Capacity Simulations and Computations of Non-Coherent Fading Channels

Chien-Sheng Lai

Advisor: Prof. Dr. Stefan M. Moser
National Chiao Tung University, Taiwan

Co-Advisor: Prof. Dr. Thomas Eriksson
Chalmers University of Technology, Sweden
1 Introduction

The demand of new wireless communication systems with much higher data rates that allow, e.g., mobile wireless broadband Internet connections inspires a quick advance in wireless transmission technology. So far most systems rely on an approach where the channel state is measured with the help of regularly transmitted training sequences. The detection of the transmitted data is then done under the assumption of perfect knowledge of the channel state. This approach will not be sufficient anymore for very high data rate systems since the loss of bandwidth due to the training sequences is too large. Therefore, the research interest on joint estimation and detection schemes has been increased considerably.

Apart from potentially higher data rates a further advantage of such a system is that it allows for a fair analysis of the theoretical upper limit, the so-called channel capacity. “Fair” is used here in the sense that the capacity analysis does not ignore the estimation part of the system, i.e., it takes into account the need of the receiver to gain some knowledge about the channel state without restricting it to assume any particular form (particularly, this approach does also include the approach with training sequences!). The capacity of such a joint estimation and detection scheme is often also known as non-coherent capacity.

Recent studies investigating the non-coherent capacity of fading channels have shown very unexpected results. In stark contrast to the capacity with perfect channel knowledge at the receiver, it has been shown that non-coherent fading channels...
become very power-inefficient in the high signal-to-noise (SNR) regime\textsuperscript{1} in the sense that increasing the transmission rate by an additional bit requires squaring the necessary power. Since transmission in such a regime will be highly inefficient, it is crucial to better understand this behavior and to be able to give an estimation as to where the inefficient regime starts. One parameter that provides a good approximation to such a border between the power-efficient low-SNR and the power-inefficient high-SNR regime is the so-called fading number which is defined as the second term in the high-SNR asymptotic expansion of channel capacity. This parameter has been computed for various cases.

In the non-asymptotic situation, however, the exact channel capacity is not known. There only exist upper and lower bounds. The goal of this project is to improve the existing non-asymptotic lower bounds using numerical algorithms. In particular the focus lies on the Blahut-Arimoto algorithm and the cutting-plane algorithm that has recently been proposed by Huang and Meyn.

2 Tasks

The following tasks may be helpful for your work. Since it cannot be foreseen how the project will evolve, you will not be evaluated exclusively on the fulfillment of these tasks, but more on the creativity that you exhibit.

1. Study the existing non-asymptotic bounds on Gaussian fading channels \cite{1, 2, 3, 4}.

2. Study the Blahut-Arimoto algorithm (make a Google-search for literature!).

3. Study the cutting-plane algorithm \cite{5}.

4. Implement both algorithms to perform some numerical bounds on the non-asymptotic capacity of non-coherent Gaussian fading channels.

5. The program code should be well documented so that it can be used in future projects of similar nature.

Depending on how the project will evolve we might add to or change some of the details of these tasks.

3 General Regulations

The project will be supervised by Prof. Dr. Stefan Moser. You will have to hand in a report (an original and a copy, both signed) that is typeset in \LaTeX. The original as well as the copy are property of the laboratory. Further copies will probably be requested by the co-advisor at Chalmers University, Sweden. At the end of the project the student is asked to present his results during an English oral presentation of 25 minutes.

Furthermore, a copy of the source files of the report as well as of the program code need to be handed in.

Computer equipment and a working place will be provided by the lab.

Further details and regulations will be discussed orally.

\textsuperscript{1}Depending on the channel in use high SNR might start between 30 to 80 dB.
Dates

**Beginning**: Monday, September 11, 2006  
**End**: Friday, March 30, 2007, 12:00 PM  
**Holidays**: February 11–24, 2007 (Chinese New Year)

Addresses

Prof. Dr. Stefan M. Moser  
Engineering Building IV, Room 727  
<stefan.moser@ieee.org>  
+886-3-571 21 21 ext. 54548

Information Theory Lab  
Engineering Building IV, Room 711

Hsinchu, May 3, 2006

Prof. Dr. Stefan M. Moser

References


Acknowledgments

I would like to thank my advisor, Prof. Stefan M. Moser, for providing me this project and offer. He provided me a great place, the Information Theory Lab, NCTU and all facilities I need, including an iMac G5, which is the most lovely and the coolest machine I have ever used, and also printers, etc. With his valuable help and instructive suggestion, I have always been able to find some way to solve problems in math and in programming. I also would like to thank him for his patience when I sometimes had difficulties following his brilliant ideas so that he had to slow down, explain once again, and wait for me to digest these ideas.

I am also very grateful to my co-advisor, Prof. Thomas Eriksson, in Chalmers University of Technology, Sweden, for his constant attention and concern. He has been in touch with me during the whole project and taken care of everything for me in Chalmers concerning this thesis project.

Next, I would like to thank my classmates and friends and those who have helped me in anything concerning this project.

Finally, I would like to thank my family for their support during my whole education career since my childhood.

Hsinchu, February 27, 2006

Chien-Sheng Lai
Abstract

In this report, we study the non-coherent capacity of SISO zero-mean Gaussian fading channels with memory. Because the non-coherent capacity in asymptotic form (in the SNR) is already well understood and so far there is not yet an exact expression for the non-asymptotic capacity, we focus on deriving upper and lower bounds on the non-asymptotic capacity.

First, we use a prediction technique on the fading channel to get rid of the channel memory and finally derive a lower bound on the non-asymptotic capacity. Second, we implement the Blahut-Arimoto algorithm to perform the optimization in the expression of the derived lower bound. Third, we present a numerically computed lower bound, and compare the new numerical lower bound with analytical results.

On the other hand, we also derive an upper bound on the non-asymptotic capacity, and then implement the cutting-plane algorithm to perform the optimization in the expression of the derived upper bound. Therefore, we also present a numerically computed upper bound, and finally compare the new numerical upper bound with analytical results.

By comparison, the two new numerical bounds are found to be better (closer) than the known analytical results. The characteristics of the input distributions that achieve the new numerical bounds are presented as well.

Unfortunately, we observe numerical problems which result from the limitation of numerical methods. The new numerical bounds, especially the new upper bound, stop growing in the SNR at high SNR regime. We believe this is because we use a finite alphabet size for the channel inputs and outputs to simulate the infinite alphabet size of the channel inputs and outputs.
# Contents

1 Introduction 1

2 Channel Model 5
   2.1 Channel Model .......................... 5
   2.2 Fading Number .......................... 6

3 Preliminaries 9
   3.1 Linear Prediction ........................ 9
   3.2 Useful Lemmas ........................... 10

4 Previous Results 13
   4.1 Asymptotic Capacity ...................... 13
   4.2 Non-Asymptotic Bounds .................... 14

5 Numerical Lower Bound 17
   5.1 Memoryless Look and Lower Bound on the Capacity 17
   5.2 Application of the Algorithm ................ 19
   5.3 Blahut-Arimoto Algorithm Review ............. 22
   5.4 Simulation Results ....................... 24

6 Numerical Upper Bound 29
   6.1 Upper Bound on the Capacity ............... 29
   6.2 Cutting-Plane Algorithm Review ............. 32
   6.3 Simulation Results ....................... 33

7 Summary and Conclusion 37

A Simplification of Entropy Calculation 39
Wireless communication channels are usually corrupted by noise, generally multiplicative and additive ones. The multiplicative noise (also known as fading) is often modeled by a Gaussian process with memory, whereas the additive noise is often modeled by a white Gaussian process.

So far, evaluating the channel capacity of wireless communication systems is usually done by an approach where the channel state is measured with the help of regularly transmitted training sequences. The detection of the transmitted data is then done under the assumption of perfect knowledge of the channel state.

For very high data rate systems, however, evaluating the channel capacity under perfect knowledge of channel state is not sufficient anymore, because we can not ignore the bandwidth that we lose when measuring the channel state. Therefore we should rely on a joint estimation and detection scheme that also takes into account the estimation part. We assume that the statistical characteristics, but not realizations, of the fading process are known to the receiver and transmitter. The capacity which is evaluated by this approach is known as non-coherent capacity. Investigating the non-coherent capacity of fading channels is sometimes referred to as investigating the capacity of non-coherent fading channels.

The exact expression of the non-coherent capacity of fading channels is not known yet. Therefore, researchers have been working on two aspects in order to better understand the non-coherent capacity. The two aspects are asymptotic capacity and non-asymptotic capacity, where asymptotic capacity means the capacity at very high SNR (signal-to-noise ratio) and non-asymptotic means the capacity at any value of SNR.
Chapter 1 Introduction

The behavior of asymptotic capacity is already well understood. In fact, it has been shown by Lapidoth and Moser [1] that the non-coherent capacity is very power-inefficient in the high SNR regime in the sense that the capacity grows only double-logarithmically in the SNR at high SNR, which means increasing the transmission rate by an additional bit would require squaring the necessary SNR.

The non-asymptotic capacity however is more difficult and not clearly understood yet. Hence, instead of continuing with the exact expression of the non-asymptotic capacity, researchers turn to investigate bounds on the non-asymptotic capacity. By efforts of the past researchers, there are already some results presented. In the particular case of single-input single-output (SISO) Gaussian fading channels with memory, an upper bound and a lower bound of the capacity have been derived in [5].

In this project, we aim to obtain a numerical lower bound on the non-asymptotic capacity using a numerical algorithm, namely the Blahut-Arimoto algorithm on the basis of Blahut [6] and Arimoto [7]. Besides, we also aim to obtain a numerical upper bound on the non-asymptotic capacity using another numerical algorithm, namely the cutting-plane algorithm proposed by Huang and Meyn [8].

Some of the contributions of this thesis project are listed as follows:

- We derive a lower bound which is applicable for the Blahut-Arimoto algorithm, and then implement the Blahut-Arimoto algorithm to do the optimization in the derived lower bound.

- We derive an upper bound which is applicable for the cutting-plane algorithm, and then implement the cutting-plane algorithm to do the optimization in the derived upper bound.

- After we have a new numerical upper bound and a new numerical lower bound on the non-asymptotic capacity, we compare the two new numerical bounds with the previous analytical bounds.

The organization of this report is as follows: in Chapter 2, we introduce the channel model and the definition of fading number. In Chapter 3, we recall theory about linear prediction and state some useful lemmas. In Chapter 4, previous results on the asymptotic capacity as well as non-asymptotic bounds are provided. In Chapter 5, a numerical lower bound obtained by the Blahut-Arimoto algorithm is presented and compared to
previous results. In Chapter 6, a numerical upper bound obtained by the
cutting-plane algorithm is presented and compared to previous results. In
Chapter 7 we summarize all results and conclude this report. Additionally,
in Appendix A, we give in detail the derivation process that simplifies the
differential entropy of the sum of a uniform-phase distribution and a Gauss-
ian distribution. In Appendix B, we briefly describe the structure of the
programs that we write to do the simulations.

We now introduce the notation used in this report. For random quan-
tities we use upper-case letters, e.g., $X$ and for their realizations we use
lower-case letters, e.g., $x$. In dealing with sequences of random variables
we use a combination of superscripts and subscripts to address consecutive
subsets. Thus, if $X_1, X_2, \ldots$, is a sequence of random variables, then $X^n_k$
will designate the sequence $X_k, \ldots, X_n$. Besides, the curly bracket are used
to denote random processes, i.e., while $H_k$ denotes a random variable (at
time $k$), $\{H_k\}$ denotes the (discrete) random process

$$\ldots, H_{-2}, H_{-1}, H_0, H_1, H_2, \ldots$$

Finally, we use $\log(\cdot)$ to denote the natural logarithmic function.
Chapter 2

Channel Model

2.1 Channel Model

We consider a discrete-time channel

\[ Y_k = H_k x_k + Z_k, \quad (2.1) \]

where \( Y_k \in \mathbb{C} \) is the time-\( k \) complex-valued output; \( x_k \in \mathbb{C} \) is the time-\( k \) complex-valued input; the complex process \( \{ H_k \} \) models multiplicative noise; and the complex process \( \{ Z_k \} \) models additive noise. The processes \( \{ H_k \} \) and \( \{ Z_k \} \) are assumed to be independent and of a joint law that does not depend on the input sequence \( \{ x_k \} \).

We shall assume that \( \{ Z_k \} \) is a sequence of i.i.d. circularly symmetric complex-Gaussian random variables of zero mean and variance \( \sigma^2 \). Thus, \( Z_k \sim \mathcal{N}_\mathbb{C}(0, \sigma^2) \) where we use the notation \( W \sim \mathcal{N}(\mu, \sigma^2) \) to indicate that the density function \( f_W(w) \) of \( W \) is given by

\[ f_W(w) = \frac{1}{\pi \sigma^2} e^{-\frac{|w-\mu|^2}{\sigma^2}}, \quad w \in \mathbb{C}. \quad (2.2) \]

In addition, we shall assume the fading process \( \{ H_k \} \) to be a zero-mean, unit-variance, stationary, circularly symmetric, Gaussian process of arbitrary spectral distribution function \( F(\lambda), -\frac{1}{2} \leq \lambda \leq \frac{1}{2} \). Thus, \( F(\cdot) \) is a monotonically non-decreasing function on \( (-\frac{1}{2}, \frac{1}{2}] \) with

\[ E[H_{k+m}H_k^*] = \int_{-\frac{1}{2}}^{\frac{1}{2}} e^{i2\pi m \lambda} dF(\lambda), \quad k, m \in \mathbb{Z}, \quad (2.3) \]

and

\[ E[|H_k|^2] = 1. \quad (2.4) \]
Notice that we denote the derivative $F'(\lambda)$ as the *spectral density function*.

We further assume that the receiver and transmitter, while fully cognizant of the distribution of the fading process $\{H_k\}$, have no knowledge of its realization.

We also assume a peak-power constraint on the inputs

$$|x_k| \leq A.$$  \hspace{1cm} (2.5)

The signal-to-noise ratio (SNR) is defined by

$$\text{SNR} = \frac{A^2}{\sigma^2}. \hspace{1cm} (2.6)$$

The subject of our investigation is the capacity $C(\text{SNR})$ which is defined by

$$C(\text{SNR}) = \lim_{n \to \infty} \frac{1}{n} \sup I(X_1, \ldots, X_n; Y_1, \ldots, Y_n), \hspace{1cm} (2.7)$$

where the supremum is over all joint distributions on $X_1, \ldots, X_n$ that satisfy the peak-power constraint (2.5), and where the limit exists because $\{H_k\}$ and $\{Z_k\}$ are stationary.

### 2.2 Fading Number

If the process $\{H_k\}$ is *regular*, it was demonstrated in [2] that the capacity of the non-coherent channel grows only double-logarithmically in the SNR at high SNR. Here, we say that the stationary fading process $\{H_k\}$ is regular if it has a finite differential entropy rate, i.e.,

$$h(\{H_k\}) \triangleq \lim_{n \to \infty} \frac{1}{n} h(H_1, \ldots, H_n) = \lim_{n \to \infty} h(H_n|H_1, \ldots, H_{n-1}) > -\infty, \hspace{1cm} (2.8)$$

where $h$ denotes the differential entropy (see [12]), and (2.9) follows from the stationarity of $\{H_k\}$.

In order to better understand the phenomenon and to find out the rate at which it occurs, the fading number is introduced as the second term in the high-SNR asymptotic expansion of capacity. Specifically, considering a stationary and ergodic fading process, the fading number is defined as

$$\chi(\{H_k\}) = \lim_{\text{SNR} \to \infty} \{C(\text{SNR}) - \log \log \text{SNR}\}. \hspace{1cm} (2.10)$$
Thus, whenever $\chi(\{H_k\})$ is finite and the limit in (2.10) exists, the capacity is given by [1]

$$C(\text{SNR}) = \log (1 + \log(1 + \text{SNR})) + \chi(\{H_k\}) + o(1),$$

(2.11)

where the $o(1)$ term is small at high SNR. Equation (2.11) indicates that at high SNR, the log log SNR term dominates, and thus capacity grows only double-logarithmically in the SNR.

Capacity grows double-logarithmically in the SNR means that if we want to increase the transmission rate by an additional bit, we will need to square the SNR, which is obviously very power-inefficient. As a matter of fact, it can be shown that the fading number is related to the threshold between power-efficient low-SNR and power-inefficient high-SNR regimes. For rates that are significantly higher than the fading number, communication becomes extremely power-inefficient.
Chapter 3

Preliminaries

In Section 3.1, we review some knowledge about linear prediction, which is crucial when deriving capacity bounds of channels with memory. In Section 3.2, we present some lemmas that will be used in later chapters.

3.1 Linear Prediction

Prediction is the estimation of the present or future values in terms of its past values. If the estimate is a linear combination of all past values, we refer to it as a linear prediction.

A noiseless prediction means that $H_0$ is predicted from past values $H_{-1}, H_{-2}, \ldots$ that are not corrupted by any noise. It has been shown that the minimum mean-squared error $\epsilon^2_{\text{pred}}$ of noiseless prediction is directly related to the spectral density of the process $F'(\cdot)$:

$$\epsilon^2_{\text{pred}} = \exp \left\{ \int_{-\frac{1}{2}}^{\frac{1}{2}} \log F'(\lambda) \, d\lambda \right\}. \quad (3.1)$$

This formula is known as Kolmogorov’s formula, see for example [11, pp. 590-591]. Note that as long as linear prediction is performed, this formula applies for all kinds of processes. However, once a Gaussian process is concerned, this prediction error will be the minimum among all others.

On the other hand, a noisy prediction means that the observations of the past are corrupted by noise that will negatively affect the prediction. Let $\{W_k\}$ be a sequence of i.i.d. $\mathcal{N}_C(0, \delta^2)$ random variables, which are independent of the process $\{A_k\}$, i.e., $A_k \perp W_m$, where $k, m \in \mathbb{Z}$. The noisy prediction problem is to predict $A_0$ based on the observations of the past
The minimum mean-squared prediction error denoted by \( \epsilon_{\text{pred}}^2(\delta^2) \) is then given by [4]

\[
\epsilon_{\text{pred}}^2(\delta^2) = \exp\left\{ \int_{-\frac{1}{2}}^{\frac{1}{2}} \log \left( F'(\lambda) + \delta^2 \right) d\lambda \right\} - \delta^2,
\]

where \( F'(\cdot) \) is the spectral density of the process \( \{A_k\} \). Note that in Section 2.2 we mentioned that a stationary process is regular if it has a finite differential entropy. For stationary Gaussian processes, this is equivalent to the condition that \( \epsilon_{\text{pred}}^2(0) > 0 \). Similarly, a stationary Gaussian process is non-regular or deterministic if \( \epsilon_{\text{pred}}^2(0) = 0 \). See [4] for more information.

Next, we recall some facts related to the prediction problem for the case of circularly symmetric stationary Gaussian processes. Let the minimum mean-squared error estimate of \( A_0 \) given the past observations \( A_{-1}, A_{-2}, \ldots, A_{-n} \) be denoted by \( \hat{A}_0^{(n)} \)

\[
\hat{A}_0^{(n)} = E[A_0|A_{-n}^{-1}].
\]

Let the error be denoted by \( \tilde{A}_0^{(n)} \) with variance \( \tilde{c}_n^2 \)

\[
\tilde{A}_0^{(n)} = A_0 - \hat{A}_0^{(n)}.
\]

In particular, if \( \{A_k\} \) is a zero-mean Gaussian process, \( \hat{A}_0^{(n)} \) and \( \tilde{A}_0^{(n)} \) will be independent of each other and have the distributions \( \hat{A}_0^{(n)} \sim N_\mathbb{C}(0, 1 - \tilde{c}_n^2) \) and \( \tilde{A}_0^{(n)} \sim N_\mathbb{C}(0, \tilde{c}_n^2) \), respectively. Additionally, if \( \{A_k\} \) is stationary, then the prediction error is monotonically non-increasing in \( n \)

\[
\lim_{n \to \infty} E\left[ |A_0 - \hat{A}_0^{(n)}|^2 \right] = E\left[ |A_0 - E[A_0|A_{-\infty}^{-1}]|^2 \right],
\]

and therefore

\[
\lim_{n \to \infty} \tilde{c}_n^2 = \epsilon_{\text{pred}}^2(0).
\]

### 3.2 Useful Lemmas

**Lemma 3.1.**

\[
I\left( X; (\hat{d} + \tilde{D})X + Z \right) = I\left( X; (\hat{d})X + Z \right), \quad \hat{d} = |\hat{d}|e^{i\phi}
\]

where \( \hat{d} \) and \( \phi \in [0, 2\pi) \) are deterministic, and where \( \tilde{D} \) and \( Z \) are circularly symmetric.

\(^1\epsilon_{\text{pred}}^2(0) \) is equivalent to \( \epsilon_{\text{pred}}^2 \) in (3.1).
3.2 Useful Lemmas

Proof.

\[
I\left(X; (|d|e^{i\phi} + \tilde{D})X + Z\right) = I\left(X; (|d|e^{i\phi} + \tilde{D}e^{i\phi})X + Ze^{i\phi}\right)
= I\left(X; ((|d| + \tilde{D})X + Z)e^{i\phi}\right)
= I\left(X; (|d| + \tilde{D})X + Z\right)
\]

where the first equality follows from the circular symmetry of \(\tilde{D}\) and \(Z\), in which case the laws of \(\tilde{D}\) and \(Z\) are identical to the respective laws of \(\tilde{D}e^{i\phi}\) and \(Ze^{i\phi}\); the last equality follows because \(\phi\) is deterministic: removing \(e^{i\phi}\) does not change mutual information.

Lemma 3.2. Consider a channel model

\[
Y_k = H_kx_k + Z_k, \quad (3.8)
\]

where the additive noise process \(\{Z_k\}\) is assumed to be circularly symmetric. Then the capacity-achieving input distribution can be assumed to be circularly symmetric, i.e., the input sequence \(\{X_k\}\) can be replaced by \(\{X_ke^{i\Theta_k}\}\), where \(\{\Theta_k\}\) are IID \(\sim \mathcal{U}([0, 2\pi])\), independent of every other random quantity.

Proof.

\[
\frac{1}{n}I(X^n_1; Y^n_1) = \frac{1}{n}I(X^n_1; Y^n_1 \mid \{e^{i\Theta_l}\}_{l=1}^n)
= \frac{1}{n}I\left(\{X_k e^{i\Theta_k}\}_{k=1}^n; \{Y_k e^{i\Theta_k}\}_{k=1}^n \mid \{e^{i\Theta_l}\}_{l=1}^n\right)
= \frac{1}{n}I\left(\{X_k e^{i\Theta_k}\}_{k=1}^n; \{H_kx_ke^{i\Theta_k} + Z_k\}_{k=1}^n \mid \{e^{i\Theta_l}\}_{l=1}^n\right)
= \frac{1}{n}I\left(\tilde{X}_1^n; \{H_1\tilde{X}_1 + Z_1\}_{l=1}^n \mid \{e^{i\Theta_l}\}_{l=1}^n\right)
= \frac{1}{n}h\left(\{H_1\tilde{X}_1 + Z_1\}_{l=1}^n \mid \{e^{i\Theta_l}\}_{l=1}^n\right)
= \frac{1}{n}h\left(\{H_1\tilde{X}_1 + Z_1\}_{l=1}^n \mid \tilde{X}_1^n, \{e^{i\Theta_l}\}_{l=1}^n\right)
\leq \frac{1}{n}h\left(\{H_1\tilde{X}_1 + Z_1\}_{l=1}^n\right) - \frac{1}{n}h\left(\{H_1\tilde{X}_1 + Z_1\}_{l=1}^n \mid \tilde{X}_1^n\right)
\]

\[
= \frac{1}{n}I\left(\tilde{X}_1^n; \{H_1\tilde{X}_1 + Z_1\}_{l=1}^n\right).
\]
Here the first equality follows because $\{\Theta_k\}$ is independent of every other random quantity; the third equality follows because $\{Z_k\}$ is circularly symmetric; in the subsequent equality we substitute $\tilde{X}_l = X_l e^{i\Theta_l}$; and the inequality follows since conditioning reduces entropy.

Hence, a circularly symmetric input achieves a mutual information that is at least as big as the original mutual information.
Chapter 4

Previous Results

In this chapter, we review some previous results based largely on [5]. In Section 4.1 we review the asymptotic behavior of capacity at high SNR. In Section 4.2 we recall the non-asymptotic upper and lower bounds of channel capacity with a peak-power constraint.

4.1 Asymptotic Capacity

The asymptotic capacity of SISO fading channels with memory is well understood. It was shown that for regular processes the capacity of non-coherent fading channels grows only double-logarithmically in high SNR. In Section 2.2 the capacity was given as

\[ C(\text{SNR}) = \log(1 + \log(1 + \text{SNR})) + \chi(\{H_k\}) + o(1), \]  

where \( \chi(\{H_k\}) \) is the fading number. Because \( o(1) \) term is small at high SNR,

\[ C(\text{SNR}) \approx \log(1 + \log(1 + \text{SNR})) + \chi(\{H_k\}), \]  

becomes a good approximation of the channel capacity in the high-SNR regime.

For zero-mean unit-variance Gaussian fading channel with spectral density \( F'(\lambda), -1/2 \leq \lambda \leq 1/2 \), the fading number is given by [1]

\[ \chi(\{H_k\}) = -1 - \gamma + \log \frac{1}{\zeta^2_{\text{pred}}}, \]  

where \( \gamma \approx 0.577 \) denotes Euler’s constant and \( \zeta^2_{\text{pred}} \) denotes the prediction error computed by (3.1).
4.2 Non-Asymptotic Bounds

The explicit expression of the non-asymptotic capacity is not known yet. Hence, researchers are trying to derive upper and lower bounds on the capacity and make the bounds as tight as possible. Non-asymptotic upper and lower bounds on the capacity of SISO Gaussian fading channels with memory under a peak-power constraint on the inputs were given in [5]. It was shown that for the case of zero-mean Gaussian fading process the capacity is upper-bounded by

\[
C(SNR) \leq C_{\text{IID}}(SNR) + \log \frac{1 + 1/\text{SNR}}{\epsilon^2(1/\text{SNR}) + 1/\text{SNR}}, \tag{4.4}
\]

where \( C_{\text{IID}}(SNR) \) denotes the capacity in the memoryless fading case, and \( \epsilon^2(\delta^2) \), computed by (3.2), denotes the present prediction error from a variance-\( \delta^2 \) noisy observation from the past. It was shown in [1] that the capacity of the memoryless SISO Gaussian fading channel with zero-mean can be upper-bounded by

\[
C_{\text{IID}}(SNR) \leq \inf_{\alpha, \beta' > 0} \inf_{\delta' \geq 0} \left\{ -1 + \alpha \log \beta' + \log \left( \frac{\alpha}{\beta'} \right) + \frac{1 + \text{SNR}}{\beta'} + \frac{\delta'}{\beta'} + (1 - \alpha) \left( \log \delta' - \epsilon \delta' \text{Ei}(-\delta') \right) \right\}. \tag{4.5}
\]

On the other hand, by considering inputs that are i.i.d. and uniformly distributed over the set \( \{ z \in \mathbb{C} : \alpha A \leq |z| \leq A \} \) with \( 0 < \alpha < 1 \), and by maximizing over \( \alpha \), the capacity can be lower-bounded by:

\[
C(SNR) \geq \sup_{0 < \alpha < 1} \left\{ - \exp \left( \frac{\epsilon^2(\delta^2) + \frac{2}{\text{SNR}(1 + \alpha)}}{1 - \epsilon^2(\delta^2)} \right) \cdot \text{Ei} \left( - \frac{\epsilon^2(\delta^2) + \frac{2}{\text{SNR}(1 + \alpha)}}{1 - \epsilon^2(\delta^2)} \right) - \log \left( \frac{e(1 + \alpha^2)}{2(1 - \alpha^2)} \right) \right\}, \tag{4.6}
\]

where \( \delta^2 = \frac{1}{\alpha^2 \text{SNR}} \).

Finally, we consider a spectral density which has the following form:

\[
F'(\lambda) = \begin{cases} 
\gamma_1, & |\lambda| < \gamma_3 \\
\gamma_2, & \gamma_3 < |\lambda| \leq \frac{1}{2}
\end{cases}, \tag{4.7}
\]

where the parameters \( \gamma_1, \gamma_2, \) and \( \gamma_3 \) are chosen such that

\[
\int_{-\frac{1}{2}}^{\frac{1}{2}} F'(\lambda) d\lambda = 1, \tag{4.8}
\]
Figure 4.1: Analytical bounds on the capacity of a SISO zero-mean Gaussian fading channel with memory. Depicted are analytical upper and lower bounds on the capacity (4.4), (4.6) for a fading process with spectral density (4.7), (4.10); the asymptotic expansion (4.2); and the fading number (4.3).

and\footnote{The value of $\epsilon_{\text{pred}}^2$ can be randomly chosen. Here we choose 0.0769, which is the same value as that in [9], for the purpose of comparison.}

$$\epsilon_{\text{pred}}^2 = \exp \left\{ \int_{-\frac{1}{2}}^{\frac{1}{2}} \log F'(\lambda) \, d\lambda \right\} = 0.0769. \quad (4.9)$$

Since there is an infinite number of combinations of $\gamma_1$, $\gamma_2$, and $\gamma_3$ that all match the conditions (4.8) and (4.9), we have randomly chosen the following combination:

$$\begin{align*}
\gamma_1 &= 1.997 \\
\gamma_2 &= 0.003 \\
\gamma_3 &= 0.25
\end{align*} \quad (4.10)$$

We plot Figure 4.1 using the values in (4.10). Note that we will keep these parameters throughout the whole report when we plot figures.
Chapter 5

Numerical Lower Bound

In this chapter, we describe how we can apply the Blahut-Arimoto algorithm to obtain a numerical lower bound on the capacity. In Section 5.1 we review a method to achieve a memoryless look on the channel, while in Section 5.2 we derive a lower bound which is suitable for applying the Blahut-Arimoto algorithm. In Section 5.3 we briefly review the Blahut-Arimoto algorithm. In Section 5.4 we present the simulation results and give some comments.

5.1 Memoryless Look and Lower Bound on the Capacity

The definition of capacity is given in (2.7); it is however very difficult to perform the maximization over all joint distributions on an infinite number of inputs. Consequently, a trick developed by Lapidoth was introduced in [4]. By using this method, we can remove the channel (fading) memory and turn it into noisy fading prediction.

To obtain a tighter lower bound on the capacity, we consider, apart from (2.5), the inputs to be i.i.d. and of magnitude no smaller than $\alpha A$, i.e.,

$$\alpha A \leq |x_k| \leq A, \quad 0 < \alpha < 1. \quad (5.1)$$

The idea of considering such an additional constraint on the inputs is induced from the concept of input distributions that escape to infinity by Lapidoth and Moser [3]. As the SNR tends to infinity, the probability of low-power portion of input distributions tends to zero. In other words, the low-power portion of input distributions becomes less significant as the SNR increases,
and therefore we try dropping the lower-power portion and see if we will obtain a better result.

Using the chain rule

\[
\frac{1}{n} I(X_1^n; Y_1^n) = \frac{1}{n} \sum_{k=1}^{n} I(X_k; Y_1^n | X_1^{k-1}),
\]

(5.2)

and a Cesáro-type theorem [12, Theorem 4.2.3], after getting rid of future outputs, we will obtain a lower bound on the capacity \( C \)

\[
C \geq \lim_{k \to \infty} I(X_k; Y_1^k | X_1^{k-1}).
\]

(5.3)

Directly cited from [4], we further have

\[
I(X_k; Y_1^k | X_1^{k-1}) \geq I(X_k; (\hat{D}_k + \tilde{D}_k) X_k + Z_k | \hat{D}_k),
\]

(5.4)

where \( X_k, Z_k, \hat{D}_k, \tilde{D}_k \) are independent random variables of the following laws: \( X_k \) is in the set \( \{x_k \in C : \alpha A \leq |x_k| \leq A\} \); the additive noise \( Z_k \sim \mathcal{N}_C(0, \sigma^2) \); the prediction error in predicting \( H_k \) from its past observations is \( \tilde{D}_k \sim \mathcal{N}_C(0, \tilde{\epsilon}_k^2) \); and \( \hat{D}_k \sim \mathcal{N}_C(0, 1 - \tilde{\epsilon}_k^2) \), where \( \tilde{\epsilon}_k^2 \) is such that

\[
\lim_{k \to \infty} \tilde{\epsilon}_k^2 = \epsilon_{\text{pred}}^2 \left( \delta^2 \right) |_{\delta^2 = \frac{\sigma^2}{(\alpha A)^2}}.
\]

(5.5)

We will use \( X, Z, \hat{D}, \tilde{D} \) as the respective representations of \( X_k, Z_k, \hat{D}_k, \tilde{D}_k \) when \( k \) goes to infinity.

To obtain a tight lower bound, we need to maximize \( I(X; (\hat{D} + \tilde{D}) X + Z | \hat{D}) \) over \( X \) and then perform maximization over \( \alpha \). Let us first define

\[
\tilde{C}_{\text{IID}} \left( \frac{A^2}{\sigma^2}, \alpha, \epsilon_{\text{pred}}^2 \right) \triangleq \sup_X I(X; (\hat{D} + \tilde{D}) X + Z | \hat{D}),
\]

(5.6)

where the maximization is performed for a given SNR over all input distributions on \( X \) of the magnitude constraint (5.1). Then performing maximization over \( \alpha \) along with (5.3), (5.4), and (5.6) gives us the lower bound on the capacity of a SISO zero-mean Gaussian fading channel with memory

\[
C(\text{SNR}) \geq \sup_{0 < \alpha < 1} \tilde{C}_{\text{IID}} \left( \frac{A^2}{\sigma^2}, \alpha, \epsilon_{\text{pred}}^2 \right).
\]

(5.7)
5.2 Application of the Algorithm

In this section we present the derivation steps which finally give a lower bound in a simulation-feasible form:

$$\sup_X I \left( X; (\hat{D} + \hat{D})X + Z \mid \hat{D} \right)$$

$$= \sup_X \int I \left( X; (\hat{D} + \hat{D})X + Z \mid \hat{D} = \hat{d} \right) f_{\hat{D}}(\hat{d}) \, d\hat{d} \quad (5.8)$$

$$= \sup_X \int I \left( X; (|\hat{d}| + \hat{D})X + Z \right) f_{|\hat{D}|}(|\hat{d}|) \, d|\hat{d}| \quad (5.9)$$

$$\geq \sup_X \sum_{|\hat{d}|} I_D \left( X; (|\hat{d}| + \hat{D})X + Z \right) P_{|\hat{D}|}(|\hat{d}|). \quad (5.10)$$

Here in the first equality we use $f_{\hat{D}}(\hat{d})$ to denote the probability density function of $\hat{D}$, and since $\hat{D}$ is $\mathcal{N}_\mathbb{C} \left( 0, 1 - \epsilon^2_{\text{pred}} \left( \frac{\sigma^2}{\alpha^2 A^2} \right) \right)$ distributed, $|\hat{D}|$ is Rayleigh distributed:

$$f_{|\hat{D}|}(|\hat{d}|) = \frac{2|\hat{d}|}{1 - \epsilon^2_{\text{pred}} \left( \frac{\sigma^2}{\alpha^2 A^2} \right)} \exp \left( \frac{-|\hat{d}|^2}{1 - \epsilon^2_{\text{pred}} \left( \frac{\sigma^2}{\alpha^2 A^2} \right)} \right). \quad (5.11)$$

The second equality follows from Lemma 3.1 because $\hat{D}$ and $Z$ are circularly symmetric.
Lastly, in (5.10) the lower bound is achieved by the following steps: we first partition \( f_{\hat{D}\mid\hat{d}}(|\hat{d}|) \) with equal step size into subintervals, and then we take the smallest value in each subinterval. \( P_{\hat{D}\mid\hat{d}}(|\hat{d}|) \) is the collection of these values. We illustrate the idea in Figure 5.1, where the blue curve can be thought of as the continuous function \( f_{\hat{D}\mid\hat{d}}(|\hat{d}|) \), and the black dots are the smallest values in each subinterval. Then, \( P_{\hat{D}\mid\hat{d}}(|\hat{d}|) \) is the discrete function that consists of these values. On the other hand, \( I_{\hat{D}}(X; (|\hat{d}| + \tilde{D})X + Z) \) is obtained by the same approach as above, and we use subscript \( D \) to denote the difference from the original mutual information. Consequently, the product of the two smallest values in each subinterval can guarantee the lower bound.

Before continuing with (5.10), we should notice one thing which helps simplify the task. It is shown in Lemma 3.2 that the capacity-achieving input distribution can be assumed to be circularly symmetric if the additive noise is also circularly symmetric. In our case the additive noise \( Z \) is circularly symmetric, so we can actually denote \( X \) by its polar representation \( Re^{i\Theta} \) and assume \( \Theta \sim U([0, 2\pi]) \). Therefore, substituting \( X \) by \( Re^{i\Theta} \) in (5.10), we have

\[
I_{\hat{D}} \left( X; (|\hat{d}| + \tilde{D})X + Z \right)
= I_{\hat{D}} \left( Re^{i\Theta}; (|\hat{d}| + \tilde{D})Re^{i\Theta} + Z \right) 
= I_{\hat{D}} \left( R, \Theta; (|\hat{d}| + \tilde{D})Re^{i\Theta} + Z \right) 
= I_{\hat{D}} \left( R; (|\hat{d}| + \tilde{D})Re^{i\Theta} + Z \right) 
+ I_{\hat{D}} \left( \Theta; (|\hat{d}| + \tilde{D})Re^{i\Theta} + Z \mid R \right),
\]

where (5.13) follows because once given \( Re^{i\Theta}, R \) and \( \Theta \) are known, and vice versa; (5.14) follows from chain rule of mutual information. Note that \( \Theta \sim U([0, 2\pi]) \), and \( \alpha A \leq R \leq A \) from (5.1). Note also that there is nothing more we can do to further simplify the first term of (5.14), but for the second term, we can proceed as follows:

\[
I_{\hat{D}} \left( \Theta; (|\hat{d}| + \tilde{D})Re^{i\Theta} + Z \mid R \right)
= I_{\hat{D}} \left( \Theta; (|\hat{d}| + \tilde{D})e^{i\Theta} + \frac{Z}{R} \mid R \right) 
\geq I_{\hat{D}} \left( \Theta; (|\hat{d}| + \tilde{D})e^{i\Theta} + \frac{Z}{\alpha A} \right)
\]
where the first equality follows because the mutual information does not change if divided by a given value; the subsequent inequality is because we replace $R$ by its smallest value $\alpha A$ which results in the biggest noise and therefore the smallest mutual information; the following equality follows because $\hat{D}$ is circularly symmetric; the following equality is due to the relationship between differential entropy and mutual information; and the last equality follows because once $\Theta$ is given, the term $|\hat{d}| e^{i \Theta}$ becomes deterministic and thus does not change the differential entropy.

We can go one step further after (5.19). Since $\tilde{D}$ and $Z$ are both complex Gaussian distributed and independent of each other, the distribution of $\tilde{D} + \frac{Z}{\alpha A}$ is equivalent to $\mathcal{N}_C \left( 0, \epsilon^2_{\text{pred}} \left( \frac{\sigma^2}{\alpha^2 A^2} \right) + \frac{\sigma^2}{\alpha^2 A^2} \right)$. Hence, we have

$$h \left( \tilde{D} + \frac{Z}{\alpha A} \right) = \log \left( \pi e \left( \epsilon^2_{\text{pred}} \left( \frac{\sigma^2}{\alpha^2 A^2} \right) + \frac{\sigma^2}{\alpha^2 A^2} \right) \right).$$  (5.20)

It is however much more complicated and needs more mathematical skills to deal with the term $h \left( |\hat{d}| e^{i \Theta} + \tilde{D} + \frac{Z}{\alpha A} \right)$, which is in fact the differential entropy of the sum of a uniform-phase and a Gaussian distribution. See Appendix A for a detailed derivation of simplifying this differential entropy. We cite the result here for convenience. Given $\Theta \sim \mathcal{U} ([0, 2\pi])$, $W \sim \mathcal{N}_C (0, \sigma^2_W)$, and $d \geq 0$, the differential entropy is

$$h \left( de^{i \Theta} + W \right) = \log \left( \pi \sigma^2_W \right) + 1 + 2 \frac{d^2}{\sigma^2_W}$$

$$- \frac{1}{\sigma^2_W} \exp \left( - \frac{d^2}{\sigma^2_W} \right) \int_0^\infty \exp \left( - \frac{z}{\sigma^2_W} \right) I_0 \left( \frac{2d \sqrt{z}}{\sigma^2_W} \right) \log I_0 \left( \frac{2d \sqrt{z}}{\sigma^2_W} \right) \, dz.$$

Hence using $\sigma^2_W = \epsilon^2_{\text{pred}} \left( \frac{\sigma^2}{\alpha^2 A^2} \right) + \frac{\sigma^2}{\alpha^2 A^2}$, we get the following lower bound:

$$I_D \left( \Theta; (|\hat{d}| + \tilde{D}) e^{i \Theta} + Z \mid R \right) \geq h \left( |\hat{d}| e^{i \Theta} + \tilde{D} + \frac{Z}{\alpha A} \right) - h \left( \tilde{D} + \frac{Z}{\alpha A} \right)$$  (5.21)
\[ C(SNR) \geq \sup_{0 < \alpha < 1} \sup_X \sum_{|\hat{d}|} I_D \left( X; (|\hat{d}| + \hat{D})X + Z \right) P_{|\hat{D}|}(|\hat{d}|) \]

\[ \geq \sup_{0 < \alpha < 1} \left\{ \sum_{|\hat{d}|} I_D \left( R; (|\hat{d}| + \hat{D})R e^{i\theta} + Z \right) P_{|\hat{D}|}(|\hat{d}|) \right. \]

\[ + \left. \sum_{|\hat{d}|} I_D \left( \Theta; (|\hat{d}| + \hat{D})R e^{i\theta} + Z \right) R P_{|\hat{D}|}(|\hat{d}|) \right\} \]

\[ \geq \sup_{0 < \alpha < 1} \left\{ \sup_R \sum_{|\hat{d}|} I_D \left( R; (|\hat{d}| + \hat{D})R e^{i\theta} + Z \right) P_{|\hat{D}|}(|\hat{d}|) \right. \]

\[ + \left. \sum_{|\hat{d}|} P_{|\hat{D}|}(|\hat{d}|) \left[ \frac{2|\hat{d}|^2}{\sigma_W^2} - \frac{1}{\sigma_W^2} \exp \left( -\frac{|\hat{d}|^2}{\sigma_W^2} \right) \right. \]

\[ \left. \cdot \int_0^\infty \exp \left( -\frac{z}{\sigma_W^2} \right) I_0 \left( \frac{2|\hat{d}|\sqrt{z}}{\sigma_W^2} \right) \log I_0 \left( \frac{2|\hat{d}|\sqrt{z}}{\sigma_W^2} \right) dz \right\}, \]

where the first term of (5.26) will be numerically computed using the Blahut-Arimoto Algorithm, which is briefly reviewed in the following section, while the second term will be computed numerically using standard mathematical tools.

### 5.3 Blahut-Arimoto Algorithm Review

In this section, we describe the Blahut-Arimoto algorithm which is used to compute the first term of (5.26). This section is largely based on [10] and
We consider a discrete memoryless channel with input alphabet \( \mathcal{X} \), output alphabet \( \mathcal{Y} \), and transition probabilities function \( W(y|x), x \in \mathcal{X}, y \in \mathcal{Y} \). (Notice that \( \mathcal{X} \subset [\alpha A, A] \).) Then the mutual information for the input distribution \( Q(x) \), \( x \in \mathcal{X} \) is given by

\[
I(Q, W) = \sum_{x,y} Q(x)W(y|x) \log \frac{W(y|x)}{\sum_{x'} Q(x')W(y|x')}.
\]

(5.27)

It was shown by Shannon that the capacity of discrete memoryless channels could be expressed as the optimum value of the mutual information over all input probability distributions, i.e.,

\[
C = \max_Q I(Q, W).
\]

(5.28)

Now let us define the function \( J \) which contains \( V(x, y) \):

\[
J(Q, V) = \sum_{x,y} Q(x)W(y|x) \log \frac{V(x,y)}{Q(x)}.
\]

(5.29)

Then for a fixed input distribution \( Q \), we can maximize \( J \) over \( V \). The maximizer is attained at

\[
V(x,y) = \frac{Q(x)W(y|x)}{\sum_{x'} Q(x')W(y|x')}.
\]

(5.30)

Next, we plug the attained maximizer \( V \) in (5.29). Now for the fixed \( V \), we can maximize \( J \) over \( Q \). The maximizer is attained at

\[
Q(x) = \frac{\alpha(x)}{\sum_{x'} \alpha(x')},
\]

(5.31)

where

\[
\alpha(x) = \exp \sum_y W(y|x) \log V(x,y).
\]

(5.32)

Then with the new \( Q \), we can maximize \( J \) over \( V \) again, and so on. In other words, the Blahut-Arimoto algorithm progresses by alternately maximizing over \( Q \) for a given \( V \) and maximizing over \( V \) for a given \( Q \). It has been shown that \( J \) eventually converges to the capacity. To summarize, the algorithm works as follows:

1. Set \( n=0 \). Choose an initial distribution \( Q^{(n)}(x) \), with \( Q^{(n)}(x) > 0 \) for \( x \in \mathcal{X} \).
2. Maximize $J(Q^{(n)}, V)$ over $V$. The maximizer $V$ is

$$V^{(n)}(x, y) = \frac{Q^{(n)}(x)W(y|x)}{\sum_{x'} Q^{(n)}(x')W(y|x')}, \quad x \in \mathcal{X}.$$  \hspace{1cm} (5.33)

3. Maximize $J(Q, V^{(n)})$ over $Q$. The maximizer $Q$ is

$$Q^{(n+1)}(x) = \frac{\alpha^{(n)}(x)}{\sum_{x'} \alpha^{(n)}(x')}, \quad x \in \mathcal{X},$$  \hspace{1cm} (5.34)

where

$$\alpha^{(n)}(x) = \exp \sum_{y} W(y|x) \log V^{(n)}(x, y), \quad x \in \mathcal{X}. \hspace{1cm} (5.35)$$

4. If $\max_x \log \left( \frac{Q^{(n+1)}(x)}{Q^{(n)}(x)} \right) < \epsilon$, stop iteration. Then the difference between the capacity and $\log \sum_x \alpha^{(n)}(x)$ will be less than $\epsilon$. Else, increment $n$ and go to step 2.

Note that it has been shown in [7] that

$$C - \log \sum_x \alpha^{(n)}(x) \leq \sum_x Q^*(x) \log \frac{Q^{(n+1)}(x)}{Q^{(n)}(x)} \leq \max_x \log \frac{Q^{(n+1)}(x)}{Q^{(n)}(x)}, \hspace{1cm} (5.36)$$

where $Q^*(x)$ is the optimal input distribution.

### 5.4 Simulation Results

In this section we present the simulation results of the non-asymptotic lower bound (5.26) on the capacity of a SISO zero-mean Gaussian fading channel with memory, where the memory is specified in (4.7) and (4.10).

Figure 5.2 shows the analytical upper and lower bounds that were already pictured in Figure 4.1 as well as a numerical lower bound using the Blahut-Arimoto algorithm. It is observed that at high SNR the numerical lower bound flattens. This results partly from the phenomenon of the power-inefficiency in the high-SNR regime, and partly from our restriction that an input is in $[\alpha A, A]$ which will lead to a bounded capacity as shown in [1, Theorem 4.3] and [2, Theorem 6.11]. Besides, another reason that could cause the flattening is due to the limited size of input and output alphabets in...
Figure 5.2: A numerical lower bound on the capacity of a SISO zero-mean Gaussian fading channel with memory. Depicted are analytical upper and lower bounds on the capacity (4.4), (4.6) for a fading process with spectral density (4.7), (4.10); a numerical lower bound using the Blahut-Arimoto algorithm; the asymptotic expansion (4.2); and the fading number (4.3).
Figure 5.3: The probability distributions of $|X|$ that achieve the numerical lower bound depicted in Figure 5.2 for various SNRs. For each plot, the horizontal axis represents the range of input magnitude corresponding to the optimal $\alpha$. For instance, the optimal $\alpha$ for SNR=20dB is 0.9, so the input is in $[0.9A, A]$, that is $[9, 10]$. 
5.4 Simulation Results

Figure 5.4: *The optimal α’s that achieve the numerical lower bound depicted in Figure 5.2 for various SNRs.*

simulated, which we have chosen to be simulation-feasible numbers. Indeed, the mutual information is upper-bounded by \( \log |\mathcal{X}| \) or \( \log |\mathcal{Y}| \), where \( |\mathcal{X}| \) and \( |\mathcal{Y}| \) denotes the cardinality of \( \mathcal{X} \) and \( \mathcal{Y} \), respectively.

Figure 5.3 depicts the probability distributions of the magnitude of inputs that achieve the numerical lower bound in Figure 5.2 at various SNRs. It is observed that the distributions of the input magnitude are more uniformly distributed at low and high SNRs, while at medium\(^1\) SNRs the distributions tend to be binary-distributed, where probability peaks on the edges of input magnitude, i.e., \( \alpha A \) and \( A \).

Figure 5.4 depicts the optimal α’s for various SNRs which we obtain by performing maximization in (5.7). It is observed that α is non-increasing in the SNR. In fact, it has been shown in [1] that if the input distribution,

\(^1\)Here “medium” means the values between 10 and 30 dB.
which is of magnitude constraint

\[ x_{\text{min}} \leq |x| \leq A, \]  

achieves the fading number given in (4.3), we will have \( x_{\text{min}} \) of the characteristic:

\[ \lim_{\text{SNR} \to \infty} \frac{\log x_{\text{min}}}{\log A} = 0. \]  

That is, as SNR increases, the ratio of \( x_{\text{min}} \) to \( A \) decreases, and this is in agreement with our simulation results.
Chapter 6

Numerical Upper Bound

In this chapter, we describe how we can apply the cutting-plane algorithm to obtain a numerical upper bound on the capacity. In Section 6.1 we derive an upper bound which is suitable for applying the cutting-plane algorithm. In Section 6.2 we briefly review the cutting-plane algorithm. In Section 6.3 we present the simulation results and give some comments.

6.1 Upper Bound on the Capacity and Application of the Algorithm

In the following, we describe the derivation steps to achieve an upper bound which is applicable for the cutting-plane algorithm. Note that we will consider inputs $X_k$ that are of magnitude constraint

$$|x_k| \leq A.$$  \hspace{1cm} (6.1)

To upper-bound the capacity given in the form of (2.7), we use the chain rule

$$\frac{1}{n} I(X^n_1; Y^n_1) = \frac{1}{n} \sum_{k=1}^{n} I(X^n_1; Y_k | Y^{k-1}) = 1.$$  \hspace{1cm} (6.2)

Then we upper-bound each of the individual terms in the sum by

$$I(X^n_1; Y_k | Y^{k-1}) = I(X^n_1, Y^k_1; Y_k) - I(Y_k; Y^{k-1}) \leq I(X^n_1, Y^k_1; Y_k) = I(X^k, Y^k_i; Y_k).$$  \hspace{1cm} (6.3)

$$I(X^n_1; Y_k | Y^{k-1}) = I(X^n_1, Y^k_1; Y_k) - I(Y_k; Y^{k-1}) \leq I(X^n_1, Y^k_1; Y_k) = I(X^k, Y^k_i; Y_k).$$  \hspace{1cm} (6.4)

$$I(X^n_1; Y_k | Y^{k-1}) = I(X^n_1, Y^k_1; Y_k) - I(Y_k; Y^{k-1}) \leq I(X^n_1, Y^k_1; Y_k) = I(X^k, Y^k_i; Y_k).$$  \hspace{1cm} (6.5)
where (6.3) follows from the chain rule; (6.4) follows due to the nonnegativity of mutual information; (6.5) follows from the absence of feedback, which results in the future inputs $X_{k+1}$ being independent of the present output $Y_k$ given $X_k$ and $Y_{k-1}$; inequality in (6.9) follows since we substitute $X_l$ by the largest input magnitude $A$, which results in the smallest noise effect and therefore makes mutual information biggest; (6.10) follows again from the chain rule; (6.11) follows because when $X_k$ is given, $Y_k$ does not depend on the past inputs $X_{k-1}$; (6.12) follows again from the chain rule; and finally (6.13) follows due to stationarity of the fading process.

The second term of (6.13) can be further simplified as follows

$$I \left( \left\{ H_l + \frac{Z_l}{A} \right\}_{l=-(k-1)}^{-1} ; Y_0 \right| X_0$$

$$\leq I \left( \left\{ H_l + \frac{Z_l}{A} \right\}_{l=-\infty}^{-1} ; Y_0 \right| X_0$$

$$\leq I \left( \left\{ H_l + \frac{Z_l}{A} \right\}_{l=-\infty}^{-1} ; H_0, Y_0 \right| X_0$$

(6.14)
where the first inequality follows by revealing the distant past; the second inequality follows revealing \( H_0 \); the subsequent equality follows because conditional on \( X_0 \) and \( H_0 \), the random variable \( Y_0 \) is independent of \( \left\{ H_l + \frac{Z_l}{A} \right\}_{l=-\infty}^{1} \); and the last equality follows from the fact that \( H_0 \) is Gaussian and the fact that \( H_0 \) is also conditionally Gaussian given the noisy past \( H_{-1} + \frac{Z_{-1}}{A}, H_{-2} + \frac{Z_{-2}}{A}, \ldots \). Note that \( \epsilon_{\text{pred}}^2 \left( \frac{\sigma^2_A}{A^2} \right) \) is computed using (3.2). Consequently, by combining (6.13) and (6.17) we have

\[
I \left( X^n_1; Y^n \middle| Y^{k-1}_1 \right) \leq I(X_0; Y_0) + \log \frac{1}{\epsilon_{\text{pred}}^2 \left( \frac{\sigma^2_A}{A^2} \right)}, \tag{6.18}
\]

and by (6.2) we then have

\[
\frac{1}{n} I \left( X^n_1; Y^n \right) \leq I(X_0; Y_0) + \log \frac{1}{\epsilon_{\text{pred}}^2 \left( \frac{\sigma^2_A}{A^2} \right)}. \tag{6.19}
\]

Hence, we obtain the upper bound of the capacity

\[
C(\text{SNR}) \leq \sup_{X_0} I(X_0; Y_0) + \log \frac{1}{\epsilon_{\text{pred}}^2 \left( \frac{\sigma^2_A}{A^2} \right)}. \tag{6.20}
\]

We can further simplify the first term of (6.20) by substituting \( X_0 \) by its polar representation \( Re^{i\Theta} \), where \( \Theta \sim \mathcal{U}([0, 2\pi]) \) from Lemma 3.2 and \( 0 \leq R \leq A \) from (6.1).

\[
I(X_0; Y_0) = I \left( X_0; H_0X_0 + Z_0 \right) \tag{6.21}
\]

\[
= I \left( Re^{i\Theta}; H_0Re^{i\Theta} + Z_0 \right) \tag{6.22}
\]

\[
= I \left( R; \Theta; H_0R + Z_0 \right) \tag{6.23}
\]

\[
= I \left( R; H_0R + Z_0 \right), \tag{6.24}
\]

where (6.21) follows from the definition of channel model in (2.1); (6.23) follows from the circular symmetry of \( H_0 \); (6.24) follows because \( H_0R + Z_0 \) does not depend on \( \Theta \). Combining (6.20) and (6.24) gives the capacity upper bound

\[
C(\text{SNR}) \leq \sup_R I(R; H_0R + Z_0) + \log \frac{1}{\epsilon_{\text{pred}}^2 \left( \frac{\sigma^2_A}{A^2} \right)}, \tag{6.25}
\]

where we will numerically compute the first term of (6.25) using the cutting-plane algorithm.
6.2 Cutting-Plane Algorithm Review

In this section we introduce the cutting-plane algorithm which is used to numerically compute the first term of (6.25). This section is largely based on [8].

We consider a discrete memoryless channel with input alphabet \( X \) of size \( n \), output alphabet \( Y \), and transition probability function \( W(y|x), x \in X, y \in Y \). (Notice that \( X \subset [0, A] \).) Let \( Q \) denote the input distribution and \( M^n \) denote the set of all probability distributions on the input, i.e.,

\[
M^n = \left\{ Q = (Q(x_1), \ldots, Q(x_n)) \mid Q(x_i) \geq 0, \sum_{i=1}^n Q(x_i) = 1 \right\}. \tag{6.26}
\]

The mutual information between \( X \) and \( Y \) for a given \( Q \in M^n \) is

\[
I(Q) = \sum_x Q(x) \left( \sum_y W(y|x) \log \frac{W(y|x)}{\sum_{x'} W(y|x') Q(x')} \right). \tag{6.27}
\]

Note that (6.27) is actually equivalent to (5.27). Therefore, the channel capacity subject to the peak-power constraint can be expressed as the value of the following nonlinear program:

\[
C = \max_Q I(Q) \quad \text{s.t.} \quad Q \in M^n. \tag{6.28}
\]

Now we describe how the cutting-plane algorithm works. The algorithm is initialized with an arbitrary distribution \( Q_0 \in M^n \), and inductively constructs a sequence of distributions as follows. At the \( k \)th stage of the algorithm, we are given \( k \) distributions \( \{Q_0, Q_1, \ldots, Q_{k-1}\} \subset M^n \). We then have the following procedures:

1. We define \( I_k(Q) \) as the piecewise-linear approximation of \( I(Q) \) as follows:

\[
I_k(Q) = \min_{0 \leq i \leq k-1} \sum_x Q(x) \left( \sum_y W(y|x) \log \frac{W(y|x)}{\sum_{x'} W(y|x') Q_i(x')} \right), \quad Q \in M^n. \tag{6.29}
\]

Then we can obtain the next distribution \( Q_k \) by\(^1\)

\[
Q_k = \arg \max \{ I_k(Q) : Q \in M^n \}. \tag{6.30}
\]

\(^1\)Step 1 can be done easily using the function \texttt{fminimax} in MATLAB Version 7.1. See appendix in Section B.3 for more detail.
2. Having obtained $Q_k$, we can compute $I(Q_k)$ using (6.27)

$$I(Q_k) = \sum_x Q_k(x) \left( \sum_y W(y|x) \log \frac{W(y|x)}{\sum_{x'} W(y|x')Q_k(x')} \right),$$

(6.31)

and besides we can obtain $I_k(Q_k)$ from (6.29).

3. Compute the difference $I_k(Q_k) - I(Q_k)$. If the difference is less than $\epsilon$, stop the iteration and both $I_k(Q_k)$ and $I(Q_k)$ are within $\epsilon$ of the capacity. Else, iteratively go to step 1.

Note that it has been shown in [8] that $I_k(Q_k)$ is always greater or equal to the capacity, while $I(Q_k)$ is always less or equal to the capacity. When $k$ tends to infinity, both $I_k(Q_k)$ and $I(Q_k)$ converge to $I(Q^*)$, where $Q^*$ is optimal.

6.3 Simulation Results

In this section we show the simulation results of a numerical upper bound (6.25) on the capacity of a SISO zero-mean Gaussian fading channel with memory, where the memory is specified in (4.7) and (4.10).

Figure 6.1 shows the analytical upper bound, the analytical lower bound, and the numerical lower bound that were already pictured in Figure 5.2 as well as a numerical upper bound using the cutting-plane algorithm. It is observed that the numerical upper bound flattens at high SNR. As in Section 5.4, we believe that this results partly from the phenomenon of the power-inefficiency in the high-SNR regime and partly from the limited size of input and output alphabets in the simulation. Due to these problems, the numerical upper bound is sort of limited and stops growing at high SNR. It should be noted however that the actual capacity does not cease growing as the SNR increases, but instead the capacity grows double-logarithmically in the SNR at high SNR as mentioned in Section 4.1.

Figure 6.2 depicts the probability distributions of the magnitude of inputs that achieve the numerical upper bound in Figure 6.1 at various SNRs. It is observed that the distributions of the input magnitude tend to be binary-distributed at 0 dB, where probability peaks at the two extremes of the input magnitude, i.e., 0 and $A$; distributions for SNRs between 20 to 60 are quite similar, where aside from two peaks on the edges, there is another peak at the magnitude of $A/3$. 

33
Figure 6.1: A numerical upper bound on the capacity of a SISO zero-mean Gaussian fading channel with memory. Depicted are analytical upper and lower bounds on the capacity (4.4), (4.6) for a fading process with spectral density (4.7), (4.10); a numerical lower bound using the Blahut-Arimoto algorithm; a numerical upper bound using the cutting-plane algorithm; the asymptotic expansion (4.2); and the fading number (4.3).
6.3 Simulation Results

Figure 6.2: The probability distributions of $|X|$ that achieve the numerical upper bound depicted in Figure 6.1 for various SNRs.
Chapter 7

Summary and Conclusion

In this report, we study the capacity of a stationary discrete-time Gaussian fading channel with memory, where both transmitter and receiver are cognizant of the fading law, but neither has access to the realizations of the fading process. Considering the fading process being regular and the inputs subject to a peak-power constraint, we propose a technique to derive a lower bound on the capacity of the channel, and a technique to derive an upper bound on the capacity as well. In addition, we apply the Blahut-Arimoto algorithm to obtain a numerical lower bound, and apply the cutting-plane algorithm to obtain a numerical upper bound.

To obtain the lower bound, we start with the definition of the capacity (2.7), next get rid of the memory of the fading channel by introducing a prediction technique on the fading channel, and then step by step lower-bound the mutual information until the expression of the lower bound (5.26) is derived. After deriving the expression, the Blahut-Arimoto algorithm is applied to numerically compute the optimization problem in (5.26), and finally a numerical lower bound is obtained. It has been shown that in our channel model the input distributions that achieve the lower bound can be always assumed to be circularly symmetric, and at various SNRs the distributions of the input magnitude present various characteristics. It has also been observed that as the SNR increases the range of the input alphabet gets wider.

To obtain the upper bound, on the other hand, we also start with the definition of the capacity (2.7) and then step by step upper-bound the mutual information until the expression of the upper bound (6.25) is derived. After deriving the expression, the cutting-plane algorithm is applied to numeri-
cally compute the optimization problem in (6.25), and finally a numerical upper bound is obtained. It has been shown that in our channel model the input distributions that achieve the upper bound can be always assumed to be circularly symmetric. At various SNRs, the distributions of the input magnitude present various characteristics.

It should be pointed out that both the numerical upper and lower bounds stop growing at high SNR probably because of numerical problems and because the mutual information is limited by the finite number of discrete inputs and outputs. This effect is more obvious at high SNR because the number of mass points is so large that it requires large values of alphabet size in order to keep track of the information, otherwise the numerical limitation would take effect before the real bounds do.

Consequently, a possible extension for future work is to increase the alphabet sizes to a reasonably high level, and then to observe the behavior of the generated numerical bounds.
Appendix A

Simplification of Entropy: Sum of A Uniform-Phase and A Gaussian Distribution

In this appendix, we give in detail the derivation process that simplifies the differential entropy of the sum of a uniform-phase distribution and a Gaussian distribution, which are independent of each other, i.e., $W \perp \Theta$, and finally derive a form which is easier to compute numerically. We use the following notation

\[ \Theta \sim U([0,2\pi]), \quad (A.1) \]
\[ W \sim N_C(0,\sigma_W^2), \quad (A.2) \]
\[ d \geq 0, \quad (A.3) \]

where $d$ is a constant. The goal is to simplify

\[ h(de^{i\Theta} + W). \quad (A.4) \]

Let

\[ X = de^{i\Theta} + W = X_1 + iX_2. \quad (A.5) \]

Thus

\[ X_1 = d\cos \Theta + W_1 \quad (A.6) \]
\[ X_2 = d\sin \Theta + W_2, \quad (A.7) \]
where $W_1 \sim \mathcal{N}_R(0, \sigma_W^2/2)$ and $W_2 \sim \mathcal{N}_R(0, \sigma_W^2/2)$. Next, we do a transformation of random variables:

\[
\begin{align*}
X_1 &= d \cos \Theta + W_1 \\
X_2 &= d \sin \Theta + W_2 \\
X_3 &= \Theta
\end{align*}
\Rightarrow
\begin{align*}
W_1 &= X_1 - d \cos X_3 \\
W_2 &= X_2 - d \sin X_3 .
\end{align*}
\tag{A.8}
\]

To transform, we first need the Jacobian matrix obtained from (A.8):

\[
J = \begin{bmatrix}
1 & 0 & -d \sin \Theta \\
0 & 1 & d \cos \Theta \\
0 & 0 & 1
\end{bmatrix},
\tag{A.9}
\]

and thus $\det(J) = 1$. Therefore, the joint distribution of $X_1$, $X_2$, and $X_3$

\[
f_{X_1 X_2 X_3}(x_1, x_2, x_3) = \frac{1}{2\pi} \cdot \frac{1}{\pi \sigma_W^2} \exp \left( -\frac{1}{\sigma_W^2} (w_1^2 + w_2^2) \right)
\tag{A.10}
\]

\[
= \frac{1}{2\pi^2 \sigma_W^2} \cdot \exp \left( -\frac{1}{\sigma_W^2} (x_1^2 + x_2^2 + d^2 - 2d(x_1 \cos x_3 + x_2 \sin x_3)) \right)
\tag{A.11}
\]

\[
= \frac{1}{2\pi^2 \sigma_W^2} \cdot \exp \left( -\frac{x_1^2 + x_2^2}{\sigma_W^2} \right) \exp \left( -\frac{d^2}{\sigma_W^2} \right)
\cdot \exp \left( \frac{2d}{\sigma_W^2} (x_1 \cos x_3 + x_2 \sin x_3) \right).
\tag{A.12}
\]

Since $X_3 = \Theta \sim \mathcal{U}([0, 2\pi])$, the joint distribution of $X_1$ and $X_2$ is

\[
f_{X_1 X_2}(x_1, x_2) = \int_0^{2\pi} f_{X_1 X_2 X_3}(x_1, x_2, x_3) \, dx_3
\tag{A.13}
\]

\[
= \int_0^{2\pi} \frac{1}{2\pi^2 \sigma_W^2} \exp \left( -\frac{d^2}{\sigma_W^2} \right) \exp \left( -\frac{x_1^2 + x_2^2}{\sigma_W^2} \right)
\cdot \exp \left( \frac{2d}{\sigma_W^2} x_1 \cos x_3 + \frac{2d}{\sigma_W^2} x_2 \sin x_3 \right) \, dx_3,
\tag{A.14}
\]

where the underbraced term can be further simplified as follows

\[
\int_0^{2\pi} \exp \left( \frac{2d}{\sigma_W^2} x_1 \cos x_3 + \frac{2d}{\sigma_W^2} x_2 \sin x_3 \right) \, dx_3
\]

\[
= \int_0^{2\pi} \exp \left( \frac{2d}{\sigma_W^2} \sqrt{x_1^2 + x_2^2} \sin \left( \arctan \frac{x_1}{x_2} + x_3 \right) \right) \, dx_3
\tag{A.15}
\]

\[40\]
\[= \int_0^{2\pi} \exp\left(\frac{2dr}{\sigma_W^2} \sin x_3\right) dx_3 \bigg|_{r=\sqrt{x_1^2 + x_2^2}} \]  
(A.16)

\[= 2 \int_0^{\pi} \exp\left(\frac{2dr}{\sigma_W^2} \cos x_3\right) dx_3 \]  
(A.17)

\[= 2\pi I_0\left(\frac{2dr}{\sigma_W^2}\right) \bigg|_{r=\sqrt{x_1^2 + x_2^2}}, \]  
(A.18)

where \(I_0(\cdot)\) refers to the Modified Bessel function of the first kind. Now, we need to transform the joint distribution

\[f_{X_1,X_2}(x_1,x_2) = \frac{1}{\pi\sigma_W^2} \exp\left(-\frac{d^2}{\sigma_W^2}\right) \exp\left(-\frac{x_1^2 + x_2^2}{\sigma_W^2}\right) I_0\left(\frac{2dr}{\sigma_W^2}\right) \bigg|_{r=\sqrt{x_1^2 + x_2^2}}, \]  
(A.19)

to its polar form

\[f_{R,\phi}(r,\phi) = \frac{r}{\pi\sigma_W^2} \exp\left(-\frac{d^2}{\sigma_W^2}\right) \exp\left(-\frac{r^2}{\sigma_W^2}\right) I_0\left(\frac{2dr}{\sigma_W^2}\right), \]  
(A.20)

which refers to the fact that \(X\) is circularly symmetric, since (A.20) does not depend on \(\phi\). Hence,

\[h(X) = \log \pi + h(|X|^2). \]  
(A.21)

(See [2, Lemma A.24] for a proof.) From (A.20), we further obtain

\[f_R(r) = \int_0^{2\pi} f_{R,\phi}(r,\phi) d\phi \]  
(A.22)

\[= \frac{2r}{\sigma_W^2} \exp\left(-\frac{d^2}{\sigma_W^2}\right) \exp\left(-\frac{r^2}{\sigma_W^2}\right) I_0\left(\frac{2dr}{\sigma_W^2}\right). \]  
(A.23)

Finally, let \(Z = |X|^2 = R^2\), so \(dz = 2r \, dr\), and then we have

\[f_Z(z) = f_R(r) \cdot \frac{1}{2r} \]  
(A.24)

\[= \frac{1}{\sigma_W^2} \exp\left(-\frac{z}{\sigma_W^2}\right) \exp\left(-\frac{z}{\sigma_W^2}\right) I_0\left(\frac{2d\sqrt{z}}{\sigma_W^2}\right). \]  
(A.25)

Now, we are able to derive \(h(|X|^2)\) by the definition of differential entropy:

\[h(|X|^2) = h(Z) \]  
(A.26)

\[= - \int_0^{\infty} f_Z(z) \log f_Z(z) \, dz \]  
(A.27)

\[= - \int_0^{\infty} f_Z(z) \]
Appendix A  Simplification of Entropy Calculation

\[
\begin{align*}
&\log \left[ \frac{1}{\sigma_W^2} \exp \left( -\frac{d^2}{\sigma_W^2} \right) \exp \left( -\frac{z}{\sigma_W^2} \right) I_0 \left( \frac{2d\sqrt{z}}{\sigma_W^2} \right) \right] dz \quad (A.28) \\
&= \log \sigma_W^2 + \frac{d^2}{\sigma_W^2} + \frac{1}{\sigma_W^2} \int_0^{\infty} z f_Z(z) dz \\
&\quad - \int_0^{\infty} f_Z(z) \log I_0 \left( \frac{2d\sqrt{z}}{\sigma_W^2} \right) dz, \quad (A.29)
\end{align*}
\]

where the third term of the last equation is the expected value of \( Z \) divided by \( \sigma_W^2 \), which is given as

\[
\int_0^{\infty} z f_Z(z) dz = E[Z] \quad (A.30)
\]

\[
= E \left[ |de^{i\Theta} + W|^2 \right] \quad (A.31)
\]

\[
= d^2 + E \left[ de^{-i\Theta} W \right] + E \left[ de^{i\Theta} W^\ast \right] + E \left[ |W|^2 \right] \quad (A.32)
\]

\[
= d^2 + \sigma_W^2. \quad (A.33)
\]

Eventually, we obtain

\[
\begin{align*}
&h \left( de^{i\Theta} + W \right) \\
&= h(X) \quad (A.34) \\
&= \log \pi + h \left( |X|^2 \right) \quad (A.35) \\
&= \log \left( \pi \sigma_W^2 \right) + 1 + 2 \frac{d^2}{\sigma_W^2} \\
&\quad - \frac{1}{\sigma_W^2} \exp \left( -\frac{d^2}{\sigma_W^2} \right) \cdot \int_0^{\infty} \exp \left( -\frac{z}{\sigma_W^2} \right) I_0 \left( \frac{2d\sqrt{z}}{\sigma_W^2} \right) \log I_0 \left( \frac{2d\sqrt{z}}{\sigma_W^2} \right) dz. \quad (A.36)
\end{align*}
\]
Appendix B

Program Structure

In this appendix, we briefly describe the structure of the programs that we use to do the simulation and give the explanations on those tricks that we have used while programming.

We divide the programs into three parts: the programs to plot the previous results in Chapter 4, the programs to compute the numerical lower bound using the Blahut-Arimoto algorithm in Chapter 5, and the programs to compute the numerical upper bound using the cutting-plane algorithm in Chapter 6.

B.1 Previous Results

This part contains three files with which we can plot the analytical results shown in Figure 4.1.

- `simplot_upperbound_rician.m` is used to calculate the capacity upper bound in the memoryless fading case for various SNRs, i.e., Equation (4.5). Detailed programming information and the trick used to compute the maximum is described in the file.

- `simplot_lowerbound.m` is to calculate capacity lower bound for various SNRs, i.e., Equation (4.6).

- `Plot_Bounds.m` is to calculate Equation (4.4) and besides plot them.
B.2 Blahut-Arimoto Algorithm

This part contains one main program and four supporting function files. In these files we implement the Blahut-Arimoto algorithm to do the optimization in Equation (5.26). After having the generated lower bound values, we can plot them as in Figure 5.2 along with the previous results for comparison.

- **BAA.m** is the main program that takes care of creating all needed data to run the algorithm, and obtaining lower bound values. In this program, there are two main iterations: one for various SNRs and the other for various $\alpha$’s\(^1\). The four supporting functions are:
  - **GeneratePd_1** is to generate the modified discrete Rayleigh distributed vector, i.e., the $P_{|\hat{D}|}(|\hat{d}|)$ in Equation (5.10). The idea of generating such a vector is illustrated in Figure 5.1.
  - **CreateMatrix_BAA** is to create the transition matrices of the channel. Detailed explanations can be found in the file.
  - **MI_theta_BAA** is to calculate the second term of Equation (5.26).
  - **MI_amp_BAA** is the function that runs the Blahut-Arimoto algorithm to compute the first term of Equation (5.26).

Finally, we will have a matrix named $C$, in which each row contains lower bound values for various $\alpha$’s given a certain value of SNR. Therefore, by picking the largest value of each row, we have a sequence of values, and those are the lower bound values we want.

B.3 Cutting-Plane Algorithm

This part contains one main program, one supporting function to create transition matrices of the channel, and one supporting function to implement the cutting-plane algorithm to do the optimization in Equation (6.25). After obtaining the generated upper bound values, we can plot them as in Figure 6.1.

- **cutting_plane.m** is the main program that contains only one main iteration due to various SNRs. The other two supporting functions are:

\(^1\alpha\) here refers to the parameter of (5.26), i.e., $0 \leq \alpha \leq 1$.\n
44
B.3 Cutting-Plane Algorithm

– CreateMatrix_cut is to create the transition matrices of the channel. The basic idea of this function is the same as the function CreateMatrix_BAA in Blahut-Arimoto part.

– MI_amp_cut is the function that runs the cutting-plane algorithm to compute the first term of Equation (6.25). An important subroutine, findmaxmin, is contained in this function, which is the most crucial part in this algorithm.

* findmaxmin is to find the distribution $Q_k$ in (6.30) and the corresponding maximum of minimum values, i.e., $I_k(Q_k)$.

Recall from Section 6.2 that at the $k$th stage, we are given $k$ distributions $\{Q_0, Q_1, \cdots, Q_{k-1}\} \subset M^n$. Now for every distribution, $Q$, Equation (6.29) chooses the smallest value out of $k$ choices, i.e., $I_k(Q)$. We theoretically have an infinite number of distributions, so $I_k(Q)$ can be thought of as a continuous function with respect to $Q$. $Q_k$ is the maximizer that maximizes $I_k(Q)$ over $Q$, and $I_k(Q_k)$ is the corresponding maximum value. This is what the subroutine findmaxmin does. Adapted from the MATLAB function fminimax which is exactly the opposite action of findmaxmin, we only need to input the vectors that are calculated by the expression

$$
\left( \sum_y W(y|x) \log \frac{W(y|x)}{\sum_{x'} W(y|x')Q_k(x')} \right)
$$

in Equation (6.29). Then the distribution and the maximum of minimum values will be returned.

Finally, combining prediction error values and the values calculated from algorithm gives the upper bound values we want.
Bibliography


