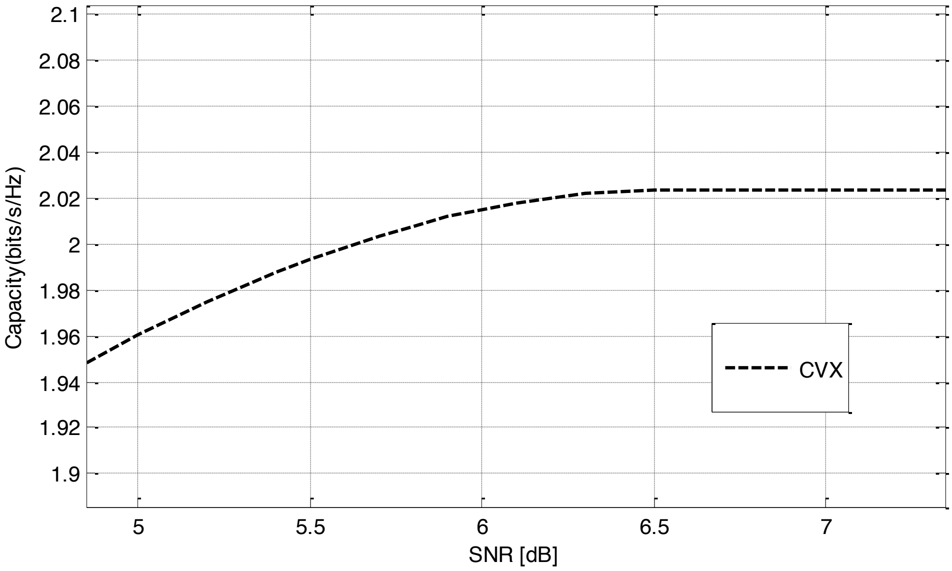


Figure 6.3. Saturation point and upper bound of approximated secrecy capacity of Case 6-1.

After $\mathrm{SNR} = 6.53$ dB, the optimal covariance matrix $\mathbf{R}_{\mathrm{cvx}}$ returned by CVX has the following properties:

Table 6.2. Eigenvectors and eigenvalues of $\mathbf{R}_{\text{cvx}}$ (SNR $\geq$ 6.53 dB).

$\boldsymbol{v}(\mathbf{R}_{\text{cvx}})$	$\boldsymbol{v}_1(\mathbf{R}_{\text{cvx}}) = \begin{bmatrix} 0.7071 \\ 0.7071 \end{bmatrix}$	$\boldsymbol{v}_2(\mathbf{R}_{\text{cvx}}) = \begin{bmatrix} -0.7071 \\ 0.7071 \end{bmatrix}$
$\lambda(\mathbf{R}_{\text{cvx}})$	$\lambda_1(\mathbf{R}_{\text{cvx}}) = 9$	$\lambda_2(\mathbf{R}_{\text{cvx}}) \in [0, P_T - 9]$

We can then observe that the power transmitted to $\boldsymbol{v}_1(\mathbf{R}_{\text{cvx}})$ is fixed at 9 while the power transmitted to $\boldsymbol{v}_2(\mathbf{R}_{\text{cvx}})$ jumps between $[0, P_T - 9]$ with the variety of SNR. The relationships of SNR with $\lambda_1(\mathbf{R}_{\text{cvx}})$ and $\lambda_2(\mathbf{R}_{\text{cvx}})$ are indicated in Figures $6.4 - 6.6$, thus proving our conclusion above.

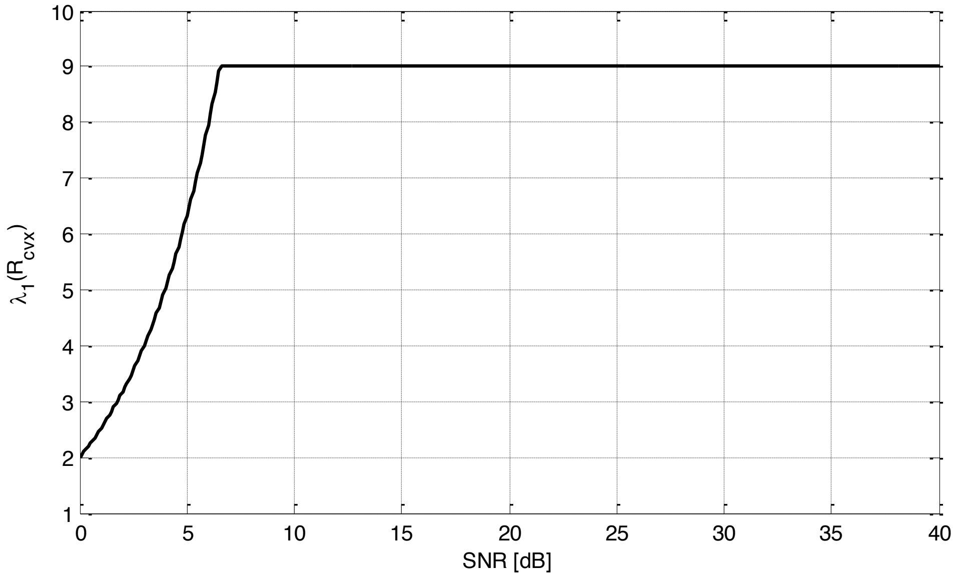


Figure 6.4. $\lambda_1(\mathbf{R}_{\text{cvx}})$ vs SNR of Case 6-1 $(\text{Tr}(\mathbf{R}) \leq 2\ \text{SNR})$.

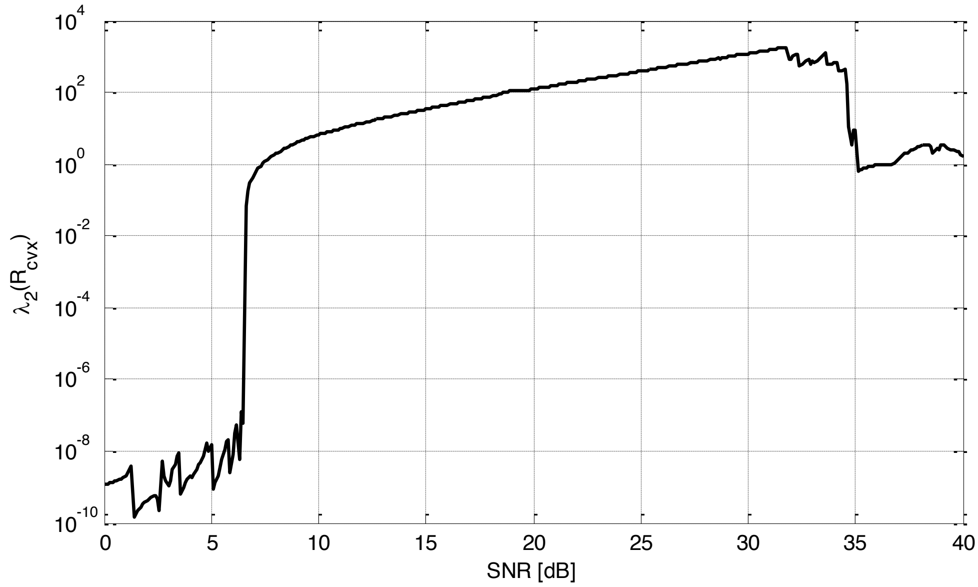


Figure 6.5. $\lambda_2(\mathbf{R}_{\text{cvx}})$ vs SNR of Case 6-1 $(\text{Tr}(\mathbf{R}) \leq 2\text{ SNR})$.

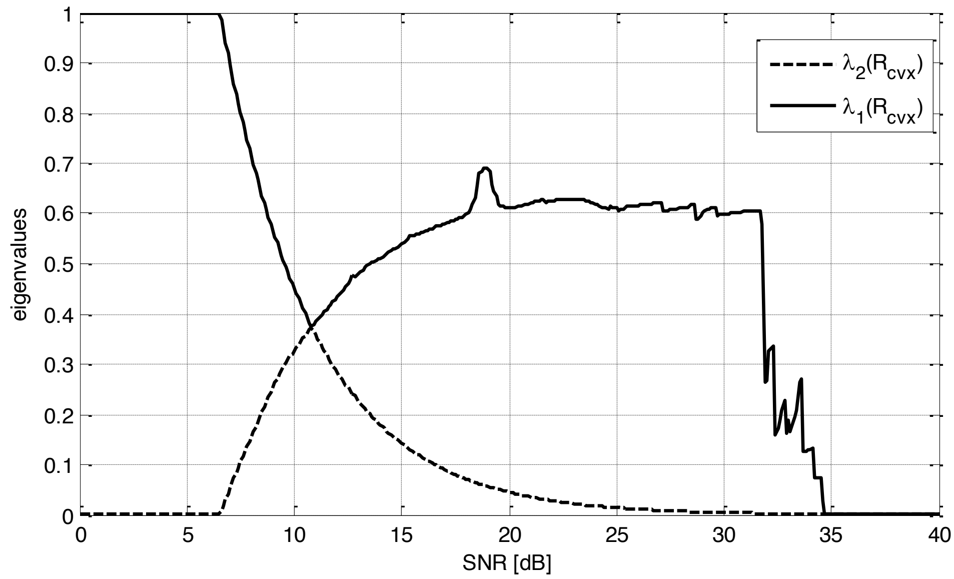


Figure 6.6. Eigenvalues of $\mathbf{R}_{\text{cvx}}$ normalized by P_T $(P_T = 2\text{ SNR})$.

Therefore, if we substitute modified $\mathbf{R}_{\text{cvx}}$ as $\mathbf{R} = P_T \cdot \mathbf{R}_{\text{cvx}}/\text{Tr}(\mathbf{R}_{\text{cvx}})$ into $C_s(\mathbf{R})$ given in (6.1), the secrecy capacity is given as

$$\begin{aligned} C_s &= \ln|\mathbf{I} + \mathbf{W}_1\mathbf{R}| - \ln|\mathbf{I} + \mathbf{W}_2\mathbf{R}| \qquad (6.5) \\ &= \ln(1 + \lambda_1(\mathbf{W}_1\mathbf{R})) - \ln(1 + \lambda_1(\mathbf{W}_2\mathbf{R})) \\ &= \ln(1 + 9P_T/\text{Tr}(\mathbf{R}_{\text{cvx}})) - \ln(1 + 0.9P_T/\text{Tr}(\mathbf{R}_{\text{cvx}})) \end{aligned}$$

where $\lambda_1(\mathbf{W}_1\mathbf{R}) = \lambda_1(\mathbf{R})$ and $\lambda_1(\mathbf{W}_2\mathbf{R}) = 0.1\,\lambda_1(\mathbf{R})$; $\lambda_2(\mathbf{W}_1\mathbf{R}) = \lambda_2(\mathbf{W}_2\mathbf{R}) = 0$.

Note that (6.5) is not a monotonically increasing function of P_T (SNR) and $\mathrm{P_T}/\mathrm{Tr}(\mathbf{R}_{\mathrm{cvx}})$ is not a constant. These two factors contribute to the oscillation of curve CVX-N shown in Figure 6.1. Figure 6.6 shows $\lambda_1(\mathbf{R}_{\mathrm{cvx}})/P_T$ and $\lambda_2(\mathbf{R}_{\mathrm{cvx}})/P_T$.

An Improvement:

Observing (6.5), for a given SNR (P_T), C_s is affected by $\mathrm{Tr}(\mathbf{R}_{\mathrm{cvx}})$ only which equals to $\lambda_1(\mathbf{R}_{\mathrm{cvx}}) + \lambda_2(\mathbf{R}_{\mathrm{cvx}})$. In Case 6-1, $\lambda_1(\mathbf{R}_{\mathrm{cvx}})$ is fixed at 9 when $\mathrm{SNR} \geq 6.53\ \mathrm{dB}$ while $\lambda_2(\mathbf{R}_{\mathrm{cvx}})$ can float in an interval (shown in Figure 6.2), so that the solutions of this case are non-unique and CVX selects only one of them. This is a possible explination of the oscillations appearing in Figures 6.1 and 6.2. Hence if we manually enforce $\lambda_2(\mathbf{R}_{\mathrm{cvx}})$ to zero, (6.5) tends to its maximum value, which coincides with the secrecy capacity obtained by Monte Carlo (Figure 6.7).

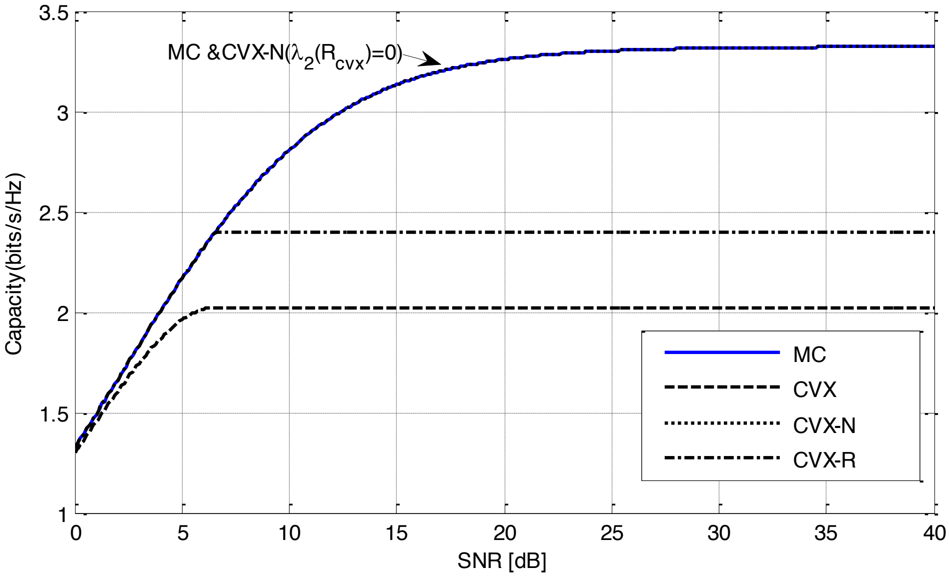


Figure 6.7. Improved result obtained by enforcing $\lambda_2(\mathbf{R}_{\mathrm{cvx}}) = 0$.

A Different CVX Constraint:

If we use "$\mathrm{Tr}(\mathbf{R}) \leq m$" as constraint of (6.2) instead of using "$\mathrm{Tr}(\mathbf{R}) \leq m$ SNR" , (6.2) becomes:

$$C_a = \max_{\mathbf{R}}\{\ln|\mathbf{I} + \gamma\mathbf{W}_1\mathbf{R}| - \mathrm{Tr}(\gamma\mathbf{W}_2\mathbf{R})\};\ \mathrm{s.t.}\,\mathbf{R} \geq \mathbf{0}, \mathrm{Tr}(\mathbf{R}) \leq m \qquad (6.6)$$

where γ denotes the SNR. Repeating the same simulations as the previous sections, we obtained the following results:

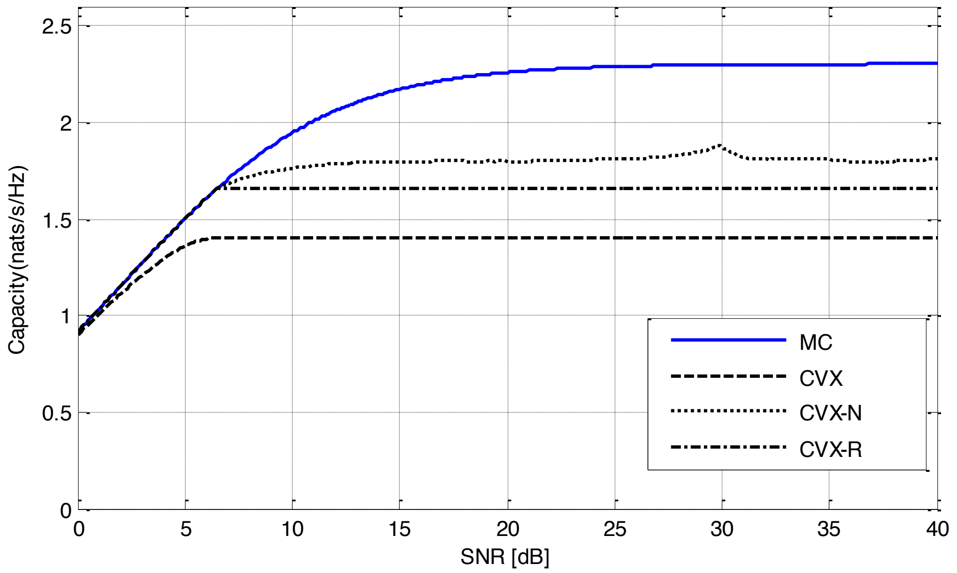


Figure 6.8. Secrecy Capacity of Case 6-1 obtained by different methods (constrained by $\mathrm{Tr}(\mathbf{R}) \leq m$).

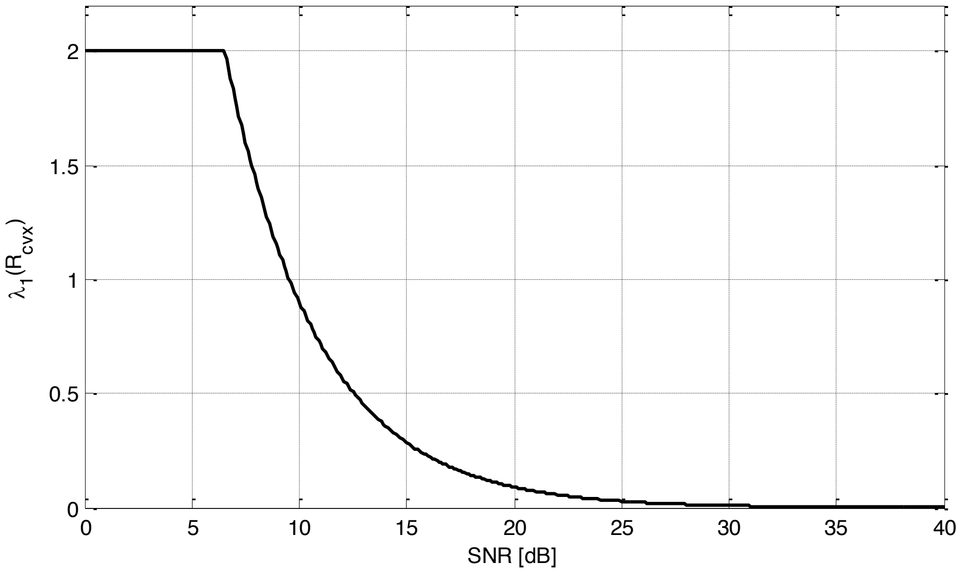


Figure 6.9. $\lambda_1(\mathbf{R}_{\text{cvx}})$ vs SNR of Case 6-1 (constrained by $\mathrm{Tr}(\mathbf{R}) \leq m$).

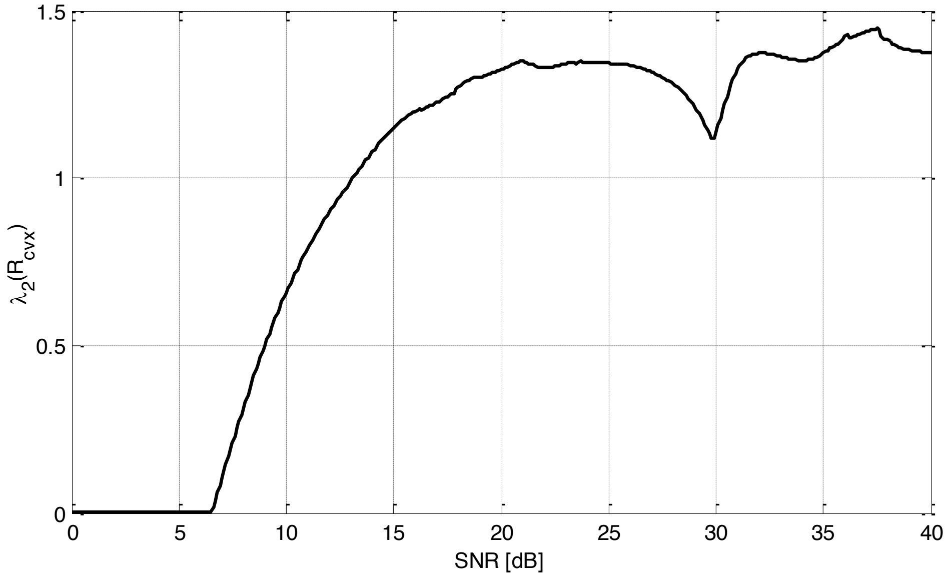


Figure 6.10. $\lambda_2(\mathbf{R}_{\text{cvx}})$ vs SNR of Case 6-1 (constrained by $\text{Tr}(\mathbf{R}) \leq m$).

Comparing Figure 6.2 with Figure 6.8, we can observe that the difference only appears between the secrecy capacity curves CVX-N ($\mathbf{R} = P_T \cdot \mathbf{R}_{\text{cvx}}/\text{Tr}(\mathbf{R}_{\text{cvx}})$). As we have concluded, the capacity is only affected by the power transmitted to the nonzero eigenmode ($\boldsymbol{v}_1(\mathbf{W}_1)$), i.e. the active power. Despite what the constraint may be, the active power is always given by

$$P_{\text{active}} = P_T \frac{\lambda_1(\mathbf{R}_{\text{cvx}})}{\lambda_1(\mathbf{R}_{\text{cvx}}) + \lambda_2(\mathbf{R}_{\text{cvx}})} \tag{6.7}$$

Comparing the P_{active} in these two cases, we can observe them in the following figures:

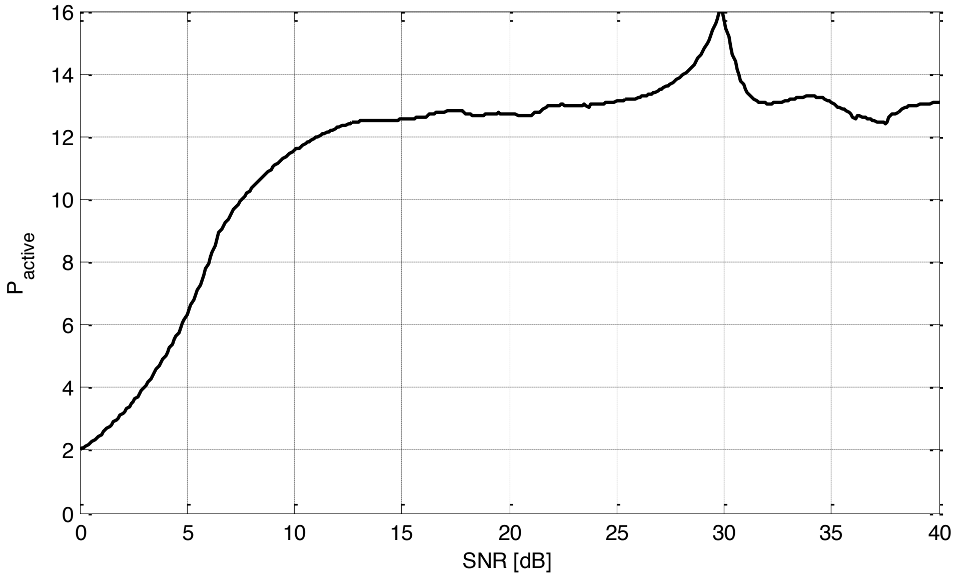


Figure 6.11. Active Power of Case 6-1 (constrained by $\mathrm{Tr}(\mathbf{R}) \leq m$).

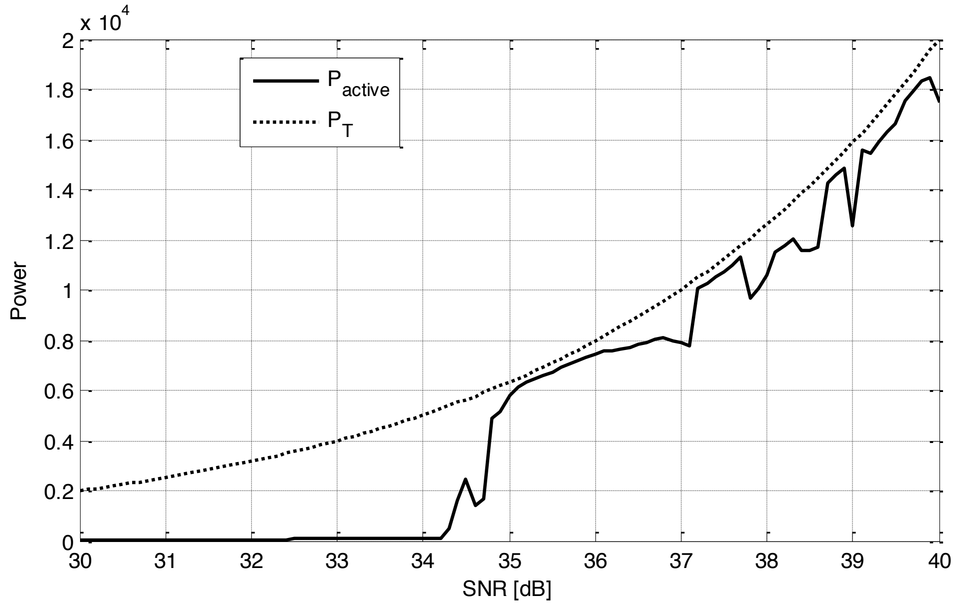


Figure 6.12. Active Power of Case 6-1 (constrained by $\mathrm{Tr}(\mathbf{R}) \leq m \cdot \mathrm{SNR}$, SNR ≥ 30 dB).

Based on Figures $6.11 - 6.13$, there is no significant difference between the active powers of these two cases at low SNR. Hence the secrecy capacities obtained by substituting $\mathbf{R} = P_T \cdot \mathbf{R}_{\mathrm{cvx}}/\mathrm{Tr}(\mathbf{R}_{\mathrm{cvx}})$ into $C_s(\mathbf{R})$ shown in Figures 6.1 and 6.8 are similar at low SNR. In Figure 6.12, however, the active power has an evident boost when SNR is greater than 35 dB. This power boosting contributes to the significant secrecy capacity (CVX-N) increase in Figure 6.1.

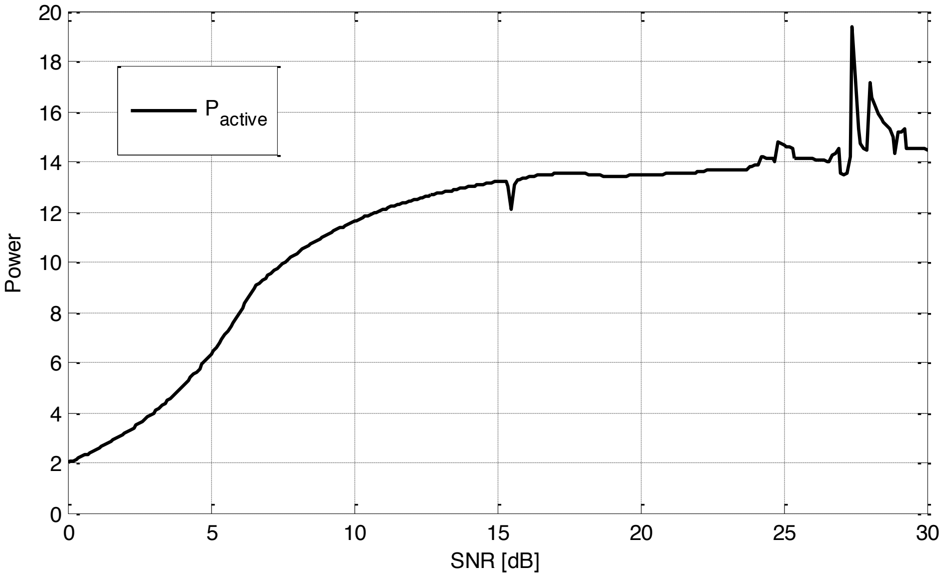


Figure 6.13. Active Power of Case 6-1 (constrained by $\mathrm{Tr}(\mathbf{R}) \leq m \cdot \mathrm{SNR}$, $\mathrm{SNR} \leq 30$ dB).

Full-Rank Channel Matrices:

In the last section, we discussed the case where $\mathbf{R}^*$ has rank-one. We found that when we try to modify $\mathbf{R}_{\mathrm{cvx}}$ to improve the result, the eigenvalue of $\mathbf{R}_{\mathrm{cvx}}$ corresponding to the zero eigenmode does not contribute to the secrecy capacity directly but instead affects the power allocation when we use $\mathbf{R} = P_T \cdot \mathbf{R}_{\mathrm{cvx}} / \mathrm{Tr}(\mathbf{R}_{\mathrm{cvx}})$, so that $C_s(\mathbf{R})$ is affected. In this section, we will discuss the case where $\mathbf{R}^*$ has full-rank.

Case 6-2:

Table 6.3. Channel matrices of Case 6-2 ($\boldsymbol{v}$ denotes the eigenvectors of given matrix; λ denotes the eigenvalues of given matrix).

	$\mathbf{W}_1=\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$		$\mathbf{W}_2=0.1\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$	
$\boldsymbol{v}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0.5257 \\ -0.8507 \end{bmatrix}$	$\begin{bmatrix} -0.8507 \\ -0.5257 \end{bmatrix}$
λ	1	2	0.0382	0.2618

We used the same methods as for Case 6-1 to compute the secrecy capacity of Case 6-2, and the results are given in Figure 6.14. Figure 6.14 shows that the secrecy capacity curve yielded by the modified $\mathbf{R}_{\mathrm{cvx}}$ is very close to the one obtained by

Monte Carlo but they are not identical. The eigenvalues of $\mathbf{R}_{\text{cvx}}$, modified $\mathbf{R}_{\text{cvx}}$ $[P_T\, \mathbf{R}_{\text{cvx}}/\text{Tr}(\mathbf{R}_{\text{cvx}})]$ and the covariance matrix returned by Monte Carlo are indicated in Figures 6.15 and 6.16. Unlike the situation in Case 6-1, both eigenvalues here may contribute to the secrecy capacity directly. Observing Figure 6.16, the eigenvalues curves shown are similar but not identical. The eigenvalues here represent power allocation directly and indicate the reason that the corresponding secrecy capacity curves are so close. At high SNR region, the transmitted power is large enough such that the slope of (6.1) is small, which shrinks the gap between the two curves ('CVX-N' and 'MC'). However, the gap still exists.

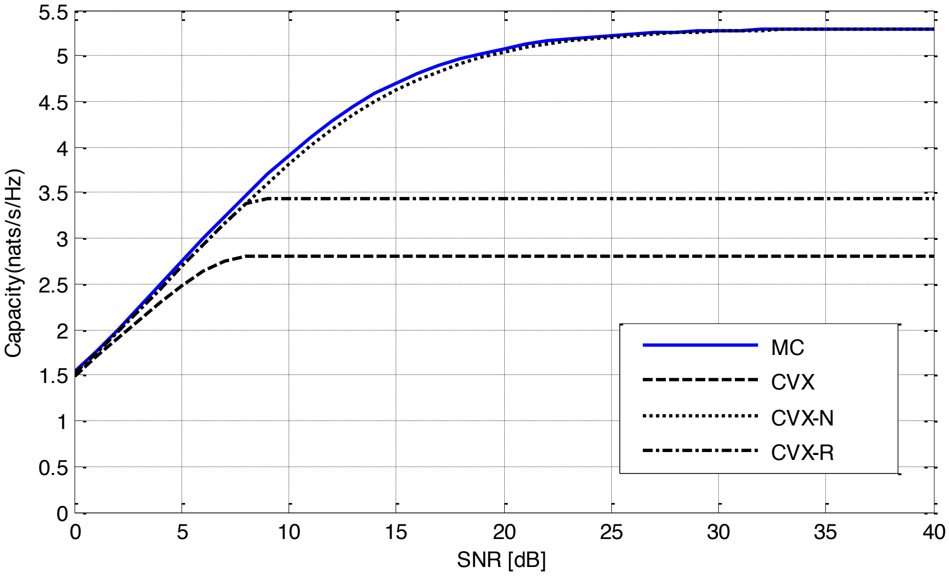


Figure 6.14 Secrecy Capacity of Case 6-2 returned by different methods (constrained by $\text{Tr}(\mathbf{R}) \leq m$ SNR)

There are two reasons contributing to this gap, the first is that the power allocation strategies are similar but not identical. This has been shown in Figure 6.16.

The second reason is that the eigenvectors of $\mathbf{R}_{\text{cvx}}$ are always the same as the eigenvectors of $\mathbf{W}_1$ regardless of what SNR is, i.e. the eigenvectors of $\mathbf{R}_{\text{cvx}}$ in Case 6-2 are always $\begin{bmatrix}0\\1\end{bmatrix}$ and $\begin{bmatrix}1\\0\end{bmatrix}$. To verify this observation, we generated a group of channel matrices randomly and substituted them into (6.2). We used CVX to compute

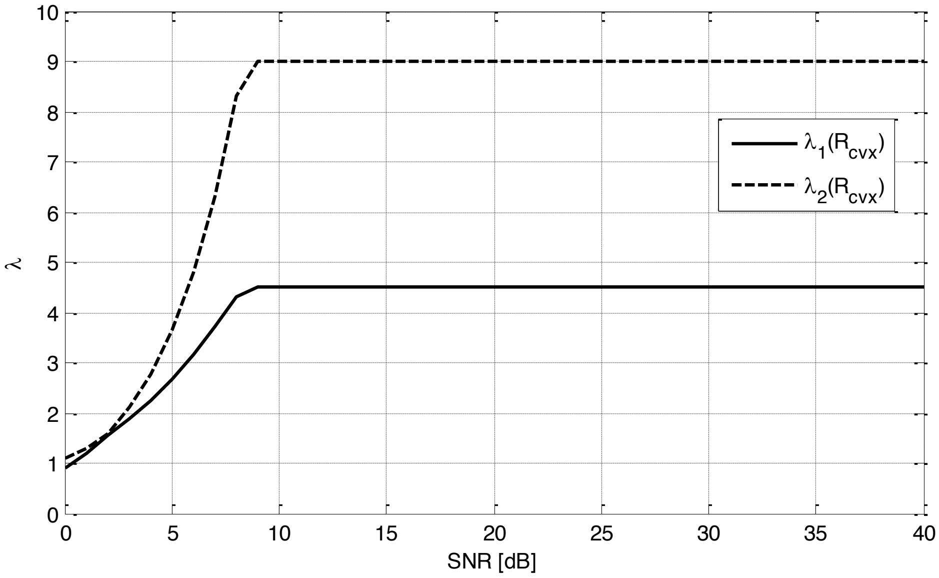


Figure 6.15. Eigenvalues of $\mathbf{R}_{\text{cvx}}$.

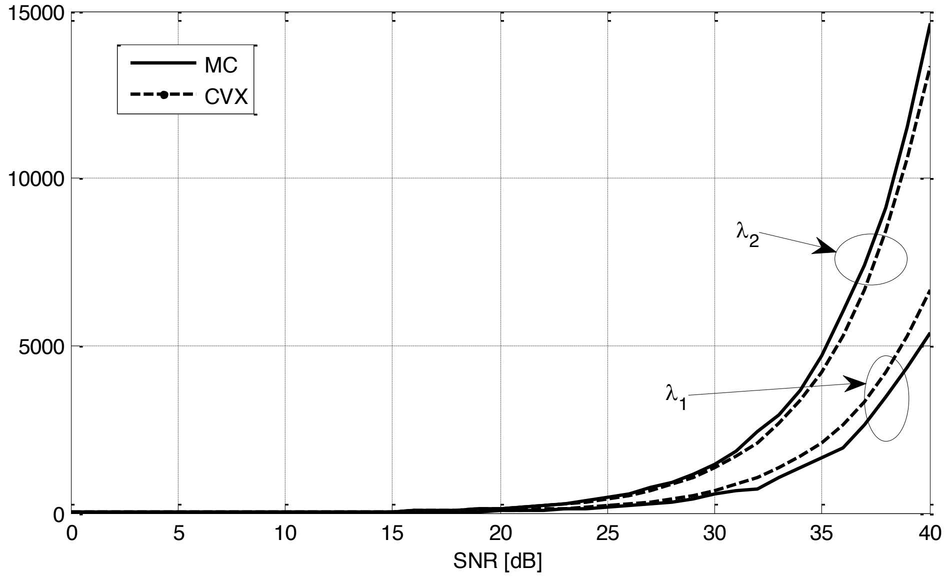


Figure 6.16. Eigenvalues of $P_T \cdot \mathbf{R}_{\text{cvx}}/\text{Tr}(\mathbf{R}_{\text{cvx}})$ and $\mathbf{R}_{\text{MC}}$.

the approximate secrecy capacity C_a in (6.2) of these random channels and then checked the eigenvectors of returned optimal covariance matrices. The results are given in Tables $6.4 - 6.5$ and show that the eigenvectors of $\mathbf{R}_{\text{cvx}}$ are always the same as the eigenvectors of $\mathbf{W}_1$.

This reveals that CVX may not able to solve the optimization problem (6.2) properly. A possible interpretation of this phenomenon is as follows: CVX maximizes the $\ln(\det)$ term first and minimizes the $\mathrm{Trace}(\cdot)$ term based on the result of the first step. Hence, unless $\mathbf{W}_1$ and $\mathbf{W}_2$ have the same eigenvectors, there is always an angle's deviation between the optimal eigen directions (eigenvectors of $\mathbf{R}^*$) and the eigen directions returned by CVX (eigenvectors of $\mathbf{R}_{\mathrm{cvx}}$). In the next section, we will discuss the analytical solution of (6.2) and compare it with results obtained by CVX.

Table 6.4. Eigenvectors of $\mathbf{W}_1$ and $\mathbf{R}_{\mathrm{cvx}}$ returned by CVX ($\mathrm{SNR} = -30$ dB).

$\mathbf{W}_1$	5.3915 2.9335 2.9335 4.1065		14.6271 14.0069 14.0069 16.8805		2.0008 2.1818 2.1818 4.2276		0.7674 0.7264 0.7264 1.1627	
$\mathbf{W}_2$	0.2896 0.5654 0.5654 1.8274		1.0371 0.1922 0.1922 0.0460		0.9653 1.9801 1.9801 4.1157		0.8756 1.2504 1.2504 1.9356	
$\boldsymbol{v}(\mathbf{W}_1)$	0.6269 −0.7791	0.7791 0.6269	−0.7349 0.6782	0.6782 0.7349	−0.8528 0.5222	0.5222 0.8528	−0.7945 0.6072	0.6072 0.7945
$\boldsymbol{v}(\mathbf{R}_{\mathrm{cvx}})$	0.6269 −0.7791	0.7791 0.6269	−0.7349 0.6782	0.6782 0.7349	−0.8528 0.5222	0.5222 0.8528	−0.7945 0.6072	0.6072 0.7945

Table 6.5. Eigenvectors of $\mathbf{W}_1$ and $\mathbf{R}_{\mathrm{cvx}}$ returned by CVX ($\mathrm{SNR} = 30$ dB).

$\mathbf{W}_1$	1.3445 0.7639 0.7639 0.5447		0.9664 1.9923 1.9923 6.7199		0.7162 0.8008 0.8008 1.0851		7.5031 2.1277 2.1277 0.7426	
$\mathbf{W}_2$	1.3643 0.3718 0.3718 1.3702		0.1995 0.9473 0.9473 6.9574		0.3322 0.5445 0.5445 0.9070		2.7584 1.5715 1.5715 1.1649	
$\boldsymbol{v}(\mathbf{W}_1)$	0.5178 −0.8555	0.8555 0.5178	−0.9545 0.2982	0.2982 0.9545	−0.7824 0.6227	0.6227 0.7824	0.2772 −0.9608	0.9608 0.2772
$\boldsymbol{v}(\mathbf{R}_{\mathrm{cvx}})$	0.5178 −0.8555	0.8555 0.5178	−0.9545 0.2982	0.2982 0.9545	−0.7824 0.6227	0.6227 0.7824	0.2772 −0.9608	0.9608 0.2772

6.2 An Analytical Solution for Weak Eavesdropper

In this section, we consider the cases where the eavesdroppers are weak. This condition can be represented as $\lambda_i(\mathbf{W}_2\mathbf{R}) \ll 1$. Under this condition, the analytical solution of the optimal covariance matrix has been obtained in [32]. As we have discussed in the last section, the secrecy capacity of Gaussian MIMO wiretap channel can be approximated as:

$$C_a = \max_{\mathbf{R}} C_a(\mathbf{R}) = \max_{\mathbf{R}}\{\ln|\mathbf{I} + \mathbf{W}_1\mathbf{R}\,| - \mathrm{Tr}(\mathbf{W}_2\mathbf{R})\};\ \mathrm{s.t.}\,\mathbf{R} \geq \mathbf{0}, \mathrm{Tr}(\mathbf{R}) \leq P_T \quad (6.8)$$

where

$$\ln|\mathbf{I}+\mathbf{W}_1\mathbf{R}| \approx \mathrm{Tr}(\mathbf{W}_2\mathbf{R}) \tag{6.9}$$

when $\lambda_i(\mathbf{W}_2\mathbf{R}) \ll 1$.

Based on the observation in [32], $C_a(\mathbf{R})$ is concave with respect to **R**, the optimal covariance is given by

$$\mathbf{R}^* = \mathbf{A}^{-\frac{1}{2}}\mathbf{U}\mathbf{\Lambda}\mathbf{U}^+\mathbf{A}^{-\frac{1}{2}} \tag{6.10}$$

where

$$\mathbf{A} = \lambda\mathbf{I} + \mathbf{W}_2; \tag{6.11}$$

the columns of unitary matrix **U** are the eigenvectors of

$$\widetilde{\mathbf{W}_1} = \mathbf{A}^{-1/2}\mathbf{W}_1\mathbf{A}^{-1/2} \tag{6.12}$$

Λ is the diagonal matrix and its entries are

$$\widetilde{\lambda_i}(1-\lambda_i^{-1}\widetilde{\mathbf{W}_1})_+ \tag{6.13}$$

where $(x)_+$=max$(x,0)$; $\lambda \geq 0$ is the Lagrange multiplier and can be found from the total power constraint:

$$\mathrm{Tr}(\mathbf{R}^*) = P_T \text{ , if } P_T \leq P_T^* \tag{6.14}$$

P_T^* is introduced here as the threshold power for the secrecy capacity to saturate. Intuitively, in some cases, the second term of (6.8) is smaller than the first term when P_T is small. If the total transmit power is greater than P_T^* , the second term will become the dominant term of the objective function in (6.8) and the secrecy capacity will decrease with increasing total transmit power. P_T^* provides the saturate point of the approximate secrecy capacity. Beyond that point, the secrecy capacity ceases to increase even if the total power continues to increase. If $P_T \geq P_T^*$, then $\lambda = 0$ [32]. To guarantee the accuracy of the approximation in (6.8), the condition $\lambda_i(\mathbf{W}_2\mathbf{R}) \ll 1$ must hold true. Thus the threshold power can be derived from this condition and has the form [32]

$$P_T^* = \mathrm{Tr}(\mathbf{W}_2^{-1}(\mathbf{I}-\mathbf{W}_2^{1/2}\mathbf{W}_1^{-1}\mathbf{W}_2^{1/2})_+ \tag{6.15}$$

where $(\mathbf{W})_+$ denotes the positive eigenmodes of Hermitian matrix **W**. If $\mathbf{W}_2$ is singular and $\mathcal{N}(\mathbf{W}_2) \notin \mathcal{N}(\mathbf{W}_1)$, then $P_T^* = \infty$ [30], where $\mathcal{N}(\mathbf{W})$ is the null space of matrix **W**.

In the above discussion, $\mathrm{Tr}(\mathbf{R}^*)$ is the function of λ that monotonically decreases with respect to λ. Based on this, we can use bisection to obtain $\mathbf{R}^*$ and the optimal λ.

Bisection:

We initialize the upper bound of λ, $\lambda_{\max} = m/P_T$ (since $\mathrm{Tr}(\mathbf{R}^*) \leq P_T$ [32]) and the lower bound of λ, $\lambda_{\min} = 0$. We then substitute $\lambda = (\lambda_{\max} + \lambda_{\min})/2$ into (6.11) and compute $\mathrm{Tr}(\mathbf{R}^*)$. If $\mathrm{Tr}(\mathbf{R}^*) > P_T$, we reset $\lambda_{\min} = \lambda$, otherwise we reset $\lambda_{\max} = \lambda$. Then we reset $\lambda = (\lambda_{\max} + \lambda_{\min})/2$ and substitute the new λ into (6.11) and compute $\mathrm{Tr}(\mathbf{R}^*)$ again. We repeat this process for a number of iterations until

$$|\mathrm{Tr}(\mathbf{R}^*) - P_T| < \varepsilon P_T \tag{6.16}$$

where $\varepsilon \ll 1$ is an acceptable accuracy. If $P_T \geq P_T^*$, λ is set to 0 directly. Figure 6.17 shows the flow chat of Bisection. Next, we validate this algorithm in several cases.

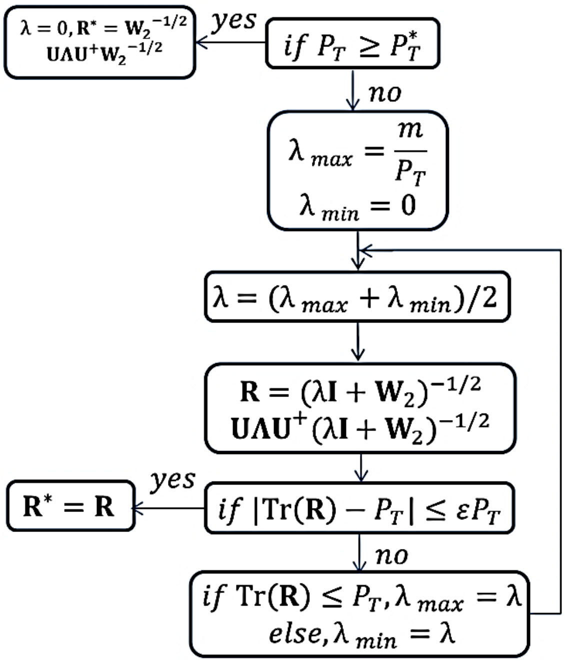

$\lambda = 0, \mathbf{R}^* = \mathbf{W}_2^{-1/2}$
$\mathbf{U\Lambda U}^+\mathbf{W}_2^{-1/2}$
yes
$if\ P_T \geq P_T^*$
no
$\lambda_{max} = \frac{m}{P_T}$
$\lambda_{min} = 0$
$\lambda = (\lambda_{max} + \lambda_{min})/2$
$\mathbf{R} = (\lambda\mathbf{I} + \mathbf{W}_2)^{-1/2}$
$\mathbf{U\Lambda U}^+(\lambda\mathbf{I} + \mathbf{W}_2)^{-1/2}$
yes
$\mathbf{R}^* = \mathbf{R}$
$if\ |\mathrm{Tr}(\mathbf{R}) - P_T| \leq \varepsilon P_T$
no
$if\ \mathrm{Tr}(\mathbf{R}) \leq P_T, \lambda_{max} = \lambda$
$else, \lambda_{min} = \lambda$

Figure 6.17. Flow chat of Bisection.

Notations

Analytical: The secrecy capacity obtained by the analytical solution (weak eavesdropper).

CVX: The secrecy capacity in (6.2) obtained by CVX.

MCa: The secrecy capacity in (6.2) obtained by Monte Carlo (approximated).

MC: The secrecy capacity in (6.1) obtained by Monte Carlo (exact).

Case 6-3: $\mathbf{W}_1 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$ $\mathbf{W}_2 = 0.1\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$

In this case,

$$P_T^* = \operatorname{Tr}(\mathbf{W}_2^{-1}(\mathbf{I} - \mathbf{W}_2^{\frac{1}{2}}\mathbf{W}_1^{-1}\mathbf{W}_2^{\frac{1}{2}})_+ = 12 = 10.79 \text{ dB}$$

We can observe in Figure 6.18 that the secrecy capacity obtained by the analytical solution and CVX both saturate at $\text{SNR} = 10.97$ dB. Since CVX cannot return the correct eigenvectors, the result returned by CVX is lower than the result returned by the analytical solution when $\text{SNR} \geq 5$ dB. By comparing this with the result obtained by Monte Carlo, we can conclude that the algorithm is correct and it is accurate if $\text{SNR} \leq 8$ dB in this case. Since in both CVX and our algorithm, the dual variable λ is utilized, we also compared the dual variables returned by these two methods. Figure 6.19 shows that the dual variables returned by the two methods do not coincide and the λ returned by the algorithm decreases to numerical zero when SNR is greater than P_T^*.

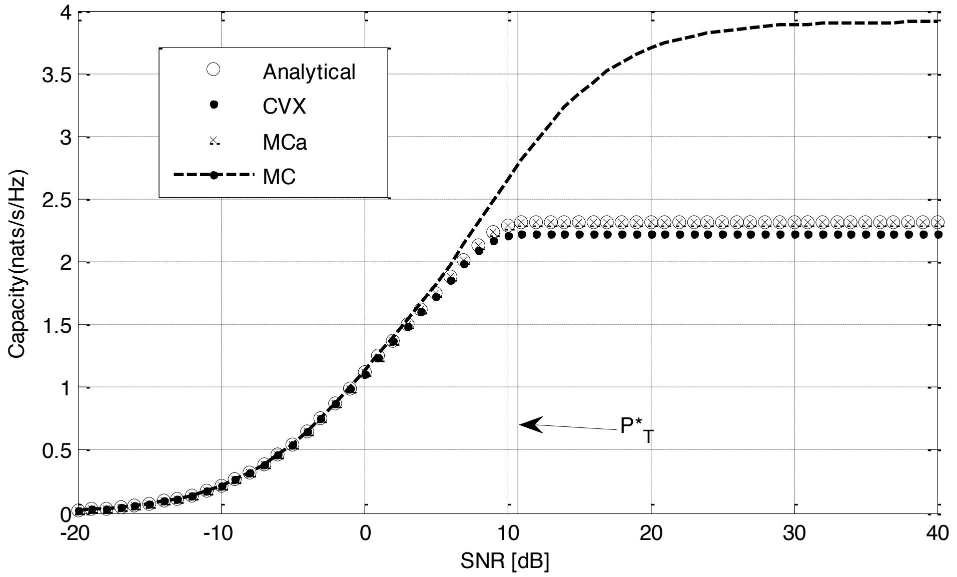


Figure 6.18. The secrecy capacity obtained by CVX, Monte Carlo, and the analytical solution (Case 6-3).

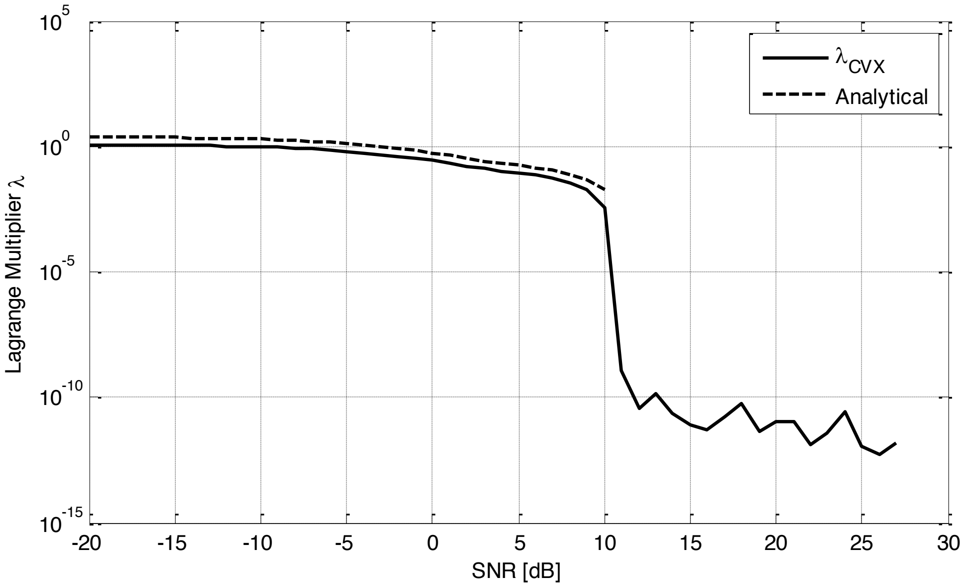


Figure 6.19. The dual variables returned by CVX and the analytical solution (Case 6-3).

We also investigated some other cases where $P_T^* = \infty$.

Case 6-4: $\mathbf{W}_1 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$ $\mathbf{W}_2 = \mathbf{E}_2$

In this case, $\mathbf{W}_2$ is a singular matrix (it has only one nonzero eigenvalue) and $\mathcal{N}(\mathbf{W}_2) \notin \mathcal{N}(\mathbf{W}_1)$, so $P_T^* = \infty$. We know that the optimal covariance matrix has rank-one since $r_+(\mathbf{W}_1 - \mathbf{W}_2) = 1$ [30] and that $\mathbf{W}_2$ has rank-one as well. Hence, at high SNR region, the optimal covariance matrix should be able to find the direction that is orthogonal to the eigenvector corresponding to the nonzero eigenvalue of $\mathbf{W}_2$. This means that all of the transmit power should be allocated to the optimal direction and there is no wasted power.

Figure 6.20 shows that the secrecy capacity obtained by CVX saturates at $\mathrm{SNR} = 8\,\mathrm{dB}$ while the results obtained by other algorithms all continue to rise with increasing SNR. Since the optimal covariance matrix $\mathbf{R}^*$ has rank-one and the nonzero eigenmode of $\mathbf{R}^*$ should be orthogonal to the nonzero eigenmode of $\mathbf{W}_2$ so that $\mathrm{Tr}(\mathbf{R}^*\mathbf{W}_2)$ and $\ln|\mathbf{I} + \mathbf{R}^*\mathbf{W}_2|$ are zero, the approximate secrecy capacity in (6.2) obtained by Monte Carlo coincides with the secrecy capacity in (6.1) obtained by Monte Carlo. We also plot the figure of $\mathrm{Tr}(\mathbf{R}^*)$ versus SNR to verify the behaviour of different algorithms knowing that the $\mathrm{Tr}(\mathbf{R}^*)$ represents the transmit power.

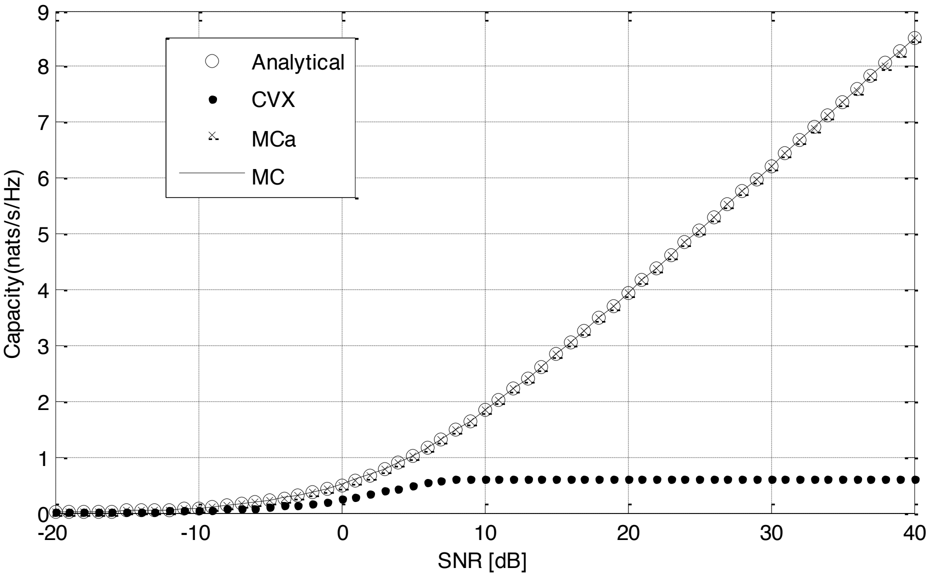


Figure 6.20. The secrecy capacity obtained by CVX, MC, and the analytical solution (Case 6-4).

Figure 6.21 shows that the $\mathbf{R}^*$ returned by CVX does not use the full power at high SNR region, which indicates that CVX is not able to figure out the direction that is orthogonal to the nonzero eigenvector of $\mathbf{W}_2$ and that the eigenvectors of $\mathbf{R}^*$ returned by CVX are not correct. Tables 6.6 and 6.7 examine this conclusion.

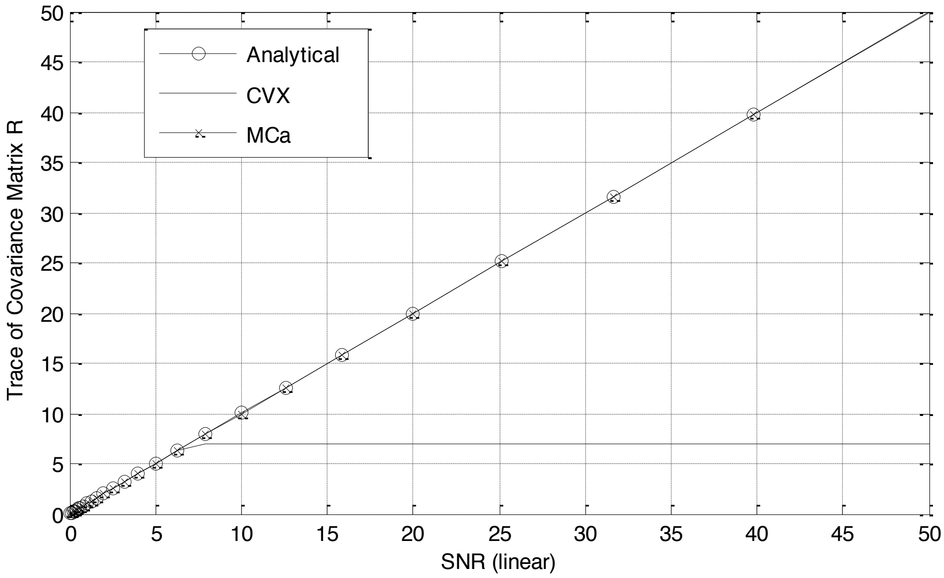


Figure 6.21. Trace$(\mathbf{R}^*)$ returned by CVX, MC, and the analytical solution (Case 6-4).

We observe that the eigenvectors of $\mathbf{R}^*$ returned by CVX are always the same as the eigenvectors of $\mathbf{W}_1$ which are $\begin{bmatrix} 0.5257 \\ -0.8507 \end{bmatrix}$ and $\begin{bmatrix} 0.8507 \\ 0.5257 \end{bmatrix}$ (see Table 6.7) which follows the observation in Tables 6.4 and 6.5. Since in Case 6-4, regardless of whether we use the approximated formula in (6.2) or the exact formula in (6.1), $\ln|\mathbf{I} + \mathbf{W}_2\mathbf{R}|$ and $\mathrm{Tr}(\mathbf{W}_2\mathbf{R})$ are both eliminated by the optimal covariance matrix in high SNR regime, hence we conclude that for the case where $\mathbf{W}_2$ is a singular matrix and $\mathcal{N}(\mathbf{W}_2) \notin \mathcal{N}(\mathbf{W}_1)$, this analytical solution is sufficient for finding the null space and returning the accurate solution when SNR is high.

Table 6.6. $\mathbf{R}^*$ returned by CVX, MC, and the analytical solution (Case 6-4).

SNR [dB]	$\mathbf{R}_{\text{cvx}}$	$\mathbf{R}_{\text{Analytical}}$	$\mathbf{R}_{\text{MCa}}$	$\mathbf{R}_{\text{MC}}$
-20	$\begin{bmatrix} 0.0072 & 0.0045 \\ 0.0045 & 0.0028 \end{bmatrix}$	$\begin{bmatrix} 0.0100 & -0.0002 \\ -0.0002 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.01 & -0.0002 \\ -0.0002 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.01 & -0.0001 \\ -0.0001 & 0 \end{bmatrix}$
0	$\begin{bmatrix} 0.3213 & -0.3574 \\ -0.3574 & 0.6787 \end{bmatrix}$	$\begin{bmatrix} 0.7473 & -0.4345 \\ -0.4345 & 0.2526 \end{bmatrix}$	$\begin{bmatrix} 0.7475 & -0.4344 \\ -0.4344 & 0.2525 \end{bmatrix}$	$\begin{bmatrix} 0.8019 & -0.3986 \\ -0.3986 & 0.1981 \end{bmatrix}$
20	$\begin{bmatrix} 2.0001 & -3.0001 \\ -3.0001 & 5.0001 \end{bmatrix}$	$\begin{bmatrix} 50.4448 & -49.9513 \\ -49.9513 & 49.4587 \end{bmatrix}$	$\begin{bmatrix} 50.4723 & -49.9972 \\ -49.9972 & 49.5277 \end{bmatrix}$	$\begin{bmatrix} 50.5258 & -49.9762 \\ -49.9762 & 49.4742 \end{bmatrix}$
40	$\begin{bmatrix} 2.0000 & -3.0000 \\ -3.0000 & 5.0000 \end{bmatrix}$	$\begin{bmatrix} 4999.2 & -4998.8 \\ -4998.8 & 4998.3 \end{bmatrix}$	$\begin{bmatrix} 4997.4 & -4999.9 \\ -4999.9 & 5002.6 \end{bmatrix}$	$\begin{bmatrix} 4999.5 & -4999.5 \\ -4999.5 & 5000.5 \end{bmatrix}$

Table 6.7. The corresponding eigenvalues and eigenvectors of matrices given in Table 6.5 ($\lambda, \boldsymbol{v}$ denote the eigenvalue and eigenvector).

SNR [dB]	$\boldsymbol{v}(\mathbf{R}_{\text{cvx}})$		$\boldsymbol{v}(\mathbf{R}_{\text{Analytical}})$		$\boldsymbol{v}(\mathbf{R}_{\text{MCa}})$		$\boldsymbol{v}(\mathbf{R}_{\text{MC}})$	
20	0.5257 −0.8507	−0.8507 −0.5257	0.9998 −0.0196	0.0196 0.9998	−0.9998 0.0211	−0.0211 −0.9998	−1 0.0093	−0.0093 −1
λ	0.01	0	0.01	0	0.01	0	0.01	0
0	0.5257 −0.8507	−0.8507 −0.5257	0.8645 −0.5026	0.5026 0.8645	−0.8646 0.5025	−0.5025 −0.8646	−0.8955 0.4451	−0.4451 −0.8955
λ	0.8996	0.1004	0.9999	0	1	0	1	0
20	0.5257 −0.8507	−0.8507 −0.5257	0.7106 −0.7036	0.7036 0.7106	−0.7104 0.7038	−0.7038 −0.7104	−0.7108 0.7034	−0.7034 −0.7108
λ	6.8543	0.1459	99.9075	0	99.9994	0.0006	99.979	0.021
40	0.5257 −0.8507	−0.8507 −0.5257	−0.7071 0.7071	−0.7071 −0.7071	−0.7069 0.7073	−0.7073 −0.7069	−0.7071 0.7071	−0.7071 −0.7071
λ	6.5843	0.1459	9997.5	0	9999.9	0.1	9999.5	0.5

The Reformulated Objective Functions:

Since we have concluded that CVX is neither able to deal with our original secrecy capacity formula nor the trace approximated formula correctly, we will try to reformulate the trace approximated formula and the associated constraints. Since CVX has several different solvers [63], we will also switch the solvers to compare their performances.

The approximation for weak eavesdropper (6.2) can be reformulated and expressed as

$$C_a = \max_{\mathbf{R},t}\{t - \operatorname{Tr}(\mathbf{W}_2\mathbf{R})\};\ \text{s.t. } \mathbf{R} \geq \mathbf{0}, t - \ln|\mathbf{I} + \mathbf{W}_1\mathbf{R}| \leq 0 \qquad (6.17)$$

We continue to use the channel model given in Case 6-4 by using different solvers of CVX respectively.

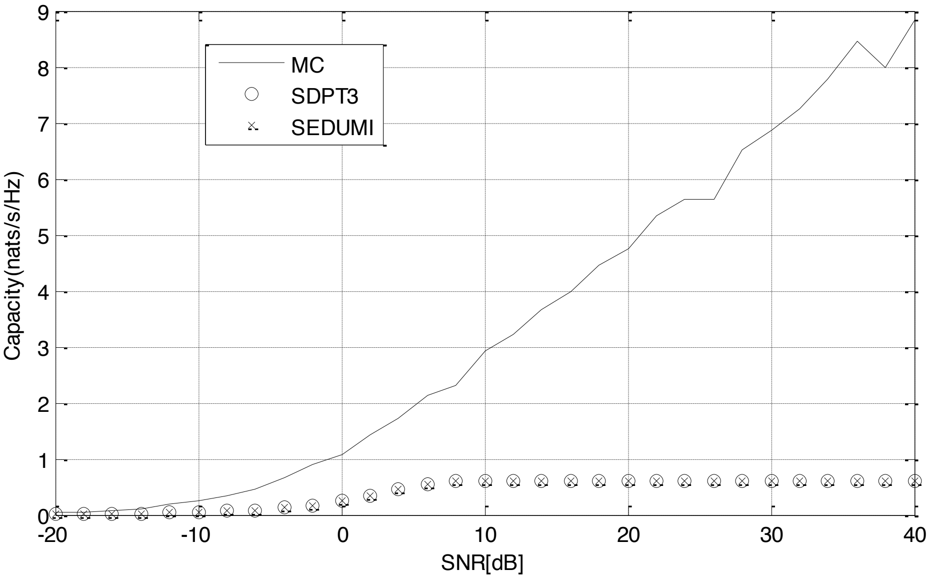


Figure 6.22. The secrecy capacity of Case 6-4 returned by Solvers SDPT3 and SEDUMI.

As we can observe in Figure 6.22, the CVX capacity is still lower than that of Monte Carlo (note that to save time, we did not use enough trials for Monte Carlo, and the results returned by Monte Carlo here are only used for references but not for precise values). There is no difference between the results returned by different solvers.

6.3 Using YALMIP

Since we have confirmed that CVX is not able to solve the optimization problem of secrecy capacity of Gaussian MIMO wiretap channel, we will try to employ another optimization toolbox of MATLAB: YALMIP [62]. We will use it to optimize (6.1) and (6.2).

Notations:

Analytical: The approximate secrecy capacity in (6.2) solved based on the analytical solution $(6.10)-(6.16)$ (weak $\mathbf{W}_2$).

MCe: The secrecy capacity in (6.1) obtained by Monte Carlo.

YALMIPa: The approximate secrecy capacity in (6.2) obtained by YALMIP.

YALMIPe: The secrecy capacity in (6.1) obtained by YALMIP.

Note that since we have confirmed that the secrecy capacity obtained by the analytical solution of trace approximation is correct (when $\mathbf{W}_2$ is weak), we will not plot the approximated secrecy capacity obtained by Monte Carlo as our reference.

Case 6-5: $\mathbf{W}_1 = \frac{1}{3}\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$ $\mathbf{W}_2 = \frac{0.1}{2}\mathbf{E}_2$

As we can observe in Figure 6.23, at low SNR region, the secrecy capacity obtained by YALMIP are even higher than that obtained by Monte Carlo. Thus, we found the covariance matrices returned by different methods and corresponding eigenvalues (given in Table 6.8). It can be concluded that at low SNR, the matrices returned by YALMIP cannot be covariance matrices since they are not positive semidefinite. However, YALMIP is able to compute the correct solution when SNR is high. If we add $\mathbf{R}^* \geq \mathbf{0}$ as a constraint into the program, the result is shown in Figure 6.24.

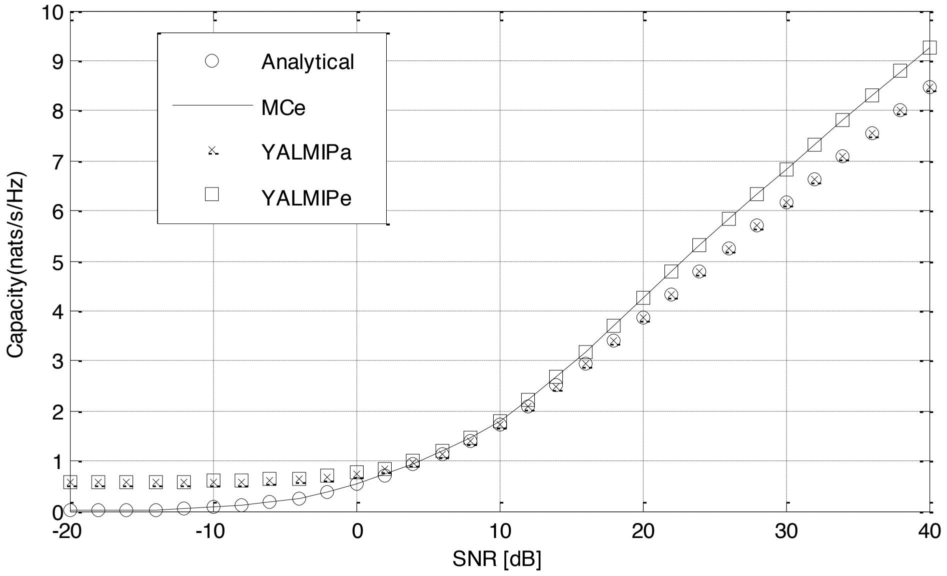


Figure 6.23. The secrecy capacity of Case 6-5 obtained by MC, YALMIP and the analytical solution.

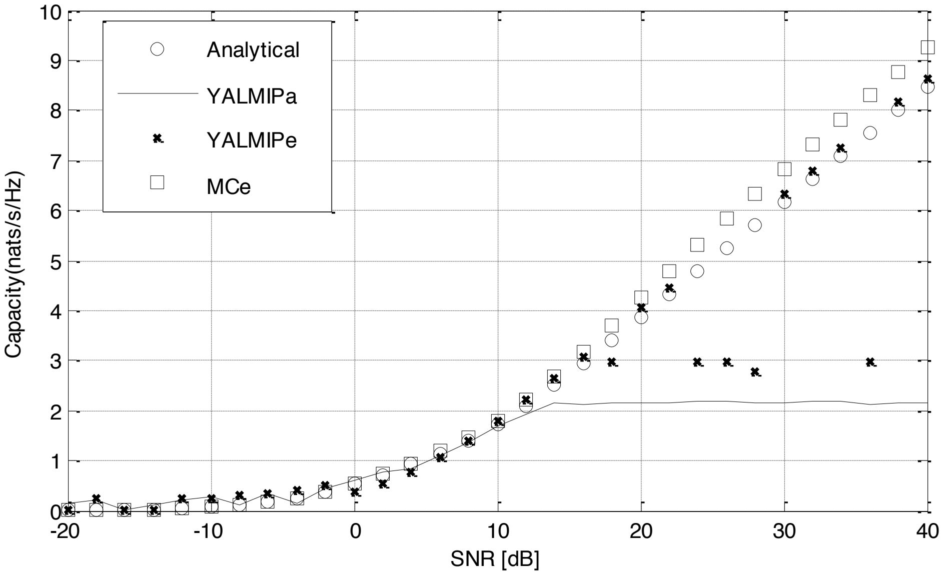


Figure 6.24. The secrecy capacity of Case 6-5 obtained by MC, YALMIP and the analytical solution (constrained by $\mathbf{R}^* \geq \mathbf{0}$).

We can see that the results returned by YALMIP do not coincide with the results returned by Monte Carlo throughout the whole SNR region. The curve ‘YAMLIPe’ oscillates and does not monotonically increase with SNR. We can conclude that YALMIP is not able to solve our optimization problem properly.

Table 6.8. $\mathbf{R}^*$, eigenvalues and eigenvalues of $\mathbf{R}^*$ returned by MC, YALMIP and the analytical solution (the capacity is in [nats/s/Hz]).

	$\mathbf{R}_{\text{analytical}}$		$\mathbf{R}_{\text{YALMIP(aprox)}}$		$\mathbf{R}_{\text{MC(exact)}}$		$\mathbf{R}_{\text{YALMIP(exact)}}$	
-20 dB	$\begin{bmatrix}0.0075 & 0.0043\\0.0043 & 0.0025\end{bmatrix}$		$\begin{bmatrix}1.505 & 2.0321\\2.0321 & -1.495\end{bmatrix}$		$\begin{bmatrix}0.0076 & 0.0043\\0.0043 & 0.0024\end{bmatrix}$		$\begin{bmatrix}1.505 & 2.1948\\2.1948 & -1.4950\end{bmatrix}$	
$\boldsymbol{v}$	0.8682 0.4962	-0.4962 0.8682	0.4506 -0.8927	-0.8927 -0.4506	0.4873 -0.8732	-0.8732 -0.4873	0.4668 -0.8844	-0.8844 -0.4668
λ	0.01	0	-2.5207	2.5307	0	0.01	-2.6534	2.6634
$C_s(C_a)$	0.0077		0.5622		0.0077		0.5818	
0 dB	$\begin{bmatrix}0.488 & 0.2645\\0.2645 & 0.1434\end{bmatrix}$		$\begin{bmatrix}2 & 1.8197\\1.8197 & -1\end{bmatrix}$		$\begin{bmatrix}0.7769 & 0.4163\\0.4163 & 0.2231\end{bmatrix}$		$\begin{bmatrix}2 & 2.0399\\2.0399 & -1\end{bmatrix}$	
$\boldsymbol{v}$	0.8861 0.4634	-0.4634 0.8861	0.4266 -0.9045	-0.9045 -0.4266	0.4724 -0.8814	-0.8814 -0.4724	0.4514 -0.8923	-0.8923 -0.4514
λ	0.9992	0	-1.8582	2.8582	0	1	-2.0321	3.0321
$C_s(C_a)$	0.534		0.7323		0.5381		0.7578	
20 dB	$\begin{bmatrix}51.5455 & -42.4545\\-42.4545 & 48.5455\end{bmatrix}$		$\begin{bmatrix}51.5 & -42.4098\\-42.4098 & 48.5\end{bmatrix}$		$\begin{bmatrix}52.1156 & -28.4266\\-28.4266 & 47.8844\end{bmatrix}$		$\begin{bmatrix}51.5 & -28.397\\-28.397 & 48.5\end{bmatrix}$	
$\boldsymbol{v}$	-0.6945 -0.7195	-0.7195 0.6945	-0.6945 -0.7195	-0.7195 0.6945	-0.6804 -0.7329	-0.7329 0.6804	-0.6882 -0.7255	-0.7255 0.6882
λ	7.5644	92.5265	7.5636	92.4363	21.4948	78.5052	21.5634	78.4366
$C_s(C_a)$	3.8561		3.8552		4.2452		4.2452	
40 dB	$\begin{bmatrix}5002 & -4992\\-4992 & 4999\end{bmatrix}$		$\begin{bmatrix}5001.5 & -4991.5\\-4991.5 & 4998.5\end{bmatrix}$		$\begin{bmatrix}5003.7 & -4749.7\\-4749.7 & 4996.3\end{bmatrix}$		$\begin{bmatrix}5001.5 & -4718.2\\-4718.2 & 4998.5\end{bmatrix}$	
$\boldsymbol{v}$	-0.7070 -0.7072	-0.7072 0.7070	-0.7070 -0.7072	-0.7072 0.7070	-0.7068 -0.7074	-0.7074 0.7068	-0.7070 -0.7072	-0.7072 0.7070
λ	8.5	9992.5	8.5	9991.5	250.3	9749.7	281.8	9718.2
$C_s(C_a)$	8.4657		8.4656		9.2583		9.2583	

6.4 Summary

In this chapter, we discussed the approximation formula for weak eavesdropper of Gaussian wiretap MIMO channel secrecy capacity that is accepted by CVX and we concluded that CVX is not able to solve it properly since it cannot return the correct optimal eigenvectors of the covariance matrix. We also verified the analytical solution, which is derived from the trace approximation formula, for the case where the eavesdropper is weak. Considering how we use the bisection to compute the solution, the accuracy of the solution may be affected by the stopping criteria of the bisection. Based on our observations, the solution is precise enough if we used proper stopping condition. For the case where $\mathbf{W}_2$ is a singular matrix and $\mathcal{N}(\mathbf{W}_2) \notin \mathcal{N}(\mathbf{W}_1)$, the analytical solution is able to find the null space of the eavesdropper such that the algorithm works for computing the secrecy capacity even if the $\mathbf{W}_2\mathbf{R} \nless \mathbf{I}$ (high SNR scenario or $\mathbf{W}_2$ is not weak). We tried using YALMIP instead of CVX, but observed that it cannot handle the $\mathbf{R} \geq \mathbf{0}$ constraint properly (see Figure 6.24). Hence CVX and YALMIP are not able to solve the computation of secrecy capacity except for some special cases.

7. Linear Approximation

As we have concluded in Chapter 6, the toolboxes of MATLAB (CVX and YALMIP) are not capable of solving (7.1) properly. In this section we will provide a new method for the optimization problem for the secrecy capacity of Gaussian MIMO wiretap channel. As we have observed in the previous section, CVX is unable to accept the '$\ln(|\mathbf{A}|) - \ln(|\mathbf{B}|)$' function and has a difficulty with the '$\ln(|\mathbf{A}|) - \mathrm{Trace}(\mathbf{B})$' function in term of returning optimal eigenvectors of the transmit covariance matrix, where $\ln(|\mathbf{A}|)$ and $\ln(|\mathbf{B}|)$ denote the logarithm function of the determinant of a matrices $\mathbf{A}$ and $\mathbf{B}$. Hence it is important to reformulate the original objective function $C_s(\mathbf{R})$ into a pattern that can be accepted by CVX directly.

It is well-known that all linear functions are both convex and concave [42], and that there is no obstacle for CVX to handle linear functions [63]. It has been discussed in Chapter 5 that $\ln(|\mathbf{A}|)$ can be approximated by $\mathrm{Trace}(\mathbf{A})$ associated with several constraints. Reference [42] shows that $\mathrm{Trace}(\mathbf{A})$ is a linear function.

We consider the optimization problem for the secrecy capacity of Gaussian MIMO wiretap channel in (3.9) which is

$$C_s = \max_{\mathbf{R}} C_s(\mathbf{R}) = \max_{\mathbf{R}}\{\ln\frac{|\mathbf{I}+\mathbf{W}_1\mathbf{R}|}{|\mathbf{I}+\mathbf{W}_2\mathbf{R}|}\};\ \mathrm{s.t.}\ \mathbf{R} \geq \mathbf{0}, \mathrm{Tr}(\mathbf{R}) \leq P_T \tag{7.1}$$

where $C_s(\mathbf{R})$ denotes the secrecy rate that is a function of $\mathbf{R}$. (7.1) can be expressed as:

$$C_s = \max_{\Delta\mathbf{R}} C_s(\Delta\mathbf{R}) = \max_{\Delta\mathbf{R}} \ln\frac{|\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R})|}{|\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R})|} \tag{7.2}$$

s.t.

$$\Delta\mathbf{R}+\mathbf{R}_0 \geq \mathbf{0}, \quad \mathrm{Tr}(\Delta\mathbf{R}) = 0$$

where $\mathbf{R}_0$ is the initial input covariance matrix and can be the identity matrix scaled by the total transmit power. We define

$$C_s(\mathbf{R}_0) = \ln\frac{|\mathbf{I}+\mathbf{W}_1\mathbf{R}_0|}{|\mathbf{I}+\mathbf{W}_2\mathbf{R}_0|} \tag{7.3}$$

which is fixed for a given $\mathbf{R}_0$. Therefore, C_s can be expressed as:

$$C_s = \max_{\Delta\mathbf{R}} C_s(\Delta\mathbf{R}) = C_s(\mathbf{R}_0) + \max_{\Delta\mathbf{R}} \ln\frac{|\mathbf{I}+(\mathbf{I}+\mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1\Delta\mathbf{R}|}{|\mathbf{I}+(\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2\Delta\mathbf{R}|} \tag{7.4}$$

s.t.

$$\Delta\mathbf{R} + \mathbf{R}_0 \geq \mathbf{0}, \quad \mathrm{Tr}(\Delta\mathbf{R}) = 0$$

where the objective function in (7.4) is a reformulation of the objective function in (7.2). If we define

$$\Delta C = \max_{\Delta\mathbf{R}} \ln \frac{|\mathbf{I} + (\mathbf{I} + \mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1\Delta\mathbf{R}|}{|\mathbf{I} + (\mathbf{I} + \mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2\Delta\mathbf{R}|}; \mathrm{s.t.}\ \Delta\mathbf{R} + \mathbf{R}_0 \geq \mathbf{0}, \mathrm{Tr}(\Delta\mathbf{R}) = 0 \qquad (7.5)$$

then ΔC can be approximated using $\ln(\mathbf{I} + \mathbf{A}) \approx \mathrm{Tr}(\mathbf{A})$ when $\mathbf{A} \ll \mathbf{I}$,

$$\Delta C \approx \max_{\Delta\mathbf{R}}(\mathrm{Tr}\{[(\mathbf{I} + \mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1 - (\mathbf{I} + \mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2]\Delta\mathbf{R}\}) \qquad (7.6)$$

when $(\mathbf{I} + \mathbf{W}_{1(2)}\mathbf{R}_0)^{-1}\mathbf{W}_{1(2)}\Delta\mathbf{R} \ll \mathbf{I}$. Hence (7.1) can be approximated as:

$$C_s = \max_{\Delta\mathbf{R}} C_s(\Delta\mathbf{R}) \qquad (7.7)$$

$$= C_s(\mathbf{R}_0) + \max_{\Delta\mathbf{R}}\{\mathrm{Tr}[(\mathbf{I} + \mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1 - (\mathbf{I} + \mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2]\Delta\mathbf{R}\}$$

s.t.

$$\Delta\mathbf{R} + \mathbf{R}_0 \geq \mathbf{0}, \mathrm{Tr}(\Delta\mathbf{R}) = 0, -\varepsilon\mathbf{I} \leq \Delta\mathbf{R} \leq \varepsilon\mathbf{I}$$

where $\varepsilon \ll 1/\lambda_{\max}(\mathbf{W}_{1(2)})$,$\lambda_{\max}(\mathbf{W}_{1(2)})$ is the largest eigenvalue of $\mathbf{W}_1$ and $\mathbf{W}_2$. Note that the last constraint ($-\varepsilon\mathbf{I} \leq \Delta\mathbf{R} \leq \varepsilon\mathbf{I}$) is derived from

$$(\mathbf{I} + \mathbf{W}_{1(2)}\mathbf{R}_0)^{-1}\mathbf{W}_{1(2)}\Delta\mathbf{R} \ll \mathbf{I} \qquad (7.8)$$

for the purpose of ensuring the validating of approximation.

Next, we apply this linearized reformulation into several optimization problems to verify. Initially, we set up

$$\varepsilon = 1/\max\{\|(\mathbf{I} + \mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1\|, \|(\mathbf{I} + \mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2\|\} \qquad (7.9)$$

where $\|\mathbf{A}\|$ denotes the 2-norm, i.e. the largest singular value of $\mathbf{A}$, and start with some simple examples where the solutions are known. The flow chart for solving Cases 7-1 and 7-2 is given in Figure 7.1.

Notations:

C_{MC}: The secrecy capacity obtained by the Monte Carlo (number of trials of MC is 10^6).

C_{cvx}: The approximated secrecy capacity in (7.7) solved by CVX.

$\mathbf{R}_{\mathrm{MC}}$: The transmit covariance matrix returned by Monte Carlo.

$\mathbf{R}_{\mathrm{CVX}}$: The transmit covariance matrix returned by CVX.

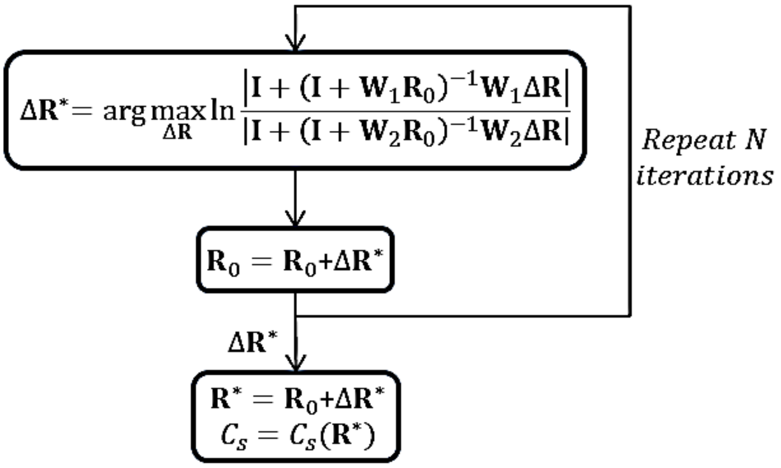

$\Delta\mathbf{R}^{*} = \arg\max_{\Delta\mathbf{R}} \ln \frac{|\mathbf{I} + (\mathbf{I} + \mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1\Delta\mathbf{R}|}{|\mathbf{I} + (\mathbf{I} + \mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2\Delta\mathbf{R}|}$
Repeat N iterations
$\mathbf{R}_0 = \mathbf{R}_0 + \Delta\mathbf{R}^{*}$
$\Delta\mathbf{R}^{*}$
$\mathbf{R}^{*} = \mathbf{R}_0 + \Delta\mathbf{R}^{*}$
$C_s = C_s(\mathbf{R}^{*})$

Figure 7.1. Flow chat for solving Cases 7-1 and 7-2.

Case 7-1: $\mathbf{W}_1 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$ $\mathbf{W}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$; $\mathrm{SNR} = 30$ dB.

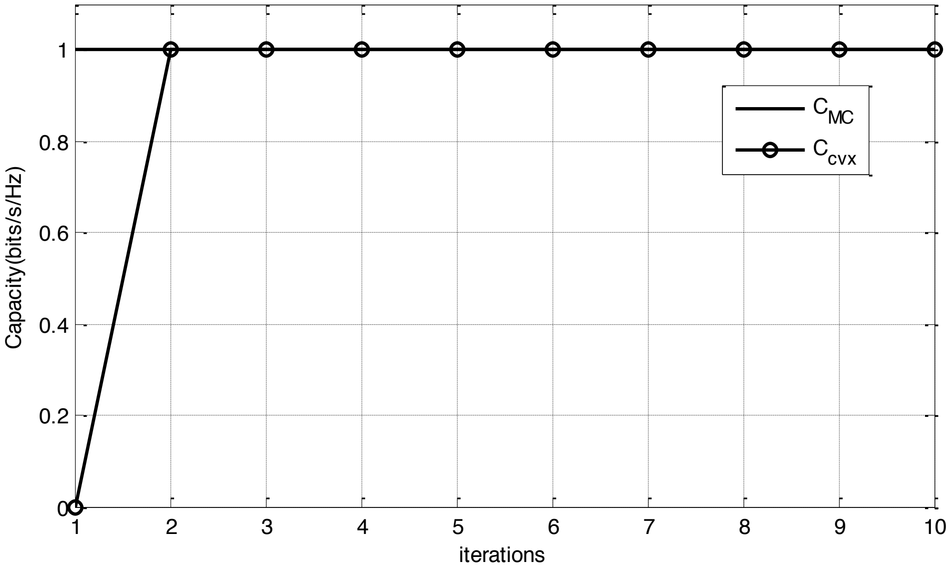


Figure 7.2. The secrecy capacity of Case 7-1 obtained by Monte Carlo and CVX ($\mathrm{SNR} = 30$ dB).

It has been shown that the optimal covariance matrix has rank-one when $r_{+}(\Delta\mathbf{W}) = r_{+}(\mathbf{W}_1 - \mathbf{W}_2) = 1$ ($r_{+}(\mathbf{A})$ denotes the number of positive eigenvalue of $\mathbf{A}$) [30]. In this case, the eigenvalues, eigenvectors of the channel matrices and the difference channel are provided in Table 7.1. The optimal covariance matrix $\mathbf{R}^* = \begin{bmatrix} P_T & 0 \\ 0 & 0 \end{bmatrix}$ and $C_s \approx 1$ are obtained based on (5.12) which is

$$C_s = \ln \lambda_{\max}, \; \mathbf{R}^* = P_T \boldsymbol{v}_{\max} \boldsymbol{v}_{\max}^{+} \tag{7.10}$$

where $\lambda_{\max}$ and $\boldsymbol{v}_{\max}$ are the largest eigenvalue and corresponding eigenvector of $(\mathbf{I} + P_T\mathbf{W}_2)^{-1}(\mathbf{I} + P_T\mathbf{W}_1)$. Observing the results in Figure 7.2 and Table 7.1, $\mathbf{R}_{\text{cvx}}$ approximately equals to $\mathbf{R}^*$ obtained based on (7.10). The approximated result obtained by Monte Carlo is also close to the precise solution. We can conclude that CVX works for this case by utilizing the approximation in (7.7). Meanwhile, it is important to notice that the channel in this example is not degraded ($\Delta\mathbf{W} = \mathbf{W}_1 - \mathbf{W}_2 \not\succeq \mathbf{0}$). This indicates that $C_s(\mathbf{R})$ is not concave but can be reformulated and optimized by CVX.

Table 7.1. Channel matrices, difference channel, covariance matrices and their eigenvectors ($\boldsymbol{v}$), eigenvalues (λ) of Case 7-1 ($\text{SNR} = 30$ dB).

	$\mathbf{W}_1$		$\mathbf{W}_2$		$\mathbf{W}_1 - \mathbf{W}_2$		$\mathbf{R}_{\text{MC}}$		$\mathbf{R}_{\text{cvx}}$		$\mathbf{R}^*$	
	$\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$		$\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$		$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$		$\begin{bmatrix} 999.9953 & 2.1631 \\ 2.1631 & 0.0047 \end{bmatrix}$		$\begin{bmatrix} 1000 & 0 \\ 0 & 0 \end{bmatrix}$		$\begin{bmatrix} 1000 & 0 \\ 0 & 0 \end{bmatrix}$	
λ	1	2	2	1	-1	1	0	1000	0	1000	0	1000
$\boldsymbol{v}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} -0.0022 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0.0022 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$

Case 7-2: $\mathbf{W}_1 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$ $\mathbf{W}_2 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$; $\text{SNR} = 30$ dB.

In this case, $r_+(\Delta\mathbf{W}) = r_+(\mathbf{W}_1 - \mathbf{W}_2) = 1$ while $\mathbf{W}_1$ and $\mathbf{W}_2$ have different eigenvectors. Hence, the optimal covariance matrix $\mathbf{R}^*$ has rank-one and can be solved based on (7.10).

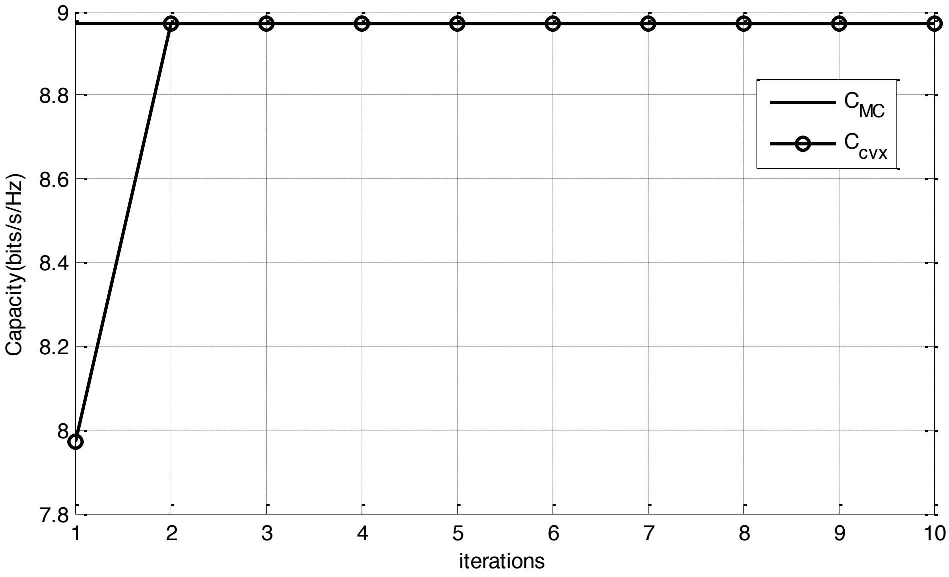


Figure 7.3. The secrecy capacity of Case 7-2 obtained by Monte Carlo and CVX (SNR = 30 dB).

Observing that $\mathbf{R}_{\text{cvx}}$ is a rank-one matrix and by comparing it with $\mathbf{R}_{\text{MC}}$ and $\mathbf{R}^*$, it can be concluded that CVX works for this case (see Table 7.2).

Table 7.2. Channel matrices, difference channel, covariance matrices and their eigenvectors ($\boldsymbol{v}$), eigenvalues (λ) of Case 7-2 (SNR = 30 dB).

	$\mathbf{W}_1$		$\mathbf{W}_2$		$\mathbf{W}_1 - \mathbf{W}_2$	
	$\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$		$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$		$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$	
λ	0.382	2.618	0	2	0	1
$\boldsymbol{v}$	$\begin{bmatrix} 0.5257 \\ -0.8057 \end{bmatrix}$	$\begin{bmatrix} 0.8507 \\ 0.5257 \end{bmatrix}$	$\begin{bmatrix} -0.7071 \\ 0.7071 \end{bmatrix}$	$\begin{bmatrix} 0.7071 \\ 0.7071 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
	$\mathbf{R}_{\text{MC}}$		$\mathbf{R}_{\text{cvx}}$		$\mathbf{R}^*$	
	$\begin{bmatrix} 500.639 & -499.996 \\ -499.996 & 499.361 \end{bmatrix}$		$\begin{bmatrix} 500.998 & -500 \\ -500 & 499.002 \end{bmatrix}$		$\begin{bmatrix} 500.556 & -500 \\ -500 & 499.0019 \end{bmatrix}$	
λ	0.0034	999.9966	0	1000	0	1000
$\boldsymbol{v}$	$\begin{bmatrix} 0.7067 \\ 0.7076 \end{bmatrix}$	$\begin{bmatrix} -0.7076 \\ 0.7067 \end{bmatrix}$	$\begin{bmatrix} 0.7064 \\ 0.7078 \end{bmatrix}$	$\begin{bmatrix} -0.7078 \\ 0.7064 \end{bmatrix}$	$\begin{bmatrix} 0.7068 \\ 0.7075 \end{bmatrix}$	$\begin{bmatrix} -0.7075 \\ 0.7068 \end{bmatrix}$

7.1. Backtracking Line Search

To use the approximation given in (7.7) for computing the secrecy capacity of

Gaussian MIMO wiretap channel, backtracking (see Ref. [42]) is introduced to establish the relationship between (7.1) and (7.7). As mentioned in last section:

$$\varepsilon = 1/\max\{\|(\mathbf{I}+\mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1\|, \|(\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2\|\} \tag{7.11}$$

is chosen as the initial step size ceiling (optimizing/searching range). In some cases, the oscillation may appear as the objective value becomes close to the optimal point if we continue with the fixed step size throughout the optimization process. Backtracking helps the system adjust the step size.

The condition of the backtracking is [42]:

$$f(x+t\Delta x) < f(x) + \alpha t \nabla f(x)^{\mathrm{T}}\Delta x \tag{7.12}$$

$$t = \frac{1}{\beta^n} \tag{7.13}$$

where $f(x)$ denotes the objective function to be maximized with respect to variable x, Δx is step size (ascent direction), $\alpha \in (0,0.5)$ is interpreted as a fraction of the ascent, n is the number of iterations, and t is reduced by the scale factor $\beta \in (0,1)$ (here we set $\beta = \frac{1}{2}$) [42] in each iteration of backtracking until the stopping condition (7.14) is satisfied. Since $\Delta \mathrm{x}$ is ascent direction, we have $\nabla f(x)^T\Delta x > 0$, when t is small enough and $f(x)$ is concave,

$$f(x+t\Delta x) \approx f(x) + t\nabla f(x)^{\mathrm{T}}\Delta x > f(x) + \alpha t\nabla f(x)^{\mathrm{T}}\Delta x \tag{7.14}$$

which indicates that the backtracking line search eventually terminates [42]. In other words, the backtracking eventually stops when $t \approx 0$ or $\Delta x \approx 0$. In the optimization problem for the secrecy capacity, t is initialized by 1, ascent direction is $\Delta\mathbf{R}$, and $f(\mathbf{R})$ is the original secrecy rate formula in (7.1) which is a function of $\mathbf{R}$ $(\Delta\mathbf{R})$ and has the form:

$$f(\mathbf{R}) = C_s(\Delta\mathbf{R}) = \ln\frac{|\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R})|}{|\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R})|} \tag{7.15}$$

and the gradient of the function in (7.15) is

$$\nabla f(\mathbf{R}) = \nabla C_s(\Delta\mathbf{R}) = \mathrm{Tr}\{[(\mathbf{I}+\mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1 - (\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2]\Delta\mathbf{R}\} \tag{7.16}$$

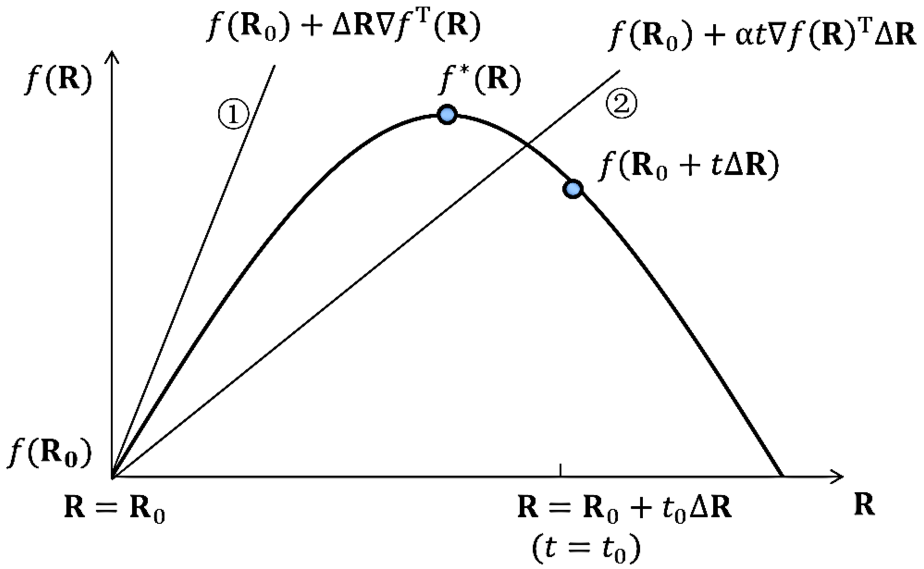

$f(\mathbf{R}_0) + \Delta\mathbf{R}\nabla f^{\mathrm{T}}(\mathbf{R})$
$f(\mathbf{R}_0) + \alpha t \nabla f(\mathbf{R})^{\mathrm{T}}\Delta\mathbf{R}$
$f(\mathbf{R})$
①
$f^*(\mathbf{R})$
②
$f(\mathbf{R}_0 + t\Delta\mathbf{R})$
$f(\mathbf{R_0})$
$\mathbf{R} = \mathbf{R}_0$
$\mathbf{R} = \mathbf{R}_0 + t_0\Delta\mathbf{R}$
$(t = t_0)$
$\mathbf{R}$

Figure 7.4 Backtracking Line search

Figure 7.4 shows a schematic diagram of backtracking line search. Note that in the optimization problem for the secrecy capacity, the objective function $C_s(\mathbf{R})$ cannot be simply represented as Figure 7.4 since $C_s(\mathbf{R})$ is neither an increasing function nor a decreasing function of $\mathbf{R}$. The *x* axis does not simply stand for the increase direction of $\mathbf{R}$. $\Delta\mathbf{R}$ in Figure 7.4 is returned by CVX via solving (7.7), i.e. ray ①, which is the linear approximation of $f(\mathbf{R})$ $(C_s(\mathbf{R}))$; the lower ray ② whose slope is α times smaller than ①. The backtracking stops when $f(\mathbf{R}_0 + t\Delta\mathbf{R})$ is greater than ②. Since the argument in the optimization problem for the secrecy capacity is transmit covariance which has to be positive semidefinite matrix. Therefore, besides the stopping condition given by (7.14), another condition which is $\mathbf{R} + t\Delta\mathbf{R}^* \geq \mathbf{0}$ needs to be added, then the t will be reduced by scale factor β if $\mathbf{R} + t\Delta\mathbf{R}^* < \mathbf{0}$ (see Figure 7.5).

In the following, we implement the backtracking line search in several examples. The flow chat of this optimization process is shown in Figure 7.5.

Notations:

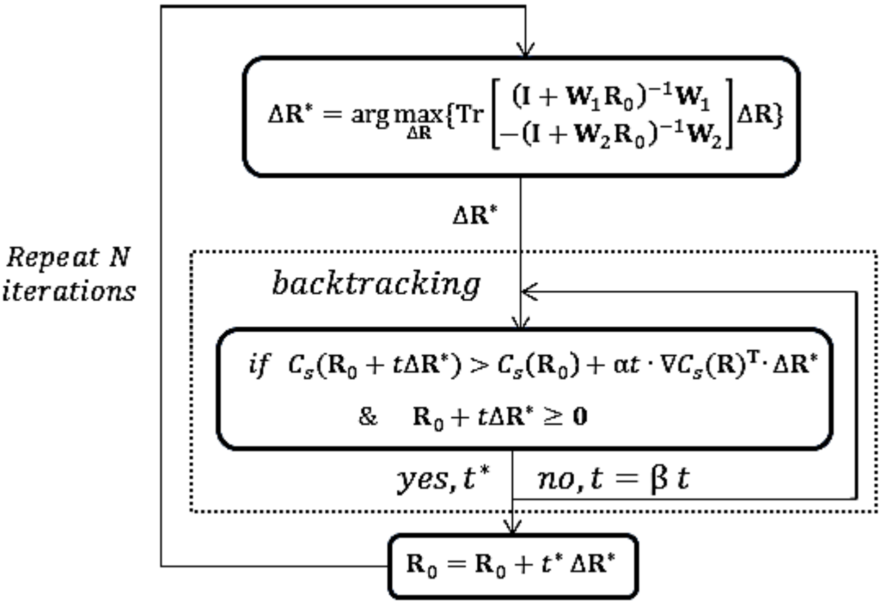
C_{MC}: The secrecy capacity obtained by Monte Carlo.

C_{cvx}: The secrecy capacity in (7.7) obtained by CVX (with backtracking)

$\mathbf{R}_{\mathrm{MC}}$: The transmit covariance matrix returned by Monte Carlo.

$\mathbf{R}_{\text{cvx}}$: The transmit covariance matrix returned by CVX (with backtracking).

n: The number of iterations of backtracking for each $\Delta\mathbf{R}$ returned by CVX.



$\Delta R^* = \arg\max_{\Delta R}\{\mathrm{Tr}\begin{bmatrix}(I+W_1R_0)^{-1}W_1\\-(I+W_2R_0)^{-1}W_2\end{bmatrix}\Delta R\}$
ΔR^*
Repeat N iterations
backtracking
if $C_s(R_0+t\Delta R^*) > C_s(R_0)+\alpha t\cdot\nabla C_s(R)^T\cdot\Delta R^*$
& $R_0+t\Delta R^* \geq 0$
yes, t^*
no, $t=\beta t$
$R_0 = R_0 + t^*\Delta R^*$

Figure 7.5. The flow chat of the optimization problem in (7.7) with backtracking (t^* denotes the optimal t obtained by backtracking when the stopping condition is satisfied).

We use the channel matrices discussed in last section where the analytical solution is available.

Case 7-3: $\mathbf{W}_1 = \begin{bmatrix}2 & 1\\1 & 1\end{bmatrix}$ $\mathbf{W}_2 = \begin{bmatrix}1 & 1\\1 & 1\end{bmatrix}$; SNR = 30 dB.

$C_{\text{MC}} = 8.969384881$

$\mathbf{R}_{\text{MC}} = \begin{bmatrix}500.50190 & -499.998867\\-499.998867 & 499.498099\end{bmatrix}$

$\mathbf{R}_{\text{cvx}} = \begin{bmatrix}500.499817 & -499.999366\\-499.999366 & 499.500183\end{bmatrix}$

The secrecy capacity of this case based on (7.10) is $C_s = \log_2(501.2499) \approx 8.969386234$ where 501.2499 is the largest eigenvalue of $(\mathbf{I}+P_T\mathbf{W}_2)^{-1}(\mathbf{I}+P_T\mathbf{W}_1)$.

Table 7.3. The secrecy capacity obtained by CVX with backtracking ($\mathrm{SNR} = 30\ \mathrm{dB}$).

Iterations	t	C_{cvx}	n
1	1	8.968669661219078	0
2	1	8.969385577578551	0
3	0.0039062500	8.969385584356981	8
4	0.0009765625	8.969385585149238	10
5	0.0312500000	8.969385591172379	5
6	0.0039062500	8.969385603654891	8
7	0.0078125000	8.969385605851480	7
8	0.0019531250	8.969385608150585	9
9	0.0019531250	8.969385608553282	9
10	0.0009765625	8.969385609231416	10

The results obtained by CVX (with backtracking) and Monte Carlo of Case 7-3 demonstrate that CVX achieves an accurate enough secrecy capacity in two iterations. If we observe more decimal places in Table 7.3, it can be noticed that the secrecy capacity increases slowly as it encounters more iterations. It is important to note that the t returned by the backtracking line search for the first two iterations are both 1 which means that the $\Delta\mathbf{R}$ (step size) returned by CVX is fully utilized in these two steps. The value of t becomes smaller for the latter iterations that reveals that the closer the result to the true secrecy capacity, the smaller the step size is needed.

Case7-4: $\mathbf{W}_1 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$, $\mathbf{W}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$; $\mathrm{SNR} = -10\ \mathrm{dB}$.

We have implemented these channel matrices where the optimal covariance matrix is $\mathbf{R}^* = \begin{bmatrix} P_T & 0 \\ 0 & 0 \end{bmatrix}$.

$C_{\mathrm{MC}} = 0.1255$; $\mathbf{R}_{\mathrm{MC}} = \begin{bmatrix} 0.1 & 0 \\ 0 & 0 \end{bmatrix}$; $\mathbf{R}_{\mathrm{cvx}} = \begin{bmatrix} 0.1 & 0 \\ 0 & 0 \end{bmatrix}$.

Table 7.4 shows that the optimization can be approximately finished in one iteration for this case. Note that the $\Delta\mathbf{R}$ returned by CVX in the second iteration is $\mathbf{0}$ (considering the precision of CVX, this is a numerical $\mathbf{0}$ instead of an exact value). The conditions of convergence terminating in one step will be discussed in A.1.

Table 7.4. The secrecy capacity obtained by CVX with backtracking ($\mathrm{SNR} = -10\ \mathrm{dB}$).

Iterations	t	C_{cvx}	n
1	1	0.1255	0
2	1	0.1255	0
3	1	0.1255	0
4	1	0.1255	0
5	1	0.1255	0
6	1	0.1255	0
7	1	0.1255	0
8	1	0.1255	0
9	1	0.1255	0
10	1	0.1255	0

Based on the observation in this section, it can be concluded that the backtracking line search improves the accuracy and the efficiency for solving the approximated optimization problem. However, there is a point need to be noticed. The objective function $C_s(\mathbf{R})$ given in (7.1) is not concave unless $\mathbf{W}_1 \geq \mathbf{W}_2$, hence the stopping condition (7.14) may not be defined properly in some cases so that the backtracking line search might be terminated when the local optimum is obtained.

7.2. Min-Max Algorithm

In this section, we will discuss an algorithm for achieving the secrecy capacity which is introduced in [35]. An auxiliary matrix $\mathbf{K}$ is introduced into the secrecy rate function $C_s(\mathbf{R})$ to assist in obtaining the secrecy capacity. The modified optimization problem has the form:

$$C_s = \min_{\mathbf{K}} \max_{\mathbf{R}} F(\mathbf{R}, \mathbf{K}) = \min_{\mathbf{K}} \max_{\mathbf{R}} \{\ln \frac{|\mathbf{I} + \mathbf{K}^{-1}\mathbf{H}\mathbf{R}\mathbf{H}^{+}|}{|\mathbf{I} + \mathbf{W}_2\mathbf{R}|}\} \tag{7.17}$$

s.t. $$\mathbf{R} \geq \mathbf{0}, \mathrm{Tr}(\mathbf{R}) \leq \mathrm{P_T}, \mathbf{K} \geq \mathbf{0}$$

where $\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \end{bmatrix}$; the matrix $\mathbf{K}$ has the fixed form $\begin{bmatrix} \mathbf{I} & \mathbf{\Phi}^{+} \\ \mathbf{\Phi} & \mathbf{I} \end{bmatrix}_{2m\times 2m}$ where $\mathbf{\Phi}$ is an $m \times m$ matrix and denotes the optimal cross-covariance. The min-max problem given in (7.17) is convex-concave with saddle point solution $(\mathbf{R}^*, \mathbf{K}^*)$, and the saddle point solution is proved to be existed [35]. It is shown that $C_s(\mathbf{R}^*)$ provides the lower bound of the secrecy capacity C_s^- of given channel matrices, where

$$\mathbf{R}^* = \arg\max_{\mathbf{R}} \left\{ \ln \frac{\left|\mathbf{I} + \mathbf{K}^{-1}\mathbf{HRH}^{+}\right|}{|\mathbf{I} + \mathbf{W}_2\mathbf{R}|} \right\} \tag{7.18}$$

An upper bound of the secrecy capacity C_s^+ is given by $F(\mathbf{R}, \mathbf{K}^*)$ where

$$\mathbf{K}^* = \arg\min_{\mathbf{K}} \left\{ \ln \frac{\left|\mathbf{I} + \mathbf{K}^{-1}\mathbf{HRH}^{+}\right|}{|\mathbf{I} + \mathbf{W}_2\mathbf{R}|} \right\} \tag{7.19}$$

The saddle point solution $(\mathbf{R}^*, \mathbf{K}^*)$ is achieved when the lower bound overlaps with the upper bound [35]. $\mathbf{K}$ is initialized as $\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}_{2m\times 2m}$ in our case. Comparing the numerator and denominator, we speculate that $F(\mathbf{R}, \mathbf{K})$ given in (7.17) is more sensitive with respects to $\mathbf{K}$ than with respects to $\mathbf{R}$, hence we first maximize $F(\mathbf{R}, \mathbf{K})$ over $\mathbf{R}$ for N_1 iterations and then minimize $F(\mathbf{R}^*, \mathbf{K})$ over $\mathbf{K}$ for N_2 iterations, the whole min-max process is repeated for *N* iterations. By substituting $\mathbf{R}^*$ into $C_s(\mathbf{R})$, the lower bound of the secrecy capacity can be achieved. The value obtained by minimizing $F(\mathbf{R}^*, \mathbf{K})$ over $\mathbf{K}$ represents the upper bound of the secrecy capacity. This process is repeated for $(N_1 + N_2) \cdot N$ iterations until the saddle point solution $(\mathbf{R}^*, \mathbf{K}^*)$ is achieved, i.e. $C_s^- = C_s^+$ [35]. The flow chat of this algorithm is shown in Figure 7.6.

Since CVX is still the modeling toolbox for solving this optimization program and (7.17) is also in the form of '$\ln(|\mathbf{A}|) - \ln(|\mathbf{B}|)$', which cannot be accepted by CVX directly, the reformulation of the (7.17) is necessary. We use the similar method as what we used in the last section by approximating $\ln(|\mathbf{A}|)$ into the form of $\mathrm{Trace}(\mathbf{A})$.

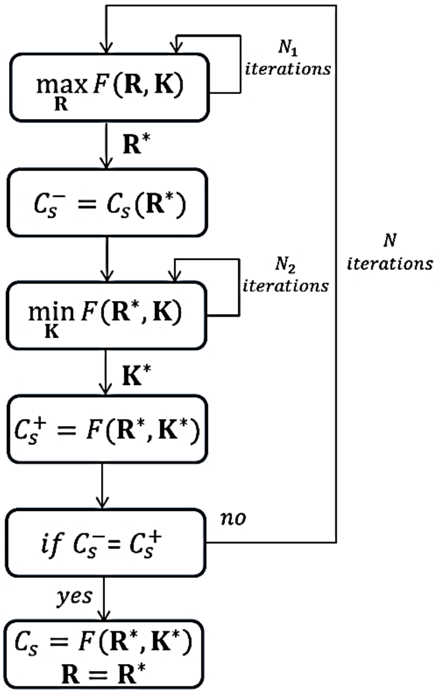

$\max_{\mathbf{R}} F(\mathbf{R},\mathbf{K})$
N_1 iterations
$\mathbf{R}^*$
$C_s^- = C_s(\mathbf{R}^*)$
N_2 iterations
N iterations
$\min_{\mathbf{K}} F(\mathbf{R}^*,\mathbf{K})$
$\mathbf{K}^*$
$C_s^+ = F(\mathbf{R}^*,\mathbf{K}^*)$
$if\ C_s^- = C_s^+$
no
yes
$C_s = F(\mathbf{R}^*,\mathbf{K}^*)$
$\mathbf{R} = \mathbf{R}^*$

Figure 7.6. Flow chat of min-max algorithm.

Lower Bound of the Secrecy Capacity:

The objective function in (7.17) is

$$F(\mathbf{R},\mathbf{K}) = \ln\frac{|\mathbf{I}+\mathbf{K}^{-1}\mathbf{H}\mathbf{R}\mathbf{H}^{+}|}{|\mathbf{I}+\mathbf{W}_2\mathbf{R}|} \tag{7.20}$$

that can be expressed as

$$\begin{aligned} F(\mathbf{R},\mathbf{K}) &= \ln\frac{|\mathbf{I}+\mathbf{W}\mathbf{R}|}{|\mathbf{I}+\mathbf{W}_2\mathbf{R}|} \\ &= \ln\frac{|\mathbf{I}+\mathbf{W}(\mathbf{R}_0+\Delta\mathbf{R})|}{|\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R})|} \end{aligned} \tag{7.21}$$

where

$$\mathbf{W} = \mathbf{K}^{-1}\mathbf{H}\mathbf{H}^{+} \tag{7.22}$$

Similar to $(7.2)-(7.7)$, (7.21) can be approximated as

$$F(\Delta\mathbf{R},\mathbf{K}) = F(\mathbf{R}_0) + \ln\frac{|\mathbf{I}+(\mathbf{I}+\mathbf{W}\mathbf{R}_0)^{-1}\mathbf{W}\Delta\mathbf{R}|}{|\mathbf{I}+(\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2\Delta\mathbf{R}|} \tag{7.23}$$

we define

$$\Delta F = \ln\frac{|\mathbf{I}+(\mathbf{I}+\mathbf{W}\mathbf{R}_0)^{-1}\mathbf{W}\Delta\mathbf{R}|}{|\mathbf{I}+(\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2\Delta\mathbf{R}|} \tag{7.24}$$

and ΔF can be approximated as

$$\Delta F \approx \mathrm{Tr}[(\mathbf{I}+\mathbf{W}\mathbf{R}_0)^{-1}\mathbf{W}-(\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2]\Delta\mathbf{R} \tag{7.25}$$

when $\left(\mathbf{I}+\mathbf{W}_{(2)}\mathbf{R}_0\right)^{-1}\mathbf{W}_{(2)}\Delta\mathbf{R} \ll \mathbf{I}$. Hence the optimization problem for the transmit covariance matrix that achieves C_s^- is equivalent to find

$$\mathbf{R}^* = \mathbf{R}_0 + \Delta\mathbf{R}^*$$

$$= \mathbf{R}_0 + \arg\max_{\Delta\mathbf{R}}(\mathrm{Tr}\{[(\mathbf{I}+\mathbf{W}\mathbf{R}_0)^{-1}\mathbf{W}-(\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2]\Delta\mathbf{R}\}) \tag{7.26}$$

s.t.

$$\Delta\mathbf{R}+\mathbf{R}_0 \geq \mathbf{0}, \mathrm{Tr}(\Delta\mathbf{R}) = 0, \Delta\mathbf{R} \geq -\varepsilon\mathbf{I}$$

the step size ceiling ε is initialized as

$$\varepsilon = 1/\max\{\|(\mathbf{I}+\mathbf{W}\mathbf{R}_0)^{-1}\mathbf{W}_1\|, \|(\mathbf{I}+\mathbf{W}\mathbf{R}_0)^{-1}\mathbf{W}_2\|\} \tag{7.27}$$

By substituting $\mathbf{R}^*$ into $C_s(\mathbf{R})$, C_s^- can be obtained [35].

Upper bound of the Secrecy Capacity:

Based on the discussion in [35] the upper bound of the secrecy capacity C_s^+ can be obtained by minimizing $\mathrm{F}(\mathbf{R}^*,\mathbf{K})$ over $\mathbf{K}$ which has the form:

$$C_s^+ = \min_{\mathbf{K}} F(\mathbf{R}^*,\mathbf{K}) = \min_{\mathbf{K}} \ln\frac{|\mathbf{I}+\mathbf{K}^{-1}\mathbf{H}\mathbf{R}^*\mathbf{H}^+|}{|\mathbf{I}+\mathbf{W}_2\mathbf{R}^*|} \tag{7.28}$$

$$= \min_{\mathbf{K}}\{\ln|\mathbf{I}+\mathbf{K}^{-1}\mathbf{Q}|\} - \ln|\mathbf{I}+\mathbf{W}_2\mathbf{R}^*|$$

$$= \min_{\mathbf{K}}\{\ln|\mathbf{K}+\mathbf{Q}| - \ln|\mathbf{K}|\} - \ln|\mathbf{I}+\mathbf{W}_2\mathbf{R}^*|$$

$$= \min_{\Delta\mathbf{K}}\{\ln|\mathbf{K}_0+\Delta\mathbf{K}+\mathbf{Q}| - \ln|\mathbf{K}_0+\Delta\mathbf{K}|\} - \ln|\mathbf{I}+\mathbf{W}_2\mathbf{R}^*|$$

$$= \min_{\Delta\mathbf{K}}\left\{\ln\frac{|\mathbf{Q}+\mathbf{K}_0|}{|\mathbf{K}_0|} + \ln\frac{|(\mathbf{Q}+\mathbf{K}_0)^{-1}\Delta\mathbf{K}|}{|(\mathbf{K}_0)^{-1}\Delta\mathbf{K}|}\right\} - \ln|\mathbf{I}+\mathbf{W}_2\mathbf{R}^*| \tag{7.29}$$

where

$$\mathbf{Q} = \mathbf{H}\mathbf{R}^*\mathbf{H}^+ \tag{7.30}$$

(7.29) is equivalent to

$$\min_{\Delta\mathbf{K}} F(\mathbf{R}^*,\mathbf{K}) = \min_{\Delta\mathbf{K}}\left\{\ln\frac{|(\mathbf{Q}+\mathbf{K}_0)^{-1}\Delta\mathbf{K}|}{|(\mathbf{K}_0)^{-1}\Delta\mathbf{K}|}\right\} - \ln|\mathbf{I}+\mathbf{W}_2\mathbf{R}^*| \tag{7.31}$$

$$\approx \min_{\Delta\mathbf{K}}\{\mathrm{Tr}[((\mathbf{Q}+\mathbf{K}_0)^{-1} - (\mathbf{K}_0)^{-1})\Delta\mathbf{K}]\} - \ln|\mathbf{I}+\mathbf{W}_2\mathbf{R}^*| \tag{7.32}$$

s.t.

$$\mathbf{K}_0 + \Delta\mathbf{K} \geq \varepsilon_1\mathbf{I},\ \ \varepsilon_1 = 10^{-3}$$

$$\Delta\mathbf{K} \geq -\varepsilon_2\mathbf{I}$$

$$\varepsilon_2 = 1/\max\{\|(\mathbf{K}+\mathbf{Q})^{-1}\|, \|\mathbf{K}^{-1}\|\} \qquad (7.33)$$

Considering the numerical property of CVX, we use the constraint $\mathbf{K}_0 + \Delta\mathbf{K} \geq \varepsilon_1\mathbf{I}$ rather than $\mathbf{K}_0 + \Delta\mathbf{K} \geq \mathbf{0}$ to guarantee that $\mathbf{K}_0 + \Delta\mathbf{K}$ is a positive semidefinite matrix [35]. The backtracking line search is utilized here since we use linear approximation for this optimization problem as well.

Since $\mathbf{K}$ is introduced into the formula, such that $F(\mathbf{R},\mathbf{K})$ is convex with respect to $\mathbf{K}$ and concave with respects to $\mathbf{R}$, there is no obstacle for us to use the stopping condition (7.14) of backtracking. Since there are two variables in this optimization problem, it is demanded that the optimization needs to be processed by considering $\mathbf{K}$ and $\mathbf{R}$ jointly. However, CVX may not be able to handle it properly and oscillations on the results may occur in some cases. We will apply this algorithm on the cases where we have known the solutions to test its validity in the following part.

Notations:

C_s^+: The upper bound of secrecy capacity in (7.19) obtained by CVX.

C_s^-: The lower bound of secrecy capacity in (7.18) obtained by CVX.

C_{R}: The secrecy capacity in (7.7) obtained by CVX (with backtracking).

$\mathbf{R}_{\mathrm{Min-Max}}$: The transmit covariance matrix returned by min-max algorithm in (7.17).

$\mathbf{R}_{\mathrm{R}}$: The transmit covariance matrix of (7.7) returned by CVX (with backtracking).

Case 7-5:

$\mathbf{H}_1 = \begin{bmatrix} 1.5 & -0.5 \\ -0.5 & 1.5 \end{bmatrix}$ $\mathbf{H}_2 = \begin{bmatrix} 1.5 & 0.5 \\ 0.5 & 1.5 \end{bmatrix}$;

$\mathbf{W}_1 = \mathbf{H}_1\mathbf{H}_1^+ = \begin{bmatrix} 2.5 & -1.5 \\ -1.5 & 2.5 \end{bmatrix}$ $\mathbf{W}_2 = \mathbf{H}_2\mathbf{H}_2^+ = \begin{bmatrix} 2.5 & 1.5 \\ 1.5 & 2.5 \end{bmatrix}$; $r_+(\mathbf{W}_1 - \mathbf{W}_2) = 1$.

The optimal transmit covariance matrix $\mathbf{R}^*$ can be obtained based on (7.10), $\mathbf{R}^* = P_T\begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$.

Note that

$\mathbf{H}_1 = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}^+$ $\mathbf{H}_2 = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}^+$, where we just rotate the eigenvectors of $\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$ and $\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$ by $\frac{\pi}{2}$. In this case, we set $N_1 = N_2 = 1, N = 10$ and validate if the secrecy capacity can be obtained in $((N_1 + N_2) \cdot N)$ iterations.

By observing Tables 7.5 and 7.6, we can conclude that the Min-Max algorithm works in both high (SNR= 30 dB) and low SNR regimes (SNR= -10 dB) for this case, and that the covariance matrix $\mathbf{R}_{\text{Min-Max}}$ equals to $\mathbf{R}^*$. To verify the general validity of this algorithm, we apply it on the randomly generated channel matrices.

Table 7.5. The secrecy capacity and its lower bound, upper bound obtained by CVX with backtracking (Case 7-5, SNR= -10 dB).

iterations	t_{upper}	C_s^+	t_{lower}	C_s^-	t_{R}	C_{R}
2	0	0.3479	1	0.3479	1	0.3479
4	0	0.3479	0.125	0.3479	1	0.3479
6	0	0.3479	0.5	0.3479	0.0625	0.3479
8	0	0.3479	1	0.3479	0.0039	0.3479
10	0	0.3479	1	0.3479	0.125	0.3479
12	0	0.3479	0.0625	0.3479	0.0625	0.3479
14	0	0.3479	1	0.3479	0.0625	0.3479
16	0	0.3479	1	0.3479	0.0625	0.3479
18	0	0.3479	0	0.3479	0.0625	0.3479
20	0	0.3479	0.5	0.3479	0.0625	0.3479

$\mathbf{R}_{\text{MC}} = \begin{bmatrix} 0.05 & -0.05 \\ -0.05 & 0.05 \end{bmatrix}$, $\mathbf{R}_{\text{Min-Max}} = \begin{bmatrix} 0.05 & -0.05 \\ -0.05 & 0.05 \end{bmatrix}$, $\mathbf{R}_{\text{R}} = \begin{bmatrix} 0.05 & -0.05 \\ -0.05 & 0.05 \end{bmatrix}$; C_{MC}=0.3479.

Table 7.6. The secrecy capacity and its lower bound, upper bound obtained by CVX with backtracking (Case 7-5, SNR = 30 dB).

iterations	t_{upper}	C_s^+	t_{lower}	C_s^-	t_{R}	C_{R}
2	0.5	1.9986	0.5	0.0029	1	1.9989
4	0.001	1.9989	1	1.9989	1	1.9989
6	0	1.9989	1	1.9989	0	1.9989
8	0	1.9989	0.0078	1.9989	0	1.9989
10	0	1.9989	0.001	1.9989	0	1.9989
12	0	1.9989	0.0005	1.9989	0	1.9989
14	0	1.9989	0.0001	1.9989	0	1.9989
16	0	1.9989	0.0005	1.9989	0	1.9989
18	0	1.9989	0.0001	1.9989	0	1.9989
20	0	1.9989	0.0001	1.9989	0	1.9989

$\mathbf{R}_{\text{MC}} = \begin{bmatrix} 500 & -500 \\ -500 & 500 \end{bmatrix}$, $\mathbf{R}_{\text{Min-Max}} = \begin{bmatrix} 500 & -500 \\ -500 & 500 \end{bmatrix}$, $\mathbf{R}_{\text{R}} = \begin{bmatrix} 500 & -500 \\ -500 & 500 \end{bmatrix}$; C_{MC}=1.9989.

For the optimization problem of each group of channel matrices, we set $N_1 = N_2 = N = 10$. To make it comparable, we set the number of iterations for solving the optimization problem in (7.7) as $10 \times (10 + 10) = 200$. In order to explore the impact produced by the channel matrices, we solve the optimization problems for both high SNR and low SNR regimes. We start with the low SNR regime and the results are shown in Figures 7.7 and 7.8.

Note that if the eigenvalues of $\mathbf{W}_1 - \mathbf{W}_2$ are all non-positive, i.e. $\mathbf{W}_1 \leq \mathbf{W}_2$, the secrecy capacity is definitely zero since the eavesdropper channel is stronger than or equal to the legitimate channel at each eigen direction. To maintain the randomness of our data, we reserve these kinds of cases in the tables. By observing Tables 7.7 and 7.8, we can conclude that whether the SNR is high or low, it is difficult for the Min-Max algorithm to approach the accurate results in some extreme cases where $\lambda_+(\mathbf{W}_1 - \mathbf{W}_2) \ll |\lambda_-(\mathbf{W}_1 - \mathbf{W}_2)|$ ($\lambda_{+(-)}(\mathbf{A})$ denotes the positive (negative) eigenvalue of $\mathbf{A}$). A possible reason of this phenomenon is that since the positive eigenmode is significant weaker than the negative eigenmode of $\mathbf{W}_1 - \mathbf{W}_2$, the convergence is more sensitive to the step size, i.e. the objective value oscillates near the global optimal point. We will pick one of these cases to discuss in the next section.

Table7.7. The algorithm performance for random channel matrices (SNR $= -30$ dB).

C_S^+	C_S^-	C_{R}	C_{MC}	$\lambda(\mathbf{W}_1\text{-}\mathbf{W}_2)$	
0	-0.0018	0.0013	0.0013	-2.6208	0.8751
0	-0.0018	0.0005	0.0005	-7.2798	0.3580
0.0006	0.0006	0.0006	0.0006	0.2529	0.4446
0	-0.0006	-0.0002	0	-2.3897	-0.1527
0.0071	0.0071	0.0071	0.0071	0.4880	4.9147
0.0008	0.0008	0.0008	0.0008	-0.2747	0.5354
0	-0.0028	0	0	-8.3554	-0.0311
0	-0.0014	-0.0017	0	-3.0394	-1.1698
0.0029	0.0029	0.0029	0.0029	-0.4693	2.0354
0.0014	0.0014	0.0014	0.0014	0.3038	0.9892
0.0043	0.0043	0.0043	0.0043	-1.2853	3.0129
0	-0.0001	-0.0001	0	-1.0543	-0.0515
0.0042	0.0042	0.0042	0.0042	0.2383	2.9381
0	-0.0005	0.0048	0.0048	-4.3984	3.3461
0	-0.0005	-0.0002	0	-5.4023	-0.1354
0	-0.0003	0.0009	0.0009	-1.0231	0.6413
0	-0.0013	0.0002	0.0002	-3.9353	0.1446
0.0023	0.0023	0.0023	0.0023	-0.2864	1.6318
0.005	0.005	0.005	0.005	-1.2202	3.4997
0.0135	0.0135	0.0135	0.0135	0.1674	9.4347

Table 7.8. The algorithm performance for random channel matrices (SNR $= 30$ dB).

C_S^+	C_S^-	C_{R}	C_{MC}	$\lambda(\mathbf{W}_1\text{-}\mathbf{W}_2)$	
6.0735	6.0734	6.0311	6.0658	-3.3247	0.6178
6.8489	6.8489	6.8489	6.8475	-0.6484	2.9111
0.0113	-5.3710	-1.9852	-0.0001	-19.4026	0.0079
0	-2.0520	-1.5211	-1.4706	-8.5696	-2.1976
9.2536	9.2459	9.2536	9.2474	-0.2674	4.2893
0.8783	0.8782	-1.1889	0.8760	-7.6928	0.3695
4.0854	4.0854	4.0854	4.0811	-3.6919	0.1634
0	-3.0232	-0.5727	-0.5417	-3.5063	-1.8618
1.1236	1.1098	-0.7408	1.1189	-7.7219	0.1005
1.6185	1.6028	1.6098	1.6185	-0.6440	2.6862
1.4912	1.4912	0.8170	1.4912	-1.5857	1.2056
0	-0.6687	-1.2348	-0.5518	-2.3920	-0.1411
3.0356	3.0215	2.6705	3.0302	-2.6236	0.1528
2.0451	2.0450	1.0324	2.0449	-1.7235	2.7253
5.9152	5.9152	5.9152	5.9803	0.1478	1.4211
2.0727	2.0609	1.8363	2.0724	-1.8244	1.2820
2.0315	2.0315	2.0315	2.0293	-2.9898	0.0829
0.0174	-5.0601	-0.2223	-0.0025	-2.6918	-0.0031
5.4258	5.4258	5.4258	5.4483	0.9039	3.0010
2.4789	2.4679	2.1607	2.4710	-0.8264	1.3525

The Difficult Case:

As we observed in the last section, if $\lambda_+(\mathbf{W}_1-\mathbf{W}_2) \ll |\lambda_-(\mathbf{W}_1-\mathbf{W}_2)|$, the convergence of the secrecy capacity is usually slow, and the oscillation occurs easier in such cases (see Figures $7.7-7.9$). We select a group of channel matrices with these characteristics and try to improve its convergence and remove the oscillation.

Case 7-6:

$$\mathbf{H}_1 = \begin{bmatrix} 0.7765 & -0.3039 \\ -0.328 & -0.6409 \end{bmatrix} \mathbf{H}_2 = \begin{bmatrix} 0.5443 & -0.9261 \\ -0.1055 & -1.7141 \end{bmatrix};$$

$$\mathbf{W}_1 = \begin{bmatrix} 0.7105 & -0.0258 \\ -0.0258 & 0.5031 \end{bmatrix} \mathbf{W}_2 = \begin{bmatrix} 1.1539 & 1.53 \\ 1.53 & 2.9493 \end{bmatrix}.$$

$\lambda_-(\mathbf{W}_1-\mathbf{W}_2) = -3.295$; $\lambda_+(\mathbf{W}_1-\mathbf{W}_2) = 0.4054$; SNR $= -10$ dB.

The step size ceilings of the upper bound optimization problem (7.32) $\varepsilon_{\text{upper}}$ and lower bound optimization problem (7.26) $\varepsilon_{\text{lower}}$ are set as they are given in

(7.27) and (7.33) with a scaling factor α:

$$\varepsilon_{\text{upper}} = \alpha/\max\{\|(\mathbf{K}+\mathbf{Q})^{-1}\|, \|\mathbf{K}^{-1}\|\} \tag{7.34}$$

$$\varepsilon_{\text{lower}} = \alpha/\max\{\|(\mathbf{I}+\mathbf{W}\mathbf{R}_0)^{-1}\mathbf{W}_1\|, \|(\mathbf{I}+\mathbf{W}\mathbf{R}_0)^{-1}\mathbf{W}_2\|\} \tag{7.35}$$

α is set as 1 first and the results are shown in Table 7.9 and Figure 7.7.

$C_{\text{MC}} = 0.0558.$

Table 7.9. The secrecy capacity and its upper/lower bounds of Case 7-6 ($\alpha = 1$).

iterations	t_{upper}	C_S^+	t_{lower}	C_S^-	t_{R}	C_{R}
20	0.5	0.0497	0.25	0.0416	0	0.0558
40	1	0.0141	1	-0.0365	0	0.0558
60	1	0.0236	1	-0.1240	0	0.0558
80	1	0.0384	1	0.0382	0	0.0558
100	1	0.0846	0.5	-0.2282	0	0.0558
120	1	0.0186	1	-0.0874	0	0.0558
140	1	0.0200	1	-0.0128	0	0.0558
160	1	0.0216	1	-0.1253	0	0.0558
180	1	0.0331	1	0.0320	0	0.0558
200	1	0.0566	1	-0.2055	0	0.0558

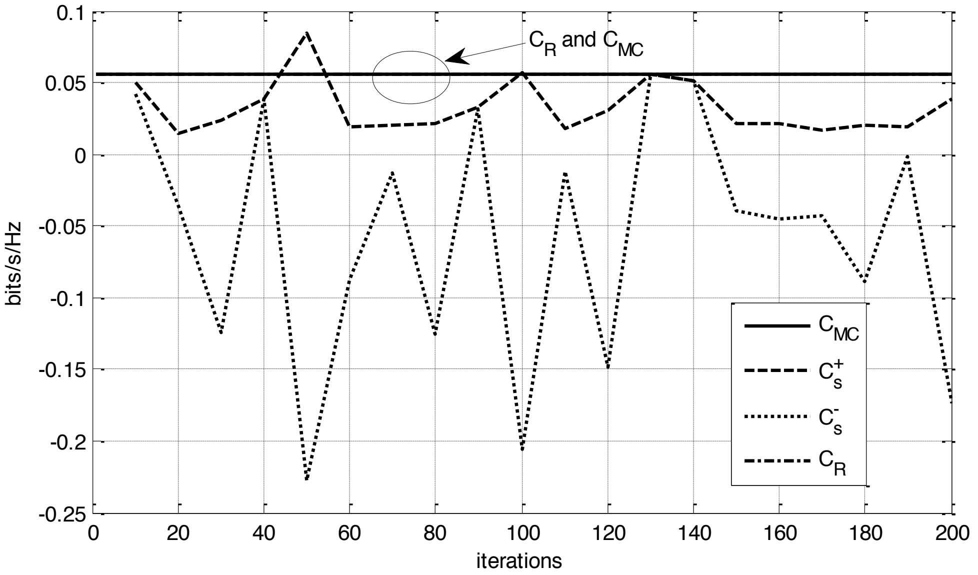


Figure 7.7. The secrecy capacity and its upper/lower bound of Case 7-6 ($\alpha = 1$).

The upper results indicate that the optimization process of problem (7.7) convergences quickly while the oscillation happens in both the optimization processes of the upper bound and the lower bound. This implies that the convergence of the upper bound and the lower bound might be more sensitive to the impact of the separate optimizations. In other word, for each iteration, the $\mathbf{R}^*$ returned by the maximization portion is just the optimal $\mathbf{R}$ for the given $\mathbf{K}$ of the last iteration. We then use this $\mathbf{R}^*$ to process the minimization portion and achieve the $\mathbf{K}^*$ of the current iteration. Next, the new $\mathbf{K}^*$ substitutes the former $\mathbf{K}$ and we use it to get the new $\mathbf{R}^*$ of the next iteration. However, the new $\mathbf{R}^*$ might be worse than the optimal $\mathbf{R}$ of the last iteration, so that the oscillations of the lower bound curve might happen. The same situation occurs on $\mathbf{K}$ as well which results in the oscillations of the upper bound curve. Therefore, we are trying to adjust the step size to inspect if the oscillations can be eliminated so that improves the convergence. The scale factor of (7.34) and (7.35) α is reset as 0.05. The N_1 and N_2 are set to be 10 while N is increased to 30. The result is shown in Figure 7.8.

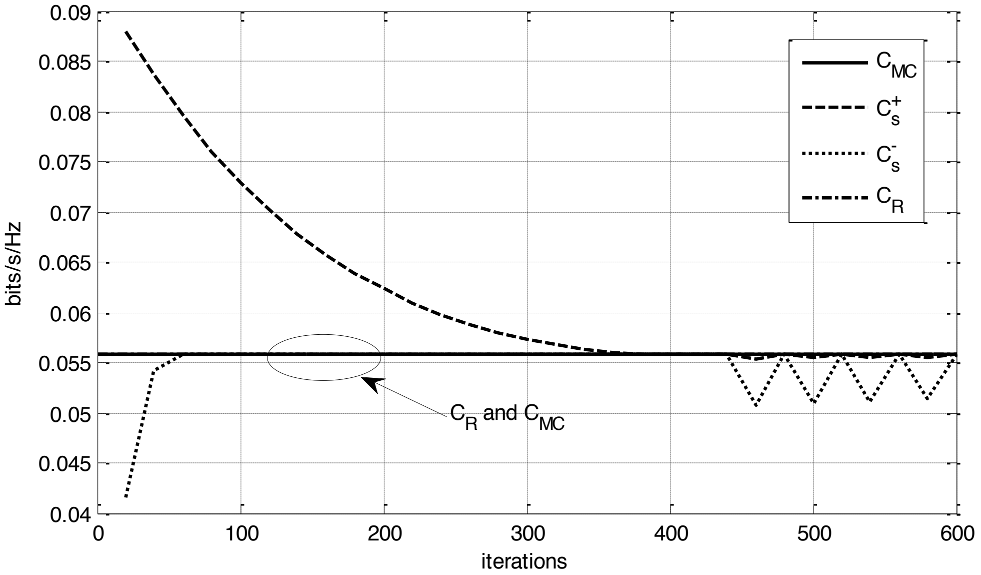


Figure 7.8. The secrecy capacity and its upper/lower bound of Case 7-6 ($\alpha = 0.05$).

Figure 7.8 indicates that the situation is improved which means that reducing the step size might work for this case. We will keep reducing the searching range and verify if it works finally. α is reset as 0.01 and we increase N to 200. The result is shown in Figure 7.9.

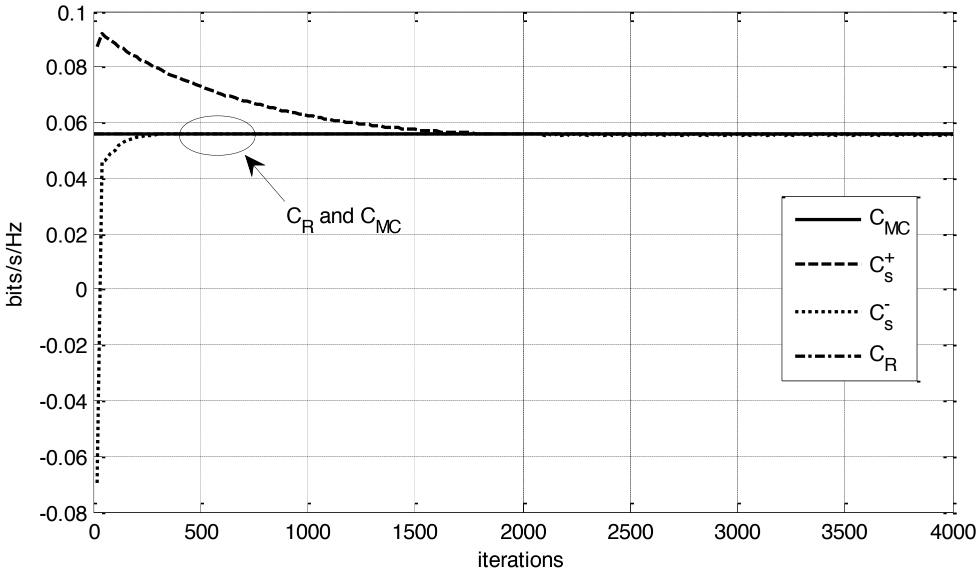


Figure 7.9. The secrecy capacity and its upper/lower bound of Case 7-6 ($\alpha = 0.01$).

By comparing the results for different step size limits, we can conclude that the oscillation introduced by the asynchronism of the optimization can be reduced by compressing the step size. However we cannot eliminate the oscillation completely and it also reduces the speed of convergence of the algorithm.

7.3 Summary

The linear approximation reformulates the secrecy capacity optimization problem to allow the latter to be handled by CVX (since CVX has no difficulty for solving the optimization problem whose objective function is linear). The optimal solution can be obtained by solving this reformulation. Backtracking line search is used to improve the convergence. This method might return a local optimal solution when $\mathbf{W}_1 \ngeq \mathbf{W}_2$ i.e. the objective is not concave. To resolve this difficulty, the Min-Max Algorithm is introduced to make the objective concave or convex with respects to the variables (**R** or **K** respectively) such that the obtained solution is globally optimal. Oscillation may appear during the convergence in some cases since CVX is not able to handle the optimization over two variables jointly.

8. Summary of the Thesis

With the widespread development and deployment of MIMO systems, which are one of the most significant evolutions in modern wireless communications, the security issue of MIMO systems is becoming more and more important. Based on information theoretic secrecy, the secrecy capacity of a Gaussian MIMO wiretap channel has been formulated as an optimization problem with respect to transmit covariance matrix while an explicit, closed-form optimal solution is not available, except for some special cases. This thesis considers the numerical methods of obtaining the optimal transmit covariance matrix of general Gaussian MIMO wiretap channel by utilizing CVX, Differential Evolution algorithm and Monte Carlo optimization.

We observed in Chapter 4 that CVX is a good simulation modelling toolbox to compute the MIMO channel capacity and corresponding optimal covariance matrix for a given channel matrix. Since the optimization problem for the secrecy capacity of wiretap MIMO channel as reviewed in Chapter 3 is not concave function unless $\mathbf{W}_1 - \mathbf{W}_2 \geq \mathbf{0}$, i.e. degraded channel, CVX cannot handle this problem directly. Based on our observations in the previous chapters, CVX is a popular tool but it is not able to solve the convex/concave optimization problems correctly in some cases.

In Chapter 5, the stochastic optimization methods for obtaining numerical results of this optimization problem were considered. We found that Monte Carlo optimization is a good algorithm to obtain the secrecy capacity approximately for the cases where the number of transmit antennas m is not too large; while for the cases where m is large, it is difficult for Monte Carlo to obtain accurate results especially when the $\mathbf{R}^*$ has low rank. We also discussed the approximated secrecy capacity formula for weak eavesdropper which can be handled by CVX. By comparing the results obtained by Monte Carlo with the results obtained by CVX, we found that CVX is able to return relatively accurate results when SNR is low.

To achieve numerical results of the optimization problem without approximation, we discussed Differential Evolution algorithm and rank-adaptive Monte Carlo. They can both improve the convergence such that the secrecy capacity and the optimal covariance matrix can be approximately obtained. Considering the processing time and the complexity of algorithm, both methods can be used to solve the optimization problem properly.

In Chapter 6, an approximation of the optimization problem for the secrecy capacity of Gaussian MIMO wiretap channel with weak eavesdropper, which can be handled by CVX, is proposed. We concluded that the CVX is not able to solve the approximated problem properly since it cannot return the correct optimal eigenvectors of the covariance matrix. An analytical solution of the approximated problem is validated in this chapter. Similar to CVX, using modeling toolbox YALMIP is discussed to solve the original optimization problem and its approximation. It is found that YALMIP is not able to solve the problems properly since it cannot guarantee to return a positive semidefinite matrix as a solution.

Based on the observations in Chapter 6, a reformulation of the optimization problem for general MIMO wiretap channels is discussed in Chapter 7. The optimization problem is approximately reformulated as an optimization problem for linear function with several constraints that can be solved by CVX properly. To improve the accuracy of the results and the efficiency, backtracking line search is considered. The reasonable results can be obtained by CVX even if the original optimization problem is not concave ($\mathbf{W}_1 \ngeq \mathbf{W}_2$). However, in the cases where $\mathbf{W}_1 \ngeq \mathbf{W}_2$, a local optimal solution might be returned by CVX instead of global optimal solution since the condition of backtracking is not defined properly. To solve this difficulty, the Min-Max Algorithm is discussed to make the objective concave or convex with respects to the variables such that the obtained solution is globally optimal. Oscillations may appear during the iterations in some cases since the CVX is not able to deal with the optimizations jointly. We found that the oscillations are affected by the choice of step size on the ascent direction. Therefore, a method for making CVX to deal with the optimizations jointly will be important and necessary in the future. Furthermore, the convergence might be improved if the step size can be chosen by the algorithm in an adaptive way.

9. References

9.1 MIMO Channel/Capacity

[1] D. Tse, P. Viswanath, *Fundamentals of Wireless Communication*, Cambridge University Press, 2005.

[2] N. Chiurtu, B. Rimoldi, E. Telatar, On the Capacity of Multi-Antenna Gaussian Channels, *IEEE Int. Symp Info. Theory* (*ISIT*), Washington, DC, USA, June. 2001, pp. 21.

[3] D. Gesbert, M. Shafi, D.Shiu, P.J. Smith, A. Naguib, From Theory to Practice: An Overview of MIMO Space-Time Coded Wireless Systems, *IEEE J. Select. Areas Commun.*, v. 21, No.3, pp. 281-301, Apr. 2003.

[4] I. E. Telatar, Capacity of Multi-Antenna Gaussian Channels, *AT&T Bell Lab. Internal Tech. Memo.*, Jun. 1995.

[5] W. Yu, W. Rhee, S. Boyd, J. M. Cioffi, Iterative Water-Filling for Gaussian Vector Multiple-Access Channels, *IEEE Trans. Info. Theory*, V. 50, N. 1, pp. 145-152, Jan. 2004.

[6] E. Visotsky, U. Madhow, Space-Time Transmit Precoding with Imperfect Feedback, *IEEE Trans. Info. Theory*, V. 47, N. 6, pp. 2632-2639, Sep. 2001.

[7] A. Narula, M. Lopez, M. Trott, F. W. Wornell, Efficient Use of Side Information in Multiple-Antenna Data Transmission over Fading Channels, *IEEE J. Select. Areas Commun.*, V. 16, N. 8, pp. 1423-2436, Oct. 1998.

[8] D. S. Shiu, G, J. Foschini, M. J. Gans, J. M. Kahn, Fading Correlation and its Effect on the Capacity of Multi-Element Antenna, *IEEE Trans. Commun.*, V. 48, N.3, pp. 502-513, Mar. 2000.

[9] Q. T. Zhang, X. W. Cui, X. M. Li, Very Tight Capacity Bounds for MIMO-Correlated Rayleigh-Fading Channels, *IEEE Trans. Wireless Commun.*, V. 4, N.2, pp. 681-888, Mar. 2005.

[10] M. R. McKay, I. B. Collings, General Capacity Bounds for Spatially Correlated Rician MIMO Channels, *IEEE Trans. Info. Theory*, V. 51, N. 9, pp. 3121-3145, Sep.

2005.

[11] A. Goldsmith, S. A. Jafar, N. Jindal, S. Vishwanath, Capacity Limits of MIMO Channels, *IEEE J. Select. Areas Commun.*, V. 21, N. 5, pp. 684-702, Jun. 2003.

[12] G. J. Foschini, Layered Space-Time Architecture for Wireless Communication in a Fading Environment when using Multiple Antennas, *Bell Lab*, *Tech.* J., V. 1, N. 2, pp. 41-59, Oct. 1996.

[13] S. A. Jafar, A. Goldsmith, Transmitter Optimization and Optimality of Beamforming for Multiple Antenna Systems, *IEEE Trans. Wireless Commun.*, V. 3, N. 4, pp. 1165-1175, Jul. 2004.

9.2 Physical-Layer Security

[14] Y. Liang, H. V. Poor, S. Shamai(Shitz), Information Theoretic Security, *Found. and Trends in Commun. and Info. Theory*, V. 5, N. 45, 2008, pp. 355-580.

[15] R. Cramer, V. Shoup, Design and Analysis of Practical Public-Key Encryption Schemes Secure Against Adaptive Chosen Ciphertext Attack, *SIAM J. Computing*, V. 33, N. 1, pp. 167–226, 2004.

[16] D. Hofheinz, E. Kiltz, Secure Hybrid Encryption from Weakened Key Encapsulation, *Crypto' 07*, *LNCS*, Santa Barbara, CA, USA, Aug 2007, pp. 553–571.

[17] A. D. Wyner, The Wiretap channel, *Bell System Tech. J.*, V. 54, N. 8, pp. 1355–1387, Oct. 1975.

[18] I. Csiszár, J. Körner, Broadcast Channels with Confidential Messages, *IEEE Trans. Info. Theory*, V. 24, N. 3, pp. 339–348, May. 1978.

[19] C. E. Shannon, Communication Theory of Secrecy Systems, *Bell System Tech. J.*, V. 28, N. 4, pp. 656–715, 1949.

[20] M. Bloch, J. Barros, Physical-Layer Security from Information Theory to Security Engineering, *Cambridge University Press*, 2011.

[21] D. Klinc, F. Ha, S. W. McLaughlin, J. Barros, B. J. Kwak, LDPC Codes for the Gaussian Wiretap Channel, *IEEE Trans. Info. Forensics Security*, V. 6, N. 3, pp. 532-540, Sep. 2011.

[22] H. Mahdavifar, A. Vardy, Achieving the Secrecy Capacity of Wiretap Channels Using Polar Codes, *IEEE Trans. Info. Theory*, V. 57, N. 10, pp. 6428-6443, Oct. 2011.

[23] H. Tyagi, A. Vardy, Explicit Capacity-Achieving Coding Scheme for the Gaussian Wiretap Channel, *IEEE Int. Symp Info. Theory* (*ISIT*), Honolulu, HI, USA, Jun. 2014, pp. 956-960.

[24] S. K. Leung-Yan-Cheong and M. E. Hellman, The Gaussian Wiretap Channel, *IEEE Trans. Info. Theory*, V. 24, N. 4, pp. 451–456, Jul. 1978.

[25] T. Liu, S. Shamai (Shitz), A Note on the Secrecy Capacity of the Multiple-Antenna Wiretap Channel, *IEEE Trans. Info. Theory*, V. 55, N. 6, pp. 2547-2553 , Jun. 2009.

[26] S. Shafiee, N. Liu, S. Ulukus, Towards the Secrecy Capacity of the Gaussian MIMO Wire-tap channel: The 2-2-1 channel, *IEEE Trans. Info. Theory*, V.55, N. 9, pp. 4033-4039, Sep. 2007.

[27] A.O.Hero, Secure Space-Time Communication, *IEEE Trans. Info. Theory*, V. 49, N. 12, pp. 3235-3249, Dec. 2003.

[28] Z. Li, W. Trappe, R. Yates, Secrecy Communication via Multi-Antenna Transmission, *Info. Sciences and Systems Conf.*, Baltimore, MD, USA, Mar. 2007, pp. 905-910.

[29] P. Parada, R. Blahut, Secrecy Capacity of SIMO and Slow Fading Channels, *IEEE Int. Symp. Info. Theory Proceedings* (*ISIT*), Adelaide, Australia, Sep. 2005, pp.2152-2155.

[30] S. Loyka, C. D. Charalambous, On Optimal Signaling over Secure MIMO Channels, *IEEE Int. Symp. Info. Theory Proceedings* (*ISIT*), Cambridge, MA, USA, Jul. 2012, pp. 443-447.

[31] J. Li, A. Petropulu, Transmitter Optimization for Achieving Secrecy Capacity in Gaussian MIMO Wiretap Channels, *IEEE Internal Conf. Acoustics Speech and Signal Processing* (*ICASSP*), Dallas, TX, USA, Mar. 2010, pp. 3362-3365.

[32] S. Loyka, C. D. Charalambous, Rank-Deficient Solutions for Optimal Signaling over Secure MIMO Channels, *IEEE Int. Symp. Info. Theory* (*ISIT*), Honolulu, HI,

USA, Jun. 2014. pp. 201-205.

[33] R. F. Schaefer, S. Loyka, The Secrecy Capacity of a Compound MIMO Gaussian Channel, *IEEE Info. Theory Workshop* (*ITW*), Sevilla, Spain, Sep. 2013, pp. 1-5.

[34] A. Khisti, G. W. Wornell, Secure Transmission with Multiple Antennas I: The MISOME Wiretap Channel, *IEEE Trans. Info. Theory*, V. 56, N. 7, pp. 3088-3104, Jul. 2010.

[35] A. Khisti, G. W. Wornell, Secure Transmission with Multiple Antennas II: The MIMOME Wiretap Channel, *IEEE Trans. Info. Theory*, V.56, N. 11, pp. 5515-5532, Nov. 2010.

[36] S. Loyka, C. D. Charalambous, Further Results on Optimal Signaling over Secure MIMO Channels, *IEEE Int. Symp Info. Theory Proceedings* (*ISIT*), Istanbul, Turkey, Jul. 2013, pp. 2019-2023.

[37] P. K. Gopala, L. Lai, H. E. Gamal, On the Secrecy Capacity of Fading Channels, *IEEE Trans. Info. Theory*, V. 54, I. 10, pp. 4687-4698, Oct. 2008.

[38] Y. Liang, H. V. Poor, S. Shamai (Shitz), Secure Communication over Fading Channels, *IEEE Trans. Info. Theory*, V. 54, N. 6, pp. 2470-2492, Jun. 2008.

[39] Z. Li, R. Yates, W. Trappe, Secret Communication with Fading Eavesdropper Channel, *IEEE Int. Symp Info. Theory Proceedings* (*ISIT*), Nice, France, Jun. 2007, pp. 1296-1300.

[40] R. Negi, S. Goel, Secret Communication using Artificial Noise, *IEEE Veh. Technol. Conf.* (*VTC*), V. 3, Sep. 2005, pp. 1906-1910.

[41] S. Goel, R. Negi, Secret Communication in Presence of Colluding Eavesdroppers, *IEEE Military Communication Conf.*, Atlantic City, NJ, USA, Oct. 2005, pp. 1501-1506.

9.3 Convex Optimization

[42] S. Boyd, L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[43] D. P. Bertsekas, A. Nedic, A. E. Ozdaglar, *Convex Analysis and Optimization*,

Belmont, MA: Athena Scientific, 2003.

[44] J. B. Hiriart-Urruty, C. Lemarechal, *Fundamentals of Convex Analysis*, New York: Springer, 2001.

[45] A. Ben-Tal, A. Nemirovski, Robust Convex Optimization, *Math. Operat. Res.*, V. 23, N.4, pp. 769–805, 1998.

[46] J. Ponstein, Seven Kinds of Convexity, *SIAM Rev.*, V.9, N.1, pp.115–119, Jan. 1967.

[47] D. G. Luenberger, Quasi-Convex Programming, *SIAM J. Applied Mathematics*, V.16, N.5, pp. 1090-1095, Sep. 1968.

[48] J. E. Kelley, The Cutting-Plane Method for Solving Convex Programs, *J. Soc. Industrial and Applied Mathematics*, V.8, N.4, pp. 708–712, 1960.

[49] J. Elzinga, T. G. Moore, A Central Cutting Plane Algorithm for the Convex Programming Problem, *Math. Programming*, V.8, pp.134–145, 1975

[50] F. Alizadeh, J.-P. A. Haeberly, M. L. Overton, Primal-Dual Interior-Point Methods for Semidefinite Programming: Convergence Rates, Stability and Numerical Results, *SIAM J. Optimiz.* V.8, N.3, pp. 746–768, 1998.

[51] C. Helmberg, F. Rendl, R. Vanderbei, H. Wolkowicz, An Interior-Point Method for Semidefinite Programming, *SIAM J. Optimiz.*, V. 6, pp. 342–361, 1996.

[52] D. Goldfarb, G. Iyengar, Robust Convex Quadratically Constrained Programs, *Math. Programming. Ser. B*, V. 97, N. 3, pp. 495–515, 2003.

[53] M. S. Lobo, L. Vandenberghe, S. Boyd, H. Lebret, Applications of Secondorder Cone Programming, *Linear Algebra and Its Applications*, V. 284, pp. 193–228, 1998.

[54] H. Hindi, S. Boyd, Analysis of Linear Systems with Saturation using Convex Optimization, *IEEE Conf. Decision and Control*, Tampa, FL, USA, Dec. 1998. pp. 903-908..

[55] C. L. Lawson, R. J. Hanson, *Solving Least Squares Problems*, Englewood Cliffs, NJ: Prentice-Hall, 1974.

[56] Z.-Q. Luo, J. F. Sturm, S. Zhang, Conic Convex Programming and Self-Dual Embedding, *Optimiz. Methods Soft.* V. 14, N. 3, pp. 169–218, 2000.

[57] Y. Zhang, On Extending Some Primal-Dual Interior-Point Algorithms from

Linear Programming to Semidefinite Programming, *SIAM J. Optimiz.*, V. 8, N. 2, pp. 365–386, 1998.

[58] Z.-Q. Luo, Applications of Convex Optimization in Signal Processing and Digital Communication, *Math. Programming Ser. B*, V. 97, pp. 177–207, 2003.

[59] H. Lebret, S. Boyd, Antenna Array Pattern Synthesis via Convex Optimization, *IEEE Trans. Signal Processing*, V. 45, N. 3, pp. 526-532, Mar. 1997.

[60] B. Wand, J. Zhang, A. H-Madsen, On the Capacity of MIMO Relay Channels, *IEEE Trans. Info. Theory*, V. 51, N. 1, pp.29-43, Jan. 2005.

[61] S. Venkatesan, S. H. Simon, R. A. Valenzuela, Capacity of a Gaussian MIMO Channel with Nonzero Mean, *IEEE Veh. Technol. Conf.* (*VTC*), Oct. 2003, pp. 1767-1771.

[62] J. Lofberg, YALMIP : A Toolbox for Modeling and Optimization in MATLAB, *IEEE Conf. Computer Aided Control Systems Design* (*CACSD*), Taipei, Taiwan, Sep. 2004, pp. 284–289.

[63] M. Grant, S. Boyd, Y. Ye, CVX: MATLAB Software for Disciplined Convex Programming. [Online]. Available: http://www.stanford.edu/~boyd/cvx/.

9.4 Monte Carlo Optimization and Differential Evolution

[64] B. H. Dickman, M. J. Gilman, Technical Note: Monte Carlo Optimization, *J. Optimiz. Theory and Applications*, V. 60, N. 1, pp. 149-157, Jan. 1989.

[65] W. Conley, *Computer Optimization Techniques*, Petrocelli Books, Princeton, NJ, USA, 1980.

[66] W. Conley, *Optimization: A Simplified Approach*, Petrocelli Books, Princeton, NJ, USA, 1981.

[67] R. Storn, K. Price, Differential Evolution – A simple and Efficient Heuristic for Global Optimization over Continuous Spaces, *J. Global Optimiz.*, V. 11, N. 4, pp. 341-359, 1997.

[68] K. V. Price, Differential Evolution: A Fast and Simple Numerical Optimizer, *North American Fuzzy Information Processing Society*, Berkeley, CA, USA, Jun. 1996.

[69] R. Storn, On the Usage of Differential Evolution for Function Optimization, *North American Fuzzy Information Processing Society*, Berkeley, CA, USA, Jun. 1996.

[70] S. Rahnamayan, H. R. Tizhoosh, M. M. A. Salama, Opposition-Based Differential Evolution, *IEEE Trans*. *Evolut*. *Computation*, V. 12, N. 1, pp. 64-79, Feb. 2008.

[71] M. M. Ali, Differential Evolution with Preferential Crossover, *European Journal of Operational Research*, V. 181, N. 3, pp. 1137-1147, Apr. 2006.

[72] A. K. Qin, V. L. Huang, P. N. Suganthan, Differential Evolution Algorithm with Strategy Adaptation for Global Numerical Optimization, *IEEE Trans*. *Evolut*. *Computation*, V. 13, N. 2, pp. 398-417, Apr. 2009.

[73] A. K. Qin, P. N. Suganthan, Self-adaptive Differential Evolution Algorithm for Numerical Optimization, *IEEE Congr*. *Evolut*. *Computation*, V. 2, Sep. 2005, pp. 1785-1791.

[74] J. Liu, J. Lampinen, A Fuzzy Adaptive Differential Evolution Algorithm, Soft. Computing—A Fusion of Foundations, *Methodologies and Applications*, V. 9, N. 6, pp. 448–462, 2005.

[75] M. G. H. Omran, A. Salman, A. P. Engelbrecht, Self-Adaptive Differential Evolution, *in Lecture Notes in Artificial Intelligence*, Berlin, Germany: Springer-Verlag, pp. 192–199, 2005.

[76] J. Brest, V. Zumer, M. S. Maucec, Self-Adaptive Differential Evolution Algorithm in Constrained Real-Parameter Optimization, *IEEE Congr. Evolut*. *Computation*, Vancouver, BC, Canada, Jul. 2006, pp. 215-222.

[77] B. V. Babu, M. M. L. Jehan, Differential Evolution for Multi-Objective Optimization, *IEEE Congr. Evolut*. *Computation*, Dec. 2003, pp. 1047-1051.

[78] J. C. Spall, *Introduction to Stochastic Search and Optimization*, *Estimation*, *Simulation*, *and Control*, Hoboken, NJ: Wiley, USA, 2003.

Appendix.

Appendix 1. Convergence being finished in one step

The condition of this situation is that the original optimization problem (7.2) and the approximated optimization problem (7.7) have the same solution. Hence we are trying to figure out which kinds of channel matrices meet this condition.

The approximated optimization problem (7.7) is equivalent to:

$$\max_{\Delta\mathbf{R}}(\mathrm{Tr}\{[(\mathbf{I}+\mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1-(\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2]\Delta\mathbf{R}\}) \qquad (\mathrm{A.1})$$

s.t.

$$\Delta\mathbf{R}+\mathbf{R}_0\geq\mathbf{0},\mathrm{Tr}(\Delta\mathbf{R})=0$$

The KKT conditions of (A.1) are given as follows. The Lagrangian of (A.1) is

$$L=-\mathrm{Tr}[(\mathbf{I}+\mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1-(\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2]\Delta\mathbf{R} \qquad (\mathrm{A.2})$$
$$+\upsilon\,\mathrm{Tr}(\Delta\mathbf{R})-\mathrm{Tr}[\mathbf{M}(\mathbf{R}_0+\Delta\mathbf{R})]$$

and we have

$$\mathbf{M}(\mathbf{R}_0+\Delta\mathbf{R})=\mathbf{0}$$
$$\upsilon\,\mathrm{Tr}(\Delta\mathbf{R})=\mathbf{0}$$
$$\mathbf{M}\geq\mathbf{0}$$
$$\frac{\partial L}{\partial\Delta\mathbf{R}}=(\mathbf{I}+\mathbf{W}_2\mathbf{R}_\mathbf{0})^{-1}\mathbf{W}_2-(\mathbf{I}+\mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1+\upsilon\mathbf{I}-\mathbf{M}=\mathbf{0} \qquad (\mathrm{A.3})$$

If $\Delta\mathbf{R}^*$ is the optimal solution→ $\mathbf{R}^*=\mathbf{R}_0+\Delta\mathbf{R}^*$, and $\Delta\mathbf{R}^*$ can be found in only one step, we have

$$\frac{\partial L}{\partial\Delta\mathbf{R}}=(\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*))^{-1}\mathbf{W}_2-(\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*))^{-1}\mathbf{W}_1 \qquad (\mathrm{A.4})$$
$$+\upsilon\mathbf{I}-\mathbf{M}=\mathbf{0}$$
$$\rightarrow\mathbf{W}_2-[\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)][\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]^{-1}\mathbf{W}_1 \qquad (\mathrm{A.5})$$
$$+\upsilon[\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)]-\mathbf{M}=\mathbf{0}$$
$$\rightarrow\mathbf{W}_2[\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]-[\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)][\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]^{-1}\mathbf{W}_1 \qquad (\mathrm{A.6})$$
$$[\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]+\upsilon[\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)][\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]-\mathbf{M}=\mathbf{0}$$

If

$$\mathbf{W}_1[\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)] = [\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]\mathbf{W}_1 \qquad \text{(A.7)}$$

$$\rightarrow \mathbf{W}_2[\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)] - [\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)]\mathbf{W}_1 \ + \upsilon[\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)][\mathbf{I}+$$

$$\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)] - \mathbf{M} = \mathbf{0} \qquad \text{(A.8)}$$

The original optimization problem (7.2) is

$$C_s(\mathbf{R}) = \max_{\mathbf{R}} \log \frac{|\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R})|}{|\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R})|} \qquad \text{(A.9)}$$

s.t.

$$(\mathbf{R}_0+\Delta\mathbf{R}) \geq \mathbf{0};\ \ \mathrm{Tr}(\mathbf{R}_0+\Delta\mathbf{R}) \leq P_T$$

The KKT conditions of (A.9) are given as follows. The Lagrangian of (A.9) is

$$L = \log|\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R})| - \log|\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R})|$$

$$+\upsilon\ \mathrm{Tr}(\Delta\mathbf{R}) - \mathrm{Tr}[\mathbf{M}(\mathbf{R}_0+\Delta\mathbf{R})] \qquad \text{(A.10)}$$

and we have

$$\mathbf{M}(\mathbf{R}_0+\Delta\mathbf{R}) = \mathbf{0}$$

$$\upsilon\ \mathrm{Tr}(\Delta\mathbf{R}) = \mathbf{0}$$

$$\mathbf{M} \geq \mathbf{0}$$

$$\frac{\partial L}{\partial \Delta\mathbf{R}} = (\mathbf{I}+\mathbf{W}_2\mathbf{R}_0)^{-1}\mathbf{W}_2 - (\mathbf{I}+\mathbf{W}_1\mathbf{R}_0)^{-1}\mathbf{W}_1 + \upsilon\mathbf{I} - \mathbf{M} = \mathbf{0} \qquad \text{(A.11)}$$

If

$$\mathbf{W}_1[\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)] = [\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]\mathbf{W}_1 \qquad \text{(A.12)}$$

$$\rightarrow \mathbf{W}_2[\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)] - [\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)]\mathbf{W}_1$$

$$+ \upsilon[\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)][\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)] - \mathbf{M} = \mathbf{0}$$

we can observe that the KKT conditions of both optimization problems (A.2) and (A.10) have the same patterns.

Note that the condition of

$$\mathbf{W}_1[\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)] = [\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]\mathbf{W}_1$$

is that $\mathbf{W}_1$ and $\mathbf{R}^*$ have the same eigenvectors, which means that $\mathbf{W}_1$ and $\mathbf{W}_2$ have the same eigenvectors. We assume that $\lambda_1(\mathbf{W}_1)$ is the eigenvalue of $\mathbf{W}_1$ and $\lambda_1(\mathbf{W}_2)$ is the eigenvalue of $\mathbf{W}_2$, and $\lambda_1(\mathbf{W}_1) > \lambda_1(\mathbf{W}_2)$.

Based on (A.12), we have

$$\mathbf{W}_2[\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]-[\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)]\mathbf{W}_1$$
$$+\upsilon[\mathbf{I}+\mathbf{W}_2(\mathbf{R}_0+\Delta\mathbf{R}^*)][\mathbf{I}+\mathbf{W}_1(\mathbf{R}_0+\Delta\mathbf{R}^*)]-\mathbf{M}=\mathbf{0}$$
$$\rightarrow \mathbf{W}_2-\mathbf{W}_1+\upsilon[\mathbf{I}+\mathbf{W}_1\mathbf{R}^*][\mathbf{I}+\mathbf{W}_2\mathbf{R}^*]-\mathbf{M}=\mathbf{0}$$
$$\rightarrow (\mathbf{W}_2-\mathbf{W}_1)\mathbf{R}^*+\upsilon[\mathbf{I}+\mathbf{W}_1\mathbf{R}^*][\mathbf{I}+\mathbf{W}_2\mathbf{R}^*]\mathbf{R}^*-\mathbf{M}\mathbf{R}^*=\mathbf{0}$$
$$\rightarrow (\mathbf{W}_2-\mathbf{W}_1)\mathbf{R}^*+\upsilon[\mathbf{I}+\mathbf{W}_1\mathbf{R}^*+\mathbf{W}_2\mathbf{R}^*+\mathbf{W}_2\mathbf{R}^*\mathbf{W}_1\mathbf{R}^*]-\mathbf{M}\mathbf{R}^*=\mathbf{0}$$

If $r_+(\mathbf{W}_1-\mathbf{W}_2)=1$, i.e. $\mathbf{R}^*$ is rank-one matrix [30] and $\mathbf{W}_1$ has the same eigenvectors as $\mathbf{W}_2$, it can be derived that:

$$\upsilon=\frac{\lambda_1(\mathbf{W}_2)-\lambda_1(\mathbf{W}_1)}{1+\lambda_1(\mathbf{W}_1)\cdot P_T+\lambda_1(\mathbf{W}_2)\cdot P_T+\lambda_1(\mathbf{W}_1)\cdot\lambda_1(\mathbf{W}_2)P_T^2}$$

so that the KKT conditions of (A.1) and (A.9) can be solved. The conditions of the convergence being finished in one step are $r_+(\mathbf{W}_1-\mathbf{W}_2)=1$ and $\mathbf{W}_1$ has the same eigenvectors as $\mathbf{W}_2$.

Appendix 2. Some MATLAB Codes

CVX

```
% SNRdB - SNR in dB, scalar integer from -30 to 30
% m - the number of transmit antennas
% W1 - main channel matrix
% W2 – eavesdropper’s channel matrix

C_CVX = zeros(1,length(SNRdB)); % Capacity returned by CVX
C_PT = zeros(1,length(SNRdB)); % Capacity returned by substituting R/Tr(T)
                                  into (3.9)

for j=1:length(SNRdB);
   rho=10^(SNRdB(j)/10); % SNR in linear domain

% cvx part

   cvx_begin; % CVX begins
     cvx_precision(10^-4); % set CVX precision
     variable R(m,m) symmetric; % define variable (covariance matrix)
      R == semidefinite(m); % R has to be positive semidefinite
      C= log_det(eye(m)+W1*R)-trace(W2*R);
    maximize C;
     0<=trace(R)<= m*rho; % Tr(R) is less than total transmit power
   cvx_end; % CVX ends
C_CVX(j)=cvx_optval; % save the capacity

% scaled R part
Rcvx = m*rho*R/trace(R); % scaling normalized R (covariance matrix
returned by CVX
C_PT(j) = log(det(eye(m)+W1*Rcvx))-log(det(eye(m)+W2*Rcvx));
end
```

YALMIP

```
Ct=zeros(1,length(SNRdB));% for saving the capacity by using true formula
as objective
R_true(:,:,length(SNRdB))=zeros(m);% for saving the optimal covariance
matrix by using true formula as objective
Ca=zeros(1,length(SNRdB));% for saving the capacity by using approximated
formula as objective
R_tr(:,:,length(SNRdB))=zeros(m);% for saving the optimal covariance
matrix by using approximated formula as objective
```

```
for i=length(SNRdB)
    rho=10^(SNRdB(i)/10);

    % using YAMLIP for log-log
    R=sdpvar(2,2,'hermitian');%variable
    F1=set(trace(R)<=rho);
    F1=F1+set(trace(R)>=0);
    F1=F1+set(eig(R)>=0);% eigenvalues of R are greater than 0 or
F=set(R>=0), R has to be positive semidefinite
    C= log(det(eye(2)+W1*(R)))-log(det(eye(2)+W2*R));
    solvesdp(F1,-C);
    Ct(i)=double(C);
    R_true(:,:,i)=double(R);

   % using YAMLIP log-trace
    Ra=sdpvar(2,2,'hermitian');%variable
    F2=set(trace(Ra)<=rho);
    F2=F2+set(trace(Ra)>=0);
    F2=F2+set(eig(Ra)>=0);% or set(Ra>=0);
    C2=log(det(eye(2)+W1*Ra))-trace(W2*Ra);
    solvesdp(F2,-C2);
    Ca(i)=double(C2);
    R_tr(:,:,i)=double(Ra);
end
```

Regular Monte Carlo

```
% SNRdB ¨C SNR in dB
% M ¨C scalar integer from 1 to 50, the number of iterations
% N=10^5 - the number of trials of Monte Carlo
% W1 - C main channel matrix
% W2 - C eavesdropper’ s channel matrix
% a - W2 = aW1
% m - the number of transmit antennas

rho = 10^(SNRdB/10); % SNR in linear domain
[m,m] = size(W41);
trials = 1:1:N;
C = zeros(1,length(trials)); % vector for saving the final capacity vs
                              trials
SD = zeros(1,length(trials)); %standard deviation
CS = zeros(length(M),length(trials));% save randomly generated cpacity
                                     for computing standard deviation
```

```
R = zeros(m); % optimal covariance matrix


for i=1:length(M);
       A = randn(m,m);
       Rt = (A'*A)/trace(A'*A);
       CM = zeros(1,length(trials1));  % vector for saving the capacity
in each iteration
       R1M = zeros(m); % for saving optimal covariance matrix in each
iteration
       R1M = Rt;
       CM(1) =
log2(det(eye(m)+rho*W1*Rt))-log2(det(eye(m)+rho*W2*Rt));
       CS(i,1) = CM(1);
      for j = 1:length(trials)-1
         A = randn(m,m);
         Rt = (A'*A)/trace(A'*A);
         Ct = log2(det(eye(m)+rho*W1*Rt))-log2(det(eye(m)+rho*W2*Rt));
        if Ct > CM(j);%
           CM(j+1) = Ct;
           R1M = Rt;
        else
           CM(j+1) = CM(j);
           R1M = R1M;
        end
          CS(i,j+1) = CM(j+1);
      end
      C = C+CM;
      R = R+R1M;

end
C = C/length(M); % do the average
R = R/length(M);

%computation of Standard deviation
for k=1:length(trials)
      sum=0;
   for n=1:length(M)
       sum=sum+(C(k)-CS(n,k))^2;
   end
   SD(k)=sqrt(sum/length(M));
end
```

Monte Carlo (Rank-Adaptive)

```
C = zeros(1,length(trials));
R = zeros(m);
Cr(:,:,m) = zeros(1,length(trials)); % the vector for saving the
                                       capacities yielded by different
                                       covariance matrices (from rank 1
                                       to rank m) vs trials
Rr(:,:,m) = zeros(m);

for r = 1:m % from rank 1 to rank m
   for i = 1:length(M);

       Ar = randn(r,m);
       R =((Ar'*Ar)/trace(Ar'*Ar)); % generating rank r matrix randomly
       CrM = zeros(1,length(trials)); % saving the capacities yielded by
                                        rank r covariance matrices vs
                                        trials
       RrM = zeros(m);
       RrM = R;
       CrM(1) = log2(det(eye(m)+rho*W1*R))-log2(det(eye(m)+rho*W2*R));
      for j = 1:length(trials)-1
          Ar = randn(r,m);
          R = ((Ar'*Ar)/trace(Ar'*Ar));
          Ct = log2(det(eye(m)+rho*W1*R))-log2(det(eye(m)+rho*W2*R));
        if Ct > CrM(j);% if the new generated covariance yields better
                         capacity than the previous one, the previous
                         capacity will be swapped by the new one
         CrM(j+1) = Ct;
         RrM = R;
        else
           CrM(j+1) = CrM(j);
           RrM = RrM;
        end
      end
      Cr(:,:,r) = Cr(:,:,r)+CrM; % the r_th vector in Cr is used for saving
                                   the capacitied yielded by rank r
                                   covariance matrix
      Rr(:,:,r) = Rr(:,:,r)+RrM;
   end
 Cr(:,:,r) = Cr(:,:,r)/length(M); % do the average
 Rr(:,:,r) = Rr(:,:,r)/length(M);
end

 % finding the optimal capacity
```

```
for k = 1:length(trials);
   C(k) = max(Cr(1,k,:));
end

% finding the optimal covariance matrix
 R = Rr(:,:,1);
for r = 1:nt1-1;
   if Cr(1,length(trials),r+1) > Cr(1,length(trials),r);
      R = Rr(:,:,r+1);
   else
      R = R;
   end
end
```

Differential Evoluiton

```
CR=0.9;%crosssover constant
F=1.5;
SNRdB=-20:20:20;
M=1:1:25;
a=0.1;
rho=10^(SNRdB(3)/10);


num_NP = 1:1:200; % number of population
num_Gen = 1:1:1000; % total number of generations
CE = zeros(1,length(num_Gen));% optimal capacities vs number of
generations
RE(:,:,length(num_Gen)) = zeros(m);

for p = 1:length(M);

    HE(:,:,length(num_NP)) = zeros(nt1);%target H
    CEM = zeros(1,length(num_Gen));% Best Capacity of each generation in
                                 each iteration
    REM(:,:,length(num_Gen)) = zeros(nt1);% Best R of each generation in
                                        each iteration
    CE = zeros(1,length(num_NP));% all Capacities of each generation
    HEbest(:,:,length(num_Gen)) = zeros(nt1);% Best H of each generation
    VE(:,:,length(num_NP)) = zeros(nt1);%Mutant H
    UE(:,:,length(num_NP)) = zeros(nt1);%Trial H

% generate NP target matrices
   for i = 1:length(num_NP);
```

```
      HE(:,:,i) = randn(m);
  end
    RE = (HE(:,:,1)'*HE(:,:,1))/trace(HE(:,:,1)'*HE(:,:,1));
%  Capacity of first Generation
  Cs = log2(det(eye(m)+rho*W1*RE))-log2(det(eye(m)+rho*W2*RE));
  HEbest(:,:,1) = HE(:,:,1);
  Score = Cs;
 for i = 2:length(num_NP); % selecting the best target of first generation
      RE = (HE(:,:,i)'*HE(:,:,i))/trace(HE(:,:,i)'*HE(:,:,i));
      Cs = log2(det(eye(m)+rho*W1*RE))-log2(det(eye(m)+rho*W2*RE));
   if Cs > Score
       Score = Cs;
       HEbest(:,:,1) = HE(:,:,i);
   else
       Score = Score;
       HEbest(:,:,1) = HEbest(:,:,1);
   end
 end
 CEM(1) = Score;
 REM(:,:,1) =
(HEbest(:,:,1)'*HEbest(:,:,1))/trace(HEbest(:,:,1)'*HEbest(:,:,1));

 for i = 1:length(num_NP); % computing all capacities yielded by target
                             matrices of first generation
      RE = (HE(:,:,i)'*HE(:,:,i))/trace(HE(:,:,i)'*HE(:,:,i));
      CE(i) =
log2(det(eye(m)+rho*W1*RE))-log2(det(eye(m)+rho*W2*RE));
 end

 for i = 2:length(num_Gen)
 %Generate NP mutant matrixs
   for j = 1:length(num_NP)
      i_1 = randi(length(num_NP));
      j_1 = randi(length(num_NP));
      k_1 = randi(length(num_NP));
     while((i_1 == j_1) || (k_1 == j_1) || (k_1 == i_1))
          i_1 = randi(length(num_NP));
          j_1 = randi(length(num_NP));
          k_1 = randi(length(num_NP));
     end
     VE(:,:,j) = HE(:,:,i_1)+F*(HE(:,:,j_1)-HE(:,:,k_1));
   end

   %Generate NP trial matrixs
  for k = 1:length(num_NP)
       row = randi(m);
       col = randi(m);
       Ut = HE(:,:,k);%Ut and Vt are intermediate variables
       Vt = VE(:,:,k);
    for r = 1:m
      for c = 1:nt1
        if rand(1) <= CR
           Ut(r,c) = Vt(r,c);
        else
           Ut(r,c) = Ut(r,c);
        end
      end
    end

 Ut(row,col) = Vt(row,col);
 UE(:,:,k) = Ut;
 Ru = (UE(:,:,k)'*UE(:,:,k))/trace(UE(:,:,k)'*UE(:,:,k)); %trial
                                                            covariance
                                                            matrix
 % comparaing trial matrix U with target matrix H, if U yields better
  capacity than H, swap H by U
 Cu =log2(det(eye(m)+rho*W1*Ru))-log2(det(eye(m)+rho*W2*Ru));
   if Cu > CE(k)
      CE(k) = Cu;
      HE(:,:,k) = UE(:,:,k);
   else
      CE(k) = CE(k);
      HE(:,:,k) = HE(:,:,k);
   end
 end
% Finding the optimal Solution of current Generation
 CEM(i) = Score;
 REM(:,:,i) = REM(:,:,1);

  for g = 1:length(num_NP)
     if CE(g)> = CEM(i)
        CEM(i) = CE(g);
        REM(:,:,i) =
(HE(:,:,g)'*HE(:,:,g))/trace(HE(:,:,g)'*HE(:,:,g));
     else
       CEM(i) = CEM(i);
       REM(:,:,i) = REM(:,:,i);
     end
   end
 end
 CE = CE+CEM;
 RE = RE+REM;
end
CE = CE./length(M);
RE = RE./length(M);
```

Analytical Solution for Weak Eavesdropper

```
SNRdB=-20:1:40;
C=zeros(1,length(SNRdB));
% the threshold of transmit power - Pstar, there is no Pstar W2 is singular
if det(W2) ~= 0
    Pinner=(eye(m)-(W2^0.5)*(W1^(-1))*(W2^0.5));
    [dp,vp]=eig(Pinner);
    vp=vp.*(vp>0);
    Pinner=dp*vp*dp';
    Pstar=trace((W2^-1)*Pinner);
    Pstar=10*log10(Pstar);
end

for i=1:length(SNRdB)
    rho=10^(SNRdB(i)/10);
    lmax=m/rho;% the upper bound of Lagrangian multiplier lambda
    lmin=0;% the lower bound of Lagrangian multiplier lambda
   for j=1:10^4 % large enough to ensure the accuracy is high
       % bisection part, we deal with the case where m = 2
     if det(W2) == 0  % there is no Pstar when W2 is singular
         lam=(lmax+lmin)/2;% Lagrangian multiplier lambda
     else
        if rho>=10^(Pstar/10)
          lam=0;
        else
          lam=(lmax+lmin)/2;%lambda
        end
     end
    A=lam*eye(m)+W2;
    W1t=A^(-0.5)*W1*A^(-0.5);% W1~
    [d1,v1]=eig(W1t);
    if v1(1,1)>1
        ir1=1-1/v1(1,1);% first entry of diagonal matrix 'capital lambda'
    else
        ir1=0;
    end
    if v1(2,2)>1
        ir2=1-1/v1(2,2);% second entry of diagonal matrix 'capital
lambda'
    else
        ir2=0;
    end
    lamRt=[ir1,0;0,ir2];%lmabda of R~
    Rt=d1*lamRt*d1';%R~
    R=A^(-0.5)*Rt*A^(-0.5);% optimal covariance matrix

    if abs((trace(R)-rho))/rho<=0.001;
        break
    end
    % bisection
    if trace(R)>rho
       lmin=lam;
    else
       lmax=lam;
    end
  end
C(i)=[log(det(eye(m)+W1*R))-trace(W2*R)];% capacity yielded by final R
returned by above discussion
end
```

Linear Approximation and Backtracking

```
SNRdB
N=1:10;% number of total trials
T=zeros(1,length(N)); % for saving the last t of each trial
n=zeros(1,length(N)); % for saving the last n of each trial
rho=10^(SNRdB/10);
C=zeros(1,length(N));% for saving the capacity of each trial
R(:,:,length(N))=zeros(m); % for saving the optimal R of each trialn
R0=(eye(m)/m)*rho; % initialized covariance matrix

I = eye(m);
for i=1:length(N)
   Z=((eye(m)+W1*R0)^-1)*W1-((eye(m)+W2*R0)^-1)*W2;
   e = 1/max(norm(I/(I+W1*R0)*W1), norm(I/(I+W2*R0)*W2));% initialized
                                                           step size

   cvx_begin quiet SDP;
     cvx_precision(10^-16);
     variable dR(m,m) symmetric;% delta R
     dC=trace(Z*dR);% delta C
     maximize dC ;
     trace(dR)==0;
     dR+R0==semidefinite(m);
     dR+e*eye(m)==semidefinite(m);
   cvx_end;
% backtracking part
   t=1;
   n=0;
   alpha=1/4; % alpha is from 0 to 1/2
  C1=log2(det(eye(m)+W1*(R0+t*dR)))-log2(det(eye(m)+W2*(R0+t*dR))); %
  substituting R0+tdR into the true formula
  C2=log2(det(eye(m)+W1*(R0)))-log2(det(eye(m)+W2*(R0)))+ alpha
  *t*trace((((eye(m)+W1*R0)^-1)*W1-((eye(m)+W2*R0)^-1)*W2)*dR); %
  substituting t, dR and alfa into the linear approximation

   [d,v]=eig(R0+t*dR);%test if R0+tdR is PSD
     psd=0; %if psd = 1 later, means that R0+dR is not PSD
     for l=1:m;
        if v(l,l)<0;
         psd=1;
        end
     end

 while (C1<C2)||(psd==1);
        n=n+1;
        t=t/2;
       % to make sure that R0+tdR is positive definite
       [d,v]=eig(R0+t*dR);
       psd=0; %if psd = 1 later, means that R0+tdR is not PSD
    for l=1:m; %test if R0+tdR is PSD
       if v(l,l)<0;
           psd=1;
       end
    end


   C1=[log2(det(eye(m)+W1*(R0+t*dR)))-log2(det(eye(m)+W2*(R0+t*dR)))
   ];%  if R0+tdR is PSD, then renew C1 and C2 and continue the
   backtracking
   C2=[log2(det(eye(m)+W1*(R0)))-log2(det(eye(m)+W2*(R0)))+
   alfa*t*trace((((eye(m)+W1*R0)^-1)*W1-((eye(m)+W2*R0)^-1)*W2)*dR)]
   ;
 end
  T(i)=t;
  n(i)=n;
  C(i)=log2(det(eye(m)+W1*(R0+t*dR)))-log2(det(eye(m)+W2*(R0+t*dR)));
  R0=R0+t*dR;% optimal R of current iteration
  R(:,:,i)=R0;
end
```

Min-Max Algorithm

```
SNRdB - SNR in dB
rho = 10^(SNRdB/10);
H1 % main channel matrix
H2 % eavesdropper' s channel matrix
H = [H1; H2];
W1 = H1'*H1;
W2 = H2'*H2;
I = eye(m);
Rmin_max = zeros(m);
alpha = 1/4;
Nout = 1:200; % total number of iterations
NC = 1:10;%max C over R/lower bound
NF = 1:10;%min F over K/upper bound
Nn1 = zeros(1,length(Nout));%number of iterration of backtracking of
                            Maxmize C over R;we only keep the t of last
                            internal iteration of each external iteration
Nn2 = zeros(1,length(Nout));%number of iterration of backtracking of
                            Minimize F over K
T1 = zeros(1,length(Nout));%t of iterration of backtracking of Maxmize
                           C over R;we only keep the t of last internal
                           iteration of each external iteration
T2 = zeros(1,length(Nout));%t of iterration of backtracking of Minimize
                           F over K

F_upper = zeros(length(Nout), 1); % for saving upper bound od capacity
C_lower = zeros(length(Nout), 1); % for saving lower bound of capacity
[row col] = size(H);
K = eye(row);% K is initialized by identity
R = I/m*rho;
% Outer loop
for j = 1:length(Nout);
     W = H'*(K\H);
   for i = 1:length(NC); % max C over R for NC iterations
       Z = (I+W*R)\W - (I+W2*R)\W2;
       eps = 0.01/max(norm((I+W*R)\W), norm((I+W2*R)\W2));% step size

       cvx_begin quiet SDP
       variable dR(m, m) symmetric
       dC = trace(Z*dR);
       maximize dC
       subject to
          dR + eps*I >= 0;
          R + dR >=0;
          trace(dR) == 0;
   cvx_end
   t = 1;
   n1 = 0;
   R_left = R+t*dR;
   % backtracking part
   C1 = log(det(I+W*R_left))-log(det(I+W2*R_left));
   C2  = log(det(I+W*R))-log(det(I+W2*R))+alpha*t*trace(Z*dR);

  [d,v]=eig(R_left);%test if R0+tdR is PSD
   psd=0; %if psd = 1 later, means that R0+dR is not PSD
  for l=1:m;
    if v(l,l)<0;
     psd=1;
    end
  end

   while (C1<C2)||(psd==1);
        t = t/2;
        n1 = n1+1;
        R_left = R+t*dR;
    [d,v]=eig(R_left);%test if R0+tdR is PSD
   psd=0; %if psd = 1 later, means that R0+dR is not PSD
     for l=1:m;
       if v(l,l)<0;
        psd=1;
       end
     end
        C1 = log(det(I+W*R_left))-log(det(I+W2*R_left));
        C2  = log(det(I+W*R))-log(det(I+W2*R))+
              alpha*t*trace(Z*dR);
   end
   R = R+t*dR;
end
Nn1(j) = n1; % n of lower bound
T1(j) = t;% t of  lower bound
C_lower(j) = log2(det(I+W1*R))-log2(det(I+W2*R));
```

```
% min F over K ,upper bound of capacity
 for i = 1:length(NF);
     Q = H*R*H';
     Z2 = inv(K)-inv(K+Q);
     eps = 0.01/max(norm( inv(K+Q)), norm(inv(K))); % step size
    cvx_begin quiet SDP
      variable dK(row, row) symmetric;
      dF = trace(Z2*dK);
      maximize dF
      subject to
         K+dK >= (10^-3)*eye(row);
         dK + eps*eye(row) >= 0;
         dK(1:row/2, 1:row/2)==zeros(m, m); % constraint of K
         dK((row/2+1):row, (row/2+1):row)==zeros(m,m)    ;
   cvx_end
     t2 = 1;
     n2 = 0;
     K_modified = K+t2*dK;
     F1 = log(det(K_modified))-log(det(K_modified+Q));
     F2 = log(det(K))-log(det(K+Q))+alpha*t2*trace(Z2*dK);
     [d,v]=eig(K_modified);%test if R K_modified is PSD
      psd=0; %if psd = 1 later, means that R0+dR is not PSD
     for l=1:m;
        if v(l,l)<0;
         psd=1;
        end
     end

  while (F1 < F2)||(psd==1);
        t2 = t2/2;
        n2 = n2+1;
        K_modified = K+t2*dK;
     [d,v]=eig(K_modified);%test if R K_modified is PSD
      psd=0; %if psd = 1 later, means that R0+dR is not PSD
     for l=1:m;
        if v(l,l)<0;
         psd=1;
        end
     end
        F1 = log(det(K_modified))-log(det(K_modified+Q));
        F2 = log(det(K))-log(det(K+Q))+alpha*t2*trace(Z2*dK);
  end

 end
```

```
   K = K+t2*dK;
   Nn2(j)=n2;
   T2(j)=t2;%t of upper bound
   F_upper(j) = log2(det(eye(row)+K\H*R*H'))- log2(det(eye(m)+W2*R));

end
Rmin_max=R; %optimal covariance matrix
```