# Estimation of Bit Error Rate of any digital Communication System

Jia Dong

**N° d'ordre** : 2013telb0295

**Sous le sceau de l'Université européenne de Bretagne**

# Télécom Bretagne

**En accréditation conjointe avec l'Ecole Doctorale Sicma**

---

# Estimation of Bit Error Rate of any digital Communication System

---

## Thèse de Doctorat

Mention : Sciences et Technologies de l'Information et de la Communication

Présentée par **Jia DONG**

Département : Signal et Communications

Laboratoire : Labsticc Pôle : CACS

Directeur de thèse : Samir SAOUDI

Soutenue le 18 décembre 2013

**Jury :**

M. Ghais EL ZEIN, Professeur, INSA Rennes (Rapporteur)
M. Jean-Pierre CANCES, Professeur, ENSIL (Rapporteur)
M. Samir SAOUDI, Professeur, Télécom Bretagne (Directeur de thèse, encadrant)
M. Frédéric GUILLOUD, Maître de Conférences, Télécom Bretagne (Co-encadrant)
M. Charles TATKEU, Chargé de recherche, HDR, IFSTTAR - Lille (Examinateur)
M. Emmanuel BOUTILLON, Professeur, UBS (Examinateur)

# Remerciements

Ce travail de thèse a été effectué au sein du département Signal et Communications de TELECOM Bretagne. J'adresse tout d'abord mes remerciements au PRACOM et à la région de Bretagne pour l'aide financière qui m'a été accordée pendant ces trois années de thèse.

Je tiens à exprimer ma reconnaissance et ma gratitude à mon directeur de thèse et mon encadrant, M. Samir SAOUDI Professeur à TELECOM Bretagne, pour m'avoir proposé ce très riche sujet, en avoir assuré la direction avec son soutien permanent, pour la diversité des discussions et pour ses encouragements tout au long de cette thèse.

Je tiens à remercier mon co-encadrant, M. Frédéric GUILLOUD Maître de Conférence à TELECOM Bretagne, pour son aide fructueuse, ses conseils et recommandations.

Je souhaite remercier M. Ghais EL ZEIN Professeur à l'INSA Rennes et M. Jean-Pierre CANCES Professeur à l'ENSIL, pour l'honneur qu'ils m'ont fait de rapporter mon manuscrit ainsi que pour leurs remarques intéressantes.

Je remercie aussi M. Charles TATKEU Chargé de recherche à l'IFSTTAR-Lille et M. Emmanuel BOUTILLON Professeur à l'UBS d'avoir accepté d'examiner mon travail de thèse.

Toutes mes affections et mes remerciements vont à M. Hexin LIU qui m'a toujours soutenu et fourni l'aide fructueuse aux niveaux professionnel et privé.

Je ne peux pas oublier de remercier M. Ramesh PYNDIAH Chef du département Signal & Communications à TELECOM Bretagne pour m'avoir accueilli dans son département. Je voudrais remercier également tous les membres du département Signal & Communications pour m'avoir aidée au cours de ma thèse et fourni un environnement très agréable.

Je remercie mes amis pour leur amitié.

Enfin, je tiens à exprimer ma pronfonde affection à ma grand-mère, mon grand-père et mes parents pour leur amour. Je les aime de tout mon cœur.

# Contents

**Remerciements** iii

**Contents** viii

**Abstract** ix

**Résumé** xi

**Acronyms** xiii

**1 Introduction** 1

1.1 Overview . . . . . 1

1.2 Requirement of real-time on-line Bit Error Rate estimation . . . . . 2

1.3 Thesis organization . . . . . 2

**2 State of the art for Bit Error Rate estimation** 5

2.1 Overview of conventional Bit Error Rate estimation techniques . . . . . 5

2.1.1 Monte-Carlo simulation . . . . . 5

2.1.2 Importance Sampling method . . . . . 7

2.1.3 Tail Extrapolation method . . . . . 9

2.1.4 Quasi-analytical estimation . . . . . 10

2.1.5 BER estimation based on Log-Likelihood Ratio . . . . . 10

2.1.6 Conclusion of BER estimation methods . . . . . 12

2.2 Probability Density Function estimation . . . . . 12

2.2.1 Introduction to PDF estimation . . . . . 13

2.2.2 Parametric density estimation : Maximum Likelihood Estimation 14

2.2.3 Non-parametric density estimation . . . . . 16

2.2.3.1 Empirical density estimation . . . . . 16

2.2.3.2 Histogram . . . . . 17

2.2.3.3 General formulation of non-parametric density estimation 18
2.2.3.4 Introduction to Kernel Density Estimation . . . . . 19
2.2.3.4.1 Naïve estimator : Parzen window . . . . . 19
2.2.3.4.2 Smooth Kernels . . . . . 20
2.2.4 Semi-parametric density estimation . . . . . 21
2.2.4.1 Introduction to Gaussian Mixture Model . . . . . 22
2.2.4.2 Difficulties of Mixture Models . . . . . 22
2.3 BER calculation with PDF estimate . . . . . 23
2.3.1 Theoretical BER : BER estimation based on parametric PDF estimation . . . . . 24
2.3.2 Practical situation : necessity of non-parametric or semi-parametric PDF estimation . . . . . 26
2.4 Conclusion . . . . . 28

**3 Bit Error Rate estimation based on Kernel method 29**
3.1 Properties of Kernel-based PDF estimator . . . . . 29
3.1.1 Bias and variance of Kernel estimator . . . . . 30
3.1.2 MSE and IMSE of Kernel estimator . . . . . 31
3.1.3 Kernel selection . . . . . 31
3.1.4 Bandwidth (smoothing parameter) selection . . . . . 32
3.1.4.1 Subjective selection . . . . . 33
3.1.4.2 Selection with reference to some given distribution : optimal smoothing parameter . . . . . 35
3.2 BER estimation based on Kernel method . . . . . 39
3.2.1 PDF estimation based on Kernel method . . . . . 39
3.2.2 Smoothing parameters optimization in practical situation . . . . 40
3.2.2.1 Curve fitting method . . . . . 42
3.2.2.2 Newton's method . . . . . 45
3.2.3 BER calculation with Kernel-based PDF estimates . . . . . 47
3.2.4 MSE of Kernel-based soft BER estimator . . . . . 48
3.3 Simulation results of BER estimation based on Kernel method . . . . . 48
3.3.1 Sequence of BPSK symbol over AWGN and Rayleigh channels . 49
3.3.2 CDMA system . . . . . 53
3.3.2.1 Standard receiver . . . . . 54
3.3.2.2 Decorrelator-based receiver . . . . . 56
3.3.3 Turbo coding system . . . . . 59
3.3.4 LDPC coding system . . . . . 61

3.4 Conclusion . . . . . 64

**4 Bit Error Rate estimation based on Gaussian Mixture Model 67**

4.1 Missing data of component assignment . . . . . 67

4.1.1 K-means clustering . . . . . 67

4.1.1.1 Principle of K-means clustering . . . . . 67

4.1.1.2 K-means clustering algorithm : KMA . . . . . 68

4.1.2 Probabilistic clustering as a mixture of models . . . . . 69

4.2 BER estimation based on Gaussian Mixture Model . . . . . 71

4.2.1 Introduction to Expectation-Maximization algorithm . . . . . 71

4.2.1.1 Jensen's inequality . . . . . 71

4.2.1.2 Principle of Expectation-Maximization algorithm . . . 72

4.2.2 Expectation-Maximization algorithm for Gaussian Mixture Model 73

4.2.2.1 Estimation step . . . . . 74

4.2.2.2 Maximization step . . . . . 74

4.2.2.2.1 Calculation of $\mu_k$ . . . . . 74

4.2.2.2.2 Calculation of $\sigma_k^2$ . . . . . 75

4.2.2.2.3 Calculation of $\alpha_k$ . . . . . 75

4.2.3 Example of GMM-based PDF estimation using Expectation-Maximization algorithm . . . . . 76

4.2.4 BER calculation with GMM-based PDF estimates . . . . . 78

4.2.5 Optimal choice of the number of Gaussian components . . . . . 79

4.2.6 Conclusion of Gaussian Mixture Model-based BER estimation using Expectation-Maximization algorithm . . . . . 81

4.3 Simulation results of BER estimation based on Gaussian Mixture Model 82

4.3.1 Sequence of BPSK symbol over AWGN and Rayleigh channels . 83

4.3.2 CDMA system with decorrelator-based receiver . . . . . 88

4.3.3 Turbo coding system . . . . . 92

4.3.4 LDPC coding system . . . . . 93

4.4 Conclusion . . . . . 95

**5 Unsupervised Bit Error Rate Estimation 97**

5.1 Unsupervised BER estimation based on Stochastic Expectation Maximization algorithm using Kernel method . . . . . 97

5.1.1 Initialization . . . . . 98

5.1.2 Estimation step . . . . . 99

5.1.3 Maximization step . . . . . 99

5.1.4 Stochastic step . . . . . 100
5.1.5 Conclusion for SEM-based unsupervised BER estimation using Kernel method . . . . . 101
5.2 Unsupervised BER estimation based on Stochastic Expectation Maximization algorithm using Gaussian Mixture Model . . . . . 102
5.3 Simulation results . . . . . 104
5.3.1 Sequence of BPSK symbol over AWGN channel . . . . . 104
5.3.2 CDMA system with standard receiver . . . . . 108
5.3.3 Turbo coding and LDPC coding systems . . . . . 111
5.4 Conclusion . . . . . 114

**Conclusion and perspectives** **115**

**Appendix** **119**
Appendix A . . . . . 119

**Résumé de la thèse** **123**

**Publications** **133**

**Bibliography** **140**

# Abstract

This thesis is related to the Bit Error Rate (BER) estimation for any digital communication system.

In many designs of communication systems, the BER is a Key Performance Indicator (KPI). The popular Monte-Carlo (MC) simulation technique is well suited to any system but at the expense of long time simulations when dealing with very low error rates. In this thesis, we propose to estimate the BER by using the Probability Density Function (PDF) estimation of the soft observations of the received bits.

First, we have studied a non-parametric PDF estimation technique named the Kernel method. Simulation results in the context of several digital communication systems are proposed. Compared with the conventional MC method, the proposed Kernel-based estimator provides good precision even for high SNR with very limited number of data samples.

Second, the Gaussian Mixture Model (GMM), which is a semi-parametric PDF estimation technique, is used to estimate the BER. Compared with the Kernel-based estimator, the GMM method provides better performance in the sense of minimum variance of the estimator.

Finally, we have investigated the blind estimation of the BER, which is the estimation when the sent data are unknown. We denote this case as unsupervised BER estimation. The Stochastic Expectation-Maximization (SEM) algorithm combined with the Kernel or GMM PDF estimation methods has been used to solve this issue. By analyzing the simulation results, we show that the obtained BER estimate can be very close to the real values. This is quite promising since it could enable real-time BER estimation on the receiver side without decreasing the user bit rate with pilot symbols for example.

# Résumé

Dans cette thèse, nous proposons d'étudier les estimations du taux d'erreurs binaire (BER) pour n'importe quel système de communications numériques.

Dans la majorité des cas, le BER est un paramètre clé de la conception du système. Les simulations de type Monte-Carlo (MC) sont alors classiquement utilisées pour estimer les taux d'erreurs ; mais elles se révèlent très côuteuse en temps de simulation lorsque les taux d'erreurs sont très faibles. De plus, elles ne sont pas applicables au cas le taux d'erreurs doit être estimé en aveugle au niveau du récepteur. Par conséquent, nous proposons de mettre en œuvre des techniques d'estimation de densités de probabilités (PDF) des observations souples en sortie du récepteur afin de réduire le nombre d'échantillons nécessaires pour estimer les taux d'erreurs binaires du système de communications numériques étudié.

Dans un premier temps, nous avons étudié l'estimation non-paramétrique appelée "méthode du noyau" (Kernel) pour estimer la PDF. Le BER est calculé par intégration (analytique) de la PDF estimée. Les résultats des simulations pour différents systèmes de communications numériques ont été analysés. Par rapport à la méthode MC, la méthode du noyau permet d'obtenir une estimation plus précise.

Ensuite, nous avons utilisé le modèle de mélanges de gaussiennes (GMM), qui est une méthode semi-paramétrique souvent employées en reconnaissance de forme, pour estimer le BER. Par rapport à la méthode du noyau, la méthode GMM permet de réaliser les meilleures performances dans le sens de la minimisation de la variance de l'estimateur.

Enfin, nous avons étudié l'estimation du BER de façon aveugle, c'est à dire sans utiliser la connaissance des informations binaires transmises. Cette estimation est basée sur l'algorithme SEM (Stochastic Expectation-Maximization), en combinaison avec les méthodes du noyau et de la GMM vues précédemment. A partir des résultats des simulations, nous constatons que le BER estimé de façon aveugle peut être très proche de la valeur réelle tout en utilisant peu d'échantillons. Cette méthode pourrait s'avérer très avantageuse pour l'estimation en ligne et en temps réel du BER au niveau du récepteur.

# Acronyms

| | |
|---|---|
| **ARQ** | Automatic Repeat Request |
| **AWGN** | Additive White Gaussian Noise |
| **BER** | Bit Error Rate |
| **BP** | Belief Propagation |
| **BPSK** | Binary Phase Shift Keying |
| **CCDF** | Complementary Cumulative Distribution Function |
| **CDF** | Cumulative Distribution Function |
| **CDMA** | Code-division Multiple-access |
| **EM** | Expectation-Maximization |
| **ENS** | Equivalent Noise Source |
| **FEC** | Forward Error Correction |
| **FPRA** | Floating-Point Relative Accuracy |
| **GEC** | Generalized Exponential Class |
| **GM** | Gaussian Mixture |
| **GMM** | Gaussian Mixture Model |
| **I.I.D.** | Independent and Identically Distributed |
| **IS** | Importance Sampling |
| **KDE** | Kernel Density Estimation |
| **KEYS** | Kernel Estimating Your Shapes |
| **KMA** | K-Means Algorithm |
| **kNN** | k-Nearest Neighbor |
| **KPI** | Key Performance Indicator |
| **LDPC** | Low Density Parity Check |
| **LLR** | Log-likelihood Ratio |
| **MAI** | Multiple Access Interferences |
| **MAP** | Maximum A-Posteriori |
| **MC** | Monte-Carlo |
| **MFSK/FH** | Multiple Frequency Shift Keying/Frequency Hopping |

| | |
|---|---|
| **MIMO** | Multiple Input Multiple Output |
| **ML** | Maximum Likelihood |
| **MLE** | Maximum Likelihood Estimation |
| **NF** | Noise Figure |
| **OLS** | Ordinary Least Squares |
| **PDF** | Probability Density Function |
| **QA** | Quasi-Analytical |
| **QC-LDPC** | Quasi-Cyclic-Low Density Parity Check |
| **QoS** | Quality of Service |
| **Rx** | Receiver |
| **SEM** | Stochastic Expectation-Maximization |
| **SISO** | Soft-Input Soft-Output |
| **SNR** | Signal-to-Noise Ratio |
| **SOVA** | Soft-Output Viterbi Algorithm |
| **SS** | Spread Spectrum |
| **TE** | Tail Extrapolation |
| **Tx** | Transmitter |
| **WCSS** | Within-Cluster Sum of Squares |

CHAPTER 1

# Introduction

## 1.1 Overview

In analog communication systems, the distortion of the signal was used as a Key Performance Indicator (KPI) of the quality of the transmission. In digital communications, the KPI of the quality of the transmission is the Bit Error Rate (BER).

The BER represents the probability of receiving an erroneous bit. Thus, BER gives end-to-end performance measurement and quantifies the reliability of the entire communication system from "bits in" to "bits out". In other words, BER depends on all the components of the communication system and may be affected by several factors, e.g., transmission channel noise, distortion, attenuation, fading, synchronization problems and interferences. But it can also be improved by implementing error correction schemes, resource allocation mechanisms, power control or link adaptation schemes.

Recently, Bit Error Rate estimation techniques have attracted much attention. BER estimation is useful in Automatic Repeat Request (ARQ) and Hybrid ARQ systems [SLS03, MZB01], in which packets may be retransmitted if the BER estimates are too high. Other applications may be found in [HW01, VWR+01, HC06]. Especially, on-line and real-time BER estimation for digital communication systems is of some practical utility. Accurate BER estimates can be used as meaningful feedback quality criteria [LH01]. As an example, power control mechanisms in digital communication systems typically use the BER as a quality measure feedback for maintaining transmit power at minimum required levels to maintain a desired Quality of Service (QoS) (i.e., the transmitted Radio Frequency signal power can be decreased or increased till the BER estimates reach a predefined criterion [Man06]). Hence, the BER estimation is very important since it is not only the most important quality criteria for digital communication systems but also the feedback information which enables system-level optimizations.

In general, the BER cannot be analytically calculated and need to be estimated. The popular Monte-Carlo (MC) simulation technique is convenient for estimating BER by dividing the number of incorrect received bits by the total number of transmitted bits during a given time. Unfortunately, it is well known that the Monte-Carlo method has a very high computational cost for very low BER. Consequently, the MC technique is not suitable for many applications, such as real-time and on-line BER estimation.

## 1.2 Requirement of real-time on-line Bit Error Rate estimation

As presented above, the BER estimate can be used as a feedback for many practical communication systems to perform system-level functions, such as power control, resource allocation, link adaption [AG99]. In these applications, the BER is required to be on-line estimated. Thus, the BER estimation faces the following challenges :

- The BER estimator should be immune to the transmitter/receiver scheme, channel condition, interference model and other information about the entire communication system, although these elements impacts the system performance and the reliability of the BER estimate. In other words, we should find a technique that provides reliable BER estimate for any communication system ;
- The BER should be estimated with the smallest possible number of samples. Moreover, BER should be estimated by observing the soft values, and, if possible, without having pilots symbols or any known bits.

This thesis is mainly motivated by the above challenges. In this report, we shall present some new BER estimators based on Samir Saoudi's works [STG09, STTP10, SAIM11] :

- First, we will present two new BER estimation techniques based on probability density function (PDF) estimations. PDF estimations will be based on either non-parametric estimation (Kernel method) and on semi-parametric estimation (Gaussian Mixture Model). We assume that the estimators do not have any information about the transceiver scheme and about the channel model. However, the transmitted data bits are still assumed to be known, which means that the estimation takes into account, for each received sample, what the sent bit is ;
- Second, we will present a novel unsupervised BER estimator : the estimation does not require the knowledge of the transmitted information bits ; hence the estimation of the BER is blind. Only the soft observations are used to meet the requirements of on-line BER estimation.

## 1.3 Thesis organization

Chapter 2 introduces the basic idea of the conventional Monte-Carlo method and the related modified MC-based techniques : the Importance Sampling method, the Tail Extrapolation method and the Quasi-Analytical method. We will also present the BER estimation method based on Log-Likelihood Ratio (LLR). Then we will show that BER estimation can be considered equivalent to estimating the Probability Density Functions of the receiver's soft outputs. We will give a brief introduction of the non-parametric method and the semi-parametric method. Special attention will be paid to the Kernel method and the Gaussian Mixture Model which are the basis of the fast BER estimation techniques studied in Chapter 3, 4 and 5.

In Chapter 3, we will investigate the use of Kernel method for BER estimation in a supervised fashion. We will show that the key parameter which determines the performance of BER estimate is the bandwidth of the Kernel-based estimator. Some methods for the bandwidth selection will be proposed and the behavior of the proposed supervised Kernel-based estimator for different communication systems will be reported. In this report, we focus on four systems :

1. Binary Phase-Shift Keying (BPSK) signal over Additive White Gaussian Noise (AWGN) and Rayleigh channels ;
2. CDMA system with standard and decorrelator-based receivers ;
3. Turbo coding system ;
4. Low Density Parity Check (LDPC) coding system.

In Chapter 4, we address the semi-parametric method denoted Gaussian Mixture Model. We will report the performance of BER estimation for the same systems as presented in the previous chapter.

In Chapter 5, we extend the BER estimation techniques studied in Chapter 3 and 4 without the knowledge of transmitted information data. This scheme is denoted "unsupervised BER estimation" and it is opposed to the "supervised" case described in Chapter 3 and 4. The unsupervised BER estimation is based on a Stochastic Expectation-Maximization (SEM) algorithm. Then we will study the behavior of the unsupervised BER estimator.

Chapter 6 concludes the whole dissertation and gives some perspectives for future research.

CHAPTER 2

# State of the art for Bit Error Rate estimation

## 2.1 Overview of conventional Bit Error Rate estimation techniques

Before focusing on the fast Bit Error Rate estimation techniques, it is necessary to give a brief introduction of the conventional BER estimation methods. In this section, we shall give a tutorial exposition of some famous techniques : the well-known Monte-Carlo simulation, the modified MC-based estimation methods and the Log-Likelihood Ratio-based BER estimation technique.

### 2.1.1 Monte-Carlo simulation

The Monte-Carlo method is the most widely used technique for estimating the BER of a communication system [JBS00, Jer84]. This technique is implemented by passing $N$ data symbols through a model of the studied digital system and by counting the number of errors that occur at receiver. The simulation will include pseudo random data and noise sources, along with the models of the devices that process the signal present in the studied system. A number of symbols are processed by the simulation, and the experimental BER is then estimated.

Let us consider a communication system transmitting BPSK symbols over an AWGN channel. Let $(b_i)_{1\leq i\leq N} \in \{-1, +1\}$ be a set of $N$ independent transferred data. For AWGN channel, the standard baseband system model can be expressed as :

$$s = g \cdot b + n, \tag{2.1}$$

where $s$ and $b$ are the received and transmitted signals respectively, $g$ is the channel gain, $n$ is the additive noise.

Let $(X_i)_{1\leq i\leq N}$ be the corresponding soft output before the decision at the receiver. Thus, $X_i = s_i, i = 1, \ldots, N$.

The hard decision is given by :

$$\hat{b}_i = sign(X_i) \tag{2.2}$$

We introduce the following Bernoulli decision function :

$$I(b_i) = \begin{cases} 1 & \text{if } \hat{b}_i \neq b_i, \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

Hence, the BER can be expressed as :

$$p_e = P(\hat{b}_i \neq b_i) = P[I(b_i) = 1] = E[I(b_i)] \tag{2.4}$$

where $E[\cdot]$ is the expectation operator. Using multiple realizations of the transmitter and channel, the MC method estimates the BER by using the ensemble average.

$$\hat{p}_e = \frac{1}{N} \sum_{i=1}^{N} I(b_i) \tag{2.5}$$

The estimation error is given by :

$$\Delta = p_e - \hat{p}_e = \frac{1}{N} \sum_{i=1}^{N} (p_e - I(b_i)) \tag{2.6}$$

The variance of the estimation error is given by [JBS00] :

$$\sigma_\Delta^2 = \frac{p_e(1-p_e)}{N} \tag{2.7}$$

Then the normalized estimation error can be expressed as :

$$\sigma_n = \frac{\sigma_\Delta}{p_e} = \sqrt{\frac{1-p_e}{p_e N}} \tag{2.8}$$

Specifically, for small BER, Eq. 2.8 can be rewritten as :

$$\sigma_n \approx \sqrt{\frac{1}{p_e N}} \tag{2.9}$$

Eq. 2.9 gives the number of transmitted data symbols needed for a desired accuracy :

$$N \approx \frac{1}{\sigma_n^2 p_e} \tag{2.10}$$

Clearly, small values of BER requires a large number of data symbols, otherwise the number of errors is too small and the estimation deviation shall be large. For example, $N \approx 100/p_e$ is needed while counting 100 errors. If we wish to study a system with a BER equal to $10^{-6}$, we need at least $10^8$ bits.

In a word, MC simulation takes excessively long time to compute small BER values. Various variance reduction solutions can be used to decrease the estimation deviation without increasing the number of data symbols [JBS00, SBG+66, Jer84]. However, the implementations of these methods are complex.

**Remark** : in the above analysis, we have assumed that the bit errors are independent, otherwise the number of errors will increase [JBS00].

### 2.1.2 Importance Sampling method

As previously discussed, small BER requires a large number of data symbols. This is often considered as a fatal weakness of the classical Monte-Carlo method, especially for Spread Spectrum (SS) communication systems [QGP99] (e.g., CDMA system) that every transmitted bit needs to be modulated by the spread spectrum codes with a large number of bits.

A widely used method that can reduce BER simulation complexity for SS communication systems is a modified Monte-Carlo method, called Importance Sampling (IS) method [Wik13a, And99]. In [CHD09], a BER estimation method based on Importance Sampling applied to Trapping Sets has been proposed.

For Importance Sampling method, the statistics of the noise sources in the system are biased in some manner so that bit errors occur with greater probability, thereby reducing the required execution time. As an example, for a BER equal to $10^{-5}$, we may artificially "degrade" the channel performance to increase the BER to $10^{-2}$.

Fig. 2.1 shows the general system structure using Importance Sampling method to estimate BER.

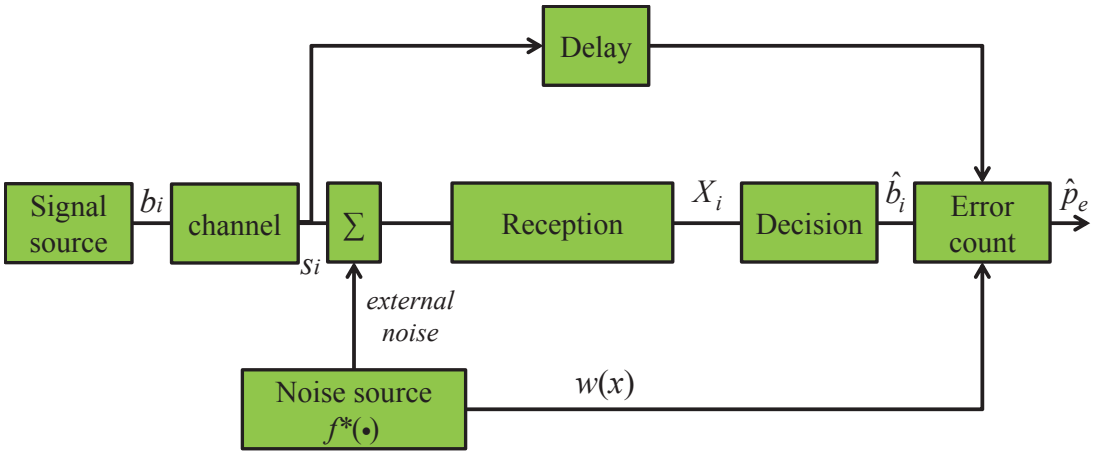


***Figure 2.1*** — Structure of a system using Importance Sampling method

Let $X_i; i = 1, 2, \ldots, N$ be the input of the decision device. Let $f(\cdot)$ be the original noise probability density function and let $f^{\star}(\cdot)$ be the increased noise probability density function using external noise source. We define the weighting coefficient :

$$w(x) = \frac{f(x)}{f^{\star}(x)}$$

For a simple threshold-sensing decision element, an error occurs when there is a large excursion of the threshold voltage $V_T$, i.e.,

$$b_i = 0 : \begin{cases} \text{error count} = 1 & \text{if } X_i \geq V_T, \\ \text{error count} = 0 & \text{otherwise.} \end{cases}$$

$$b_i = 1 : \begin{cases} \text{error count} = 1 & \text{if } X_i \leq V_T, \\ \text{error count} = 0 & \text{otherwise.} \end{cases}$$

When "zero" is sent ($b_i = 0$), the error probability is simply :

$$p_e = \int_{-\infty}^{+\infty} I(x) f(x) dx$$

where $I(\cdot)$ is an indicator function that is one when an error occurs and zero when the correct symbol is obtained, i.e.,

$$I(X_i) = \begin{cases} 1 & \text{if } X_i \geq V_T \text{ when } b_i = 0 \text{ or } X_i \leq V_T \text{ when } b_i = 1, \\ 0 & \text{otherwise.} \end{cases}$$

Considering the natural estimator of the expectation which is the sample mean, we can write :

$$\hat{p}_e = \frac{1}{N} \sum_{i=1}^{N} I(X_i) \tag{2.11}$$

Considering the probability density function of the external noise, we can rewrite :

$$\begin{aligned} p_e &= \int_{-\infty}^{+\infty} I(x) \frac{f(x)}{f^\star(x)} f^\star(x) dx \\ &= \int_{-\infty}^{+\infty} I^\star(x) f^\star(x) dx \\ &= E[I^\star(X)] \end{aligned}$$

The above equation is not merely a mathematical artifice. In fact, the statistics of the noise processes are altered and the expectation is performed with respect to $f^\star(\cdot)$.

As before, we can obtain the estimator by using the sample mean.

$$\hat{p}_e^\star = \frac{1}{N} \sum_{i=1}^{N} I^\star(X_i) = \frac{1}{N} \sum_{i=1}^{N} w(X_i) I(X_i) \tag{2.12}$$

Comparing with Eq. 2.11, the weight, $w(x)$, is added and evaluated at $X_i$. This means that it is possible to reduce the variance by introducing an external noise with biased density.

The performance of IS-based BER estimation strongly depends on biasing scheme $w(x)$. If a good biasing scheme is selected for a given system, an accurate BER estimate can be obtained with very short simulation run time. Otherwise, the BER estimate may even converge at a slower rate than the conventional Monte-Carlo simulation. This means that the IS method cannot be considered as a generic method for BER estimation for any given communication system.

### 2.1.3 Tail Extrapolation method

The BER estimation problem is essentially a numerical integration problem. Let us consider the eye diagram in Fig. 2.2, measured for a GSM system when $SNR = 20\ dB$. We can find the worst case of the received bit sequence.

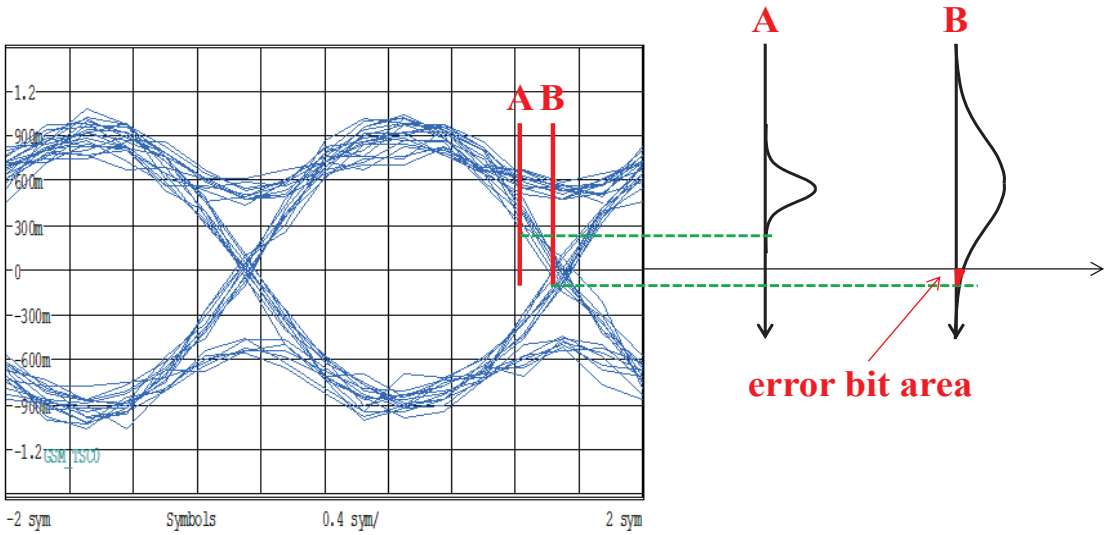


***Figure 2.2*** — Probability density function tails from eye diagram

We consider the probability density function of the eye slice at line A and B. The lower bound on the probability density function (green line) is the worst case bit sequence, and the small red area contains all of the bit errors. The BER of the given system can be considered as the area under the tail of the probability density function.

In general, we cannot describe which kind of distribution the slopes of the bathtub curve of eye diagram belong to. We can assume that the probability density function belongs to a particular class and then perform a curve fit to the observed data. This approach for BER estimation is called Tail Extrapolation (TE) method [JBS00].

We shall set multiple thresholds of lower bound. A normal Monte-Carlo simulation is executed, and the number of times the decision metric exceeds each threshold is recorded. A broad class of the probability density functions is then identified. The tail region is often described by some member of the Generalized Exponential Class (GEC), which is defined as :

$$f_{\upsilon,\sigma,\mu}(x) = \frac{\upsilon}{2\sqrt{2}\Gamma(\frac{1}{\upsilon})} \exp\Big(-\Big|\frac{x-\mu}{\sqrt{2}\sigma}\Big|^{\upsilon}\Big)$$

where

- $\Gamma(\cdot)$ is the gamma function ;
- $\mu$ is the mean of the distribution ;

– $\sigma$ is related to the variance $V_\upsilon$ through :

$$V_\upsilon = \frac{2\sigma^2\Gamma(\frac{3}{\upsilon})}{\Gamma(\frac{1}{\upsilon})}$$

The parameters $(\upsilon, \sigma, \mu)$ are then adjusted to find the probability density function that best fits the data samples. Thereby the BER can be estimated by evaluating the integral of the probability density function for the used threshold.

However, it is not always clear which class of probability density function and which thresholds should be selected. In general, it is difficult to evaluate the accuracy of the BER estimate [JBS00, SBG+66, Jer84].

### 2.1.4 Quasi-analytical estimation

The above methods consist in analyzing the entire received waveform (data + noise) at the output of receiver. Now we consider solving the BER estimation problem in two steps :

– One deals with the transmitted signal ;
– The other deals with the noise contribution to the waveform.

Particularly, we assume that :

– The noise is referred to as an Equivalent Noise Source (ENS) ;
– The probability density function of the ENS is known and specifiable.

Therefore, we can assume that the system performance can be closely evaluated by an ENS having a suitable distribution. This method is called the Quasi-Analytical (QA) estimation [Jer84]. By taking into account the noiseless waveform, we can compute the BER with the ENS statistics. More specifically, we let the simulation itself compute the effect of signal fluctuations in the absence of noise, and then superimpose the noise on the noiseless waveform.

The assumption of the noise statistics leads to a great reduction in computation effort. The usefulness of the QA estimation will depend on how closely the assumption matches reality [SPKK99]. However, except for the linear system, the ENS statistics may be very difficult to predict before the fact.

### 2.1.5 BER estimation based on Log-Likelihood Ratio

Receiver can implement soft-output decoding (e.g. A Posteriori Probability (APP) decoder) to minimize the bit error rate in each information bit. The APP decoder may output probabilities or Log-Likelihood ratio values.

Let $(b_i)_{1\leq i\leq N} \in \{+1, -1\}$ be the transmitted bits and $X_i; i = 1, 2, \ldots, N$ be the received values. The LLR is defined as :

$$LLR_i = LLR(b_i|X_i = x_i) = \log \frac{P(b_i = +1|X_i = x_i)}{P(b_i = -1|X_i = x_i)} \tag{2.13}$$

Using Bayes' theorem, we get :

$$LLR_i = \underbrace{\log \frac{P(b_i = +1)}{P(b_i = -1)}}_{\text{a-priori information}} + \underbrace{\log \frac{P(X_i = x_i|b_i = +1)}{P(X_i = x_i|b_i = -1)}}_{\text{channel information}} \tag{2.14}$$

The hard decision is performed by computing the sign of the LLR, i.e.,

$$\hat{b}_i = \begin{cases} +1 & \text{if } LLR_i(b_i|X_i) > 0, \\ -1 & \text{otherwise.} \end{cases}$$

In [LHS00], some fundamental properties of LLR values are derived and new BER estimators are proposed based on the statistical moments of the LLR distribution.

Consider the following criterion :

$$P(X = +1|Y = y) + P(X = -1|Y = y) = 1$$

Solving Eq. 2.13 by using the above criterion allows to derive the a posteriori probabilities $P(b_i = +1|X_i)$ and $P(b_i = -1|X_i)$.

$$P(b_i = +1|X_i) = \frac{e^{LLR_i}}{1 + e^{LLR_i}} \qquad P(b_i = -1|X_i) = \frac{1}{1 + e^{LLR_i}}$$

We take the absolute value $\Lambda = |LLR_i|$ of the LLR. Then, we can derive the probability that the hard decision of the $i^{th}$ information bit is wrong.

$$p_i = \frac{1}{1 + e^{-\Lambda}} \tag{2.15}$$

Hence, an estimate of BER can be given by :

$$\hat{p}_{e,1} = \frac{1}{N} \sum_{i=1}^{N} p_i \tag{2.16}$$

Another estimate of BER can be derived by using the exponential symmetric property of the LLR [LHS00].

$$\hat{p}_{e,2} = \int_{\lambda} \hat{f}_{\Lambda}(\lambda) \frac{1}{1 + e^{\lambda}} d\lambda \tag{2.17}$$

where $f_{\Lambda}$ is the estimated PDF of $\Lambda$.

The limitations of the LLR-based BER estimation method are :
- the $1^{st}$ estimate of BER given by Eq. 2.16 might be less efficient than the $2^{nd}$ estimate of BER given by Eq. 2.17 since $f_\Lambda(\lambda)$ is typically Gaussian and smooth enough ;
- the $2^{nd}$ estimator is more complex to be performed since an estimate of $f_\Lambda(\lambda)$ must be computed (e.g., by means of a histogram) before the integral ;
- both methods are sensible to the variance of channel noise since the LLR distribution strongly depends on the accuracy of the SNR estimate. In fact, the above estimators implicitly assume that the SNR is known to the decoder.

In [SSB05], a new BER estimator which does not exhibit a dependence on the SNR uncertainty has been proposed. However, the estimator relies on the erroneous Gaussian LLR distribution assumption.

### 2.1.6 Conclusion of BER estimation methods

There is no shortage of techniques that can be applied to the Bit Error Rate estimation. In this section, we have presented the conventional Monte-Carlo simulation. In consideration of the very long execution time for low BER, we have discussed three techniques, Importance Sampling method, Tail Extrapolation method and Quasi-Analytical estimation.

These solutions require the assumptions concerning the behavior of real system, and the performance is strongly determined by the assumed parameters, which probably need to be modified for different communication system. For general case, it is difficult to find the ideal model or the suitable values of the parameters.

Then, some new BER estimators based on the LLR distribution have been presented but still with some drawbacks, e.g. BER estimators present dependence on the SNR uncertainty and on the particular channel characteristics. In addition, other recent papers on this topic can be found in [DWY09, LRT03].

Unfortunately, all these methods require the knowledge of the transmitted information bits, whereas in practical situation the estimator does not know transmitted data.

## 2.2 Probability Density Function estimation

To speed up Monte-Carlo simulation, the QA estimation method proposed using the Probability Density Function of ENS. As previously discussed, the required ENS statistics is difficult to be found in practical situation. Anyway, this has drawn great inspiration for BER estimation : instead of using the PDF of noise, estimating the PDF of soft observations may be more helpful since :

– the soft observations are perfectly known by receiver ;

– the PDF of receiver's soft outputs includes all the information about the system, such as the transceiver scheme, the distortion, the channel model, etc. In other words, the PDF itself provides a description of end-to-end performance of system. Thus, the PDF estimate should have a very close relationship with BER estimate.

Hence, PDF estimation can be an interesting alternative to BER estimation. Such BER estimation method using the channel/receiver's soft outputs is called soft BER estimation since it directly use the soft observations without requiring hard decisions about transmitted bits. The principle is quite simple :

– firstly we estimate the Probability Density Function of received values ;

– then we compute the BER estimate by using the PDF estimate.

In this section, we will discuss the widely used methods that can be applied to the probability density function estimation and present how to use the PDF estimate to compute BER estimate.

### 2.2.1 Introduction to PDF estimation

Probability Density Function estimation deals with the problem of modeling a probability density function given a finite number of data set. In the last 25 years, density estimation has experienced a tremendous development. The techniques of density estimation are very useful in the context of parametric statistics. They have been applied in many fields, such as banking (e.g., Tortosa-Ausina, 2003 [TA03]), economics (e.g., DiNardo and Fortin, 1996 [DFL95]), archaeology (e.g., Baxter, Beardah and Westwood, 2000 [BBW00]), etc.

There are three basic approaches to perform density estimation :

– Parametric density estimation :
normally, parametric density estimation is referred to parameter estimation. This approach consists in assuming a given functional form for the density function (e.g., Gaussian). The parameters of the distribution function (e.g., mean and variance for Gaussian distribution) are then optimized to fit the dataset ;

– Non-parametric density estimation :
compared to parametric density estimation, no functional form of the density function is assumed. The density estimation is determined entirely from the dataset. As an example, a histogram calculation is typically performed by non-parametric density estimation ;

– Semi-parametric density estimation :
semi-parametric approach consists in using mixture models that have parametric and non-parametric components.

**Remark** : the mentioned "dataset" is a set of observation samples from experiment, typically the values $X_i; i = 1, 2, \ldots, N$, which consists of repeated I.I.D. (Independent and Identically Distributed) sampling from a probability distribution.

### 2.2.2 Parametric density estimation : Maximum Likelihood Estimation

Assume that the probability density function takes a particular parametric form ; the parametric density estimation consists in estimate the values of the parameters of the density function.

Two main decisions have to be made for implementing a parametric approach :

- specify the parametric form of the density function :
  the parametric form of the density estimator determines what family of density functions can be expressed using that functional form. Thus, a number of chosen assumptions are the inductive bias of the density estimator. With a large number of data samples, estimation result is good if the chosen model fits the training data.
- learn the parametric model based on observation dataset :
  three main approaches can be implemented to solve this problem :
  a) Maximum Likelihood Estimation (MLE) : this approach consists in choosing parameter values that maximize the probability of the dataset ;
  b) Bayesian inference [DY79] : this approach consists in maintaining a probability distribution over all possible parameter values, balancing a prior distribution with the evidence of the dataset ;
  c) Maximum A-Posteriori (MAP) [Wol80, Gre90, DHS12] : this approach can be considered as a regularization of MLE and usually used to obtain a point estimate of an unobserved/unsupervised dataset on the basis of empirical data. The MAP is widely used for decoding algorithm (e.g., Turbo decoding).

In this report, we focus on the most widely used approaches : Maximum Likelihood estimation.

For a fixed dataset and underlying distribution, Maximum Likelihood estimation selects values of the distribution parameters that give the observed data the greatest probability.

Considering an I.I.D. dataset including $N$ random variables, $X_i; i = 1, 2, \ldots, N$, drawn from the distribution $f_X(x|\theta)$ where $\theta$ is a number of unknown parameters $[\theta_1, \ldots, \theta_m]$, then we can write :

$$f_X(x|\theta) = \prod_{i=1}^{N} f_X(X_i|\theta) = L(\theta) \tag{2.18}$$

where $L(\theta)$ is the likelihood of the dataset. To give the greatest probability, simply maximize $L(\theta)$ for parameters $\theta$.

$$\hat{\theta} = \arg\max_{\theta}[f_X(x|\theta)] \tag{2.19}$$

In practice, rather than using the likelihood, we use :

– logarithm of $L(\theta)$, called the log-likelihood, because it is generally easier to work with sums rather than products ;

$$\log[L(\theta)] = \sum_{i=1}^{N} \log[f_X(X_i|\theta)] \tag{2.20}$$

– average log-likelihood :

$$\frac{1}{N} \log[L(\theta)] \tag{2.21}$$

Also, it is often more convenient to minimize the negative log-likelihood :

$$\hat{\theta} = \arg\min_{\theta}[-\log L(\theta)] \tag{2.22}$$

A general approach to maximize the log-likelihood or to minimize the negative log-likelihood consists in taking the analytic derivative of the error function and equating to zero. For Gaussian estimators, we obtain the following equation for the estimates $\hat{\mu}$ and $\hat{\sigma}^2$ :

$$\begin{aligned} \hat{\mu} &= \frac{1}{N} \sum_{i=1}^{N} X_i \\ \hat{\sigma}^2 &= \frac{1}{N} \sum_{i=1}^{N} (X_i - \hat{\mu})^2 \end{aligned} \tag{2.23}$$

Thus, the ML estimates for normal Gaussian distribution $N(\mu, \sigma^2)$ is given by :

$$\hat{\theta} = (\hat{\mu}, \hat{\sigma}^2) \tag{2.24}$$

In statistics, the bias of an estimator is defined as the difference between the estimator's expected value and the true value of the estimated parameter. An estimator is said to be unbiased when :

$$Bias[\hat{\theta}] = E[\hat{\theta}] - \theta = 0 \tag{2.25}$$

It turns out that the ML estimate of the mean for normal Gaussian distribution is unbiased. However, the ML estimate of the variance is biased since :

$$E[\hat{\sigma}^2] = \frac{N-1}{N} \sigma^2 \neq \sigma^2 \tag{2.26}$$

The parameter that we really want to measure is the true variance which represents the average distance of the samples from the center $\mu$ of the actual distribution, but we use the center of the samples $\hat{\mu}$ instead. Basically, the sample mean is an Ordinary Least Squares (OLS) [Wik13b] estimator for the true center $\mu$, and $\hat{\mu}$ makes the sum $\sum_{i=1}^{N}(X_i - \hat{\mu})^2$ as small as possible. In general, the average distance of the samples from the center of the samples is less than the average distance from the center of the distribution.

Because of this bias, the definition of sample variance is usually based on the Laplace approximation which multiplies the estimate of the variance by $\frac{N}{N-1}$. Notice that MATLAB uses this unbiased estimate of the variance.

$$\hat{\sigma}^2 = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - \hat{\mu})^2 \tag{2.27}$$

For a large number of samples, the bias of the variance for normal distribution becomes zero asymptotically. The bias is only noticeable when there are very few samples.

In general, we define two parameters to measure the performance of statistical estimates [Que56] :

- bias : it measures how close is the estimate to the true value of the parameter ;
- variance (or standard deviation) : it measures how much it changes for different datasets.

**Remark** : the normal distribution can be expressed in terms of the Q-function. $Q(x)$ is the Complementary Cumulative Distribution Function (CCDF) for a zero mean and a unit variance Gaussian distribution.

$$Q(x) = 1 - \Phi(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} \exp(-\frac{\mu^2}{2})d\mu \tag{2.28}$$

where $\Phi(x)$ is the Cumulative Distribution Function (CDF) of the normal Gaussian distribution.

### 2.2.3 Non-parametric density estimation

So far we have been discussing the parametric estimation. Either the likelihood or at least the parametric form was known.

Instead, the non-parametric approach avoids any assumptions about the density distribution of samples, which means that the non-parametric density estimation attempts to estimate the density function directly from the dataset without assuming a particular form for the underlying distribution.

#### 2.2.3.1 Empirical density estimation

The Empirical Probability Density Function is considered as the simplest density estimate [WW78]. For a dataset $X_i, i = 1, \ldots, N$, the empirical estimator is given by :

$$\hat{f}_X(x) = \frac{1}{N}\sum_{i=1}^{N}\delta(x - X_i) \tag{2.29}$$

The Empirical PDF estimation consists in placing a delta function at each data point. By introducing the bins, this method can evolve into a well-known non-parametric estimator : the histogram.

#### 2.2.3.2 Histogram

A histogram is constructed by a starting point $x_0$, and a bin width $h$. The bins which enclose the data point $X_i$ are of the form $I_m = [x_0 + (m-1)h, x_0 + mh), m \in \{1, 2, \ldots, M\}$. The estimator is given by :

$$\hat{f}_X(x) = \frac{1}{Nh} \sum_{i=1}^{N} \sum_{m=1}^{M} B_{I_m}(X_i) \tag{2.30}$$

where :

$$B_{I_m}(X_i) = \begin{cases} 1 & \text{if } X_i \in I_m, \\ 0 & \text{otherwise.} \end{cases} \tag{2.31}$$

The choice of the bin width (or the number of bins) has a substantial effect on the shape and other properties of estimator [VRV94].

Fig. 2.3 shows two histograms and the true PDF $f_X(x)$ for the data set, with different bin widths (so different bins number). The red curve represents the theoretical density function. Note that the estimates are piecewise constant and that they are strongly influenced by the choice of bin width.

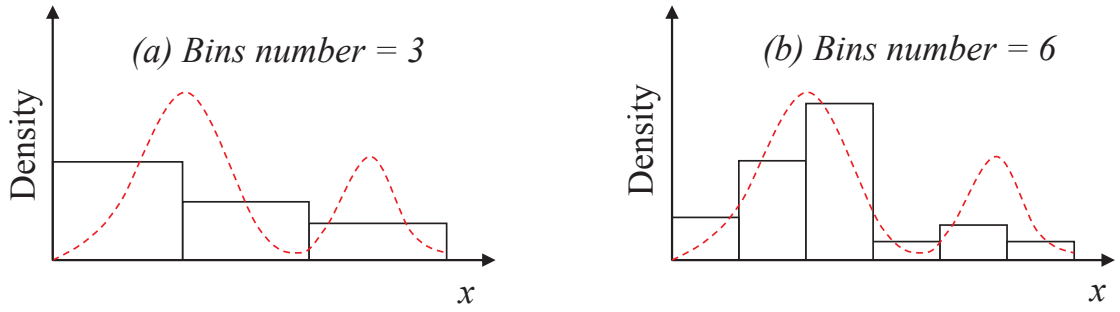


***Figure 2.3*** — Histograms with different bin widths

We can make a summary of the drawbacks of the histogram :

- strict dependence on bin width : in high dimensions we would require a very large number of samples, otherwise most of the bins would be empty ;
- we obtain a step function even if the theoretical PDF is a smooth one.

These issues make the histogram unsuitable for most practical applications.

#### 2.2.3.3 General formulation of non-parametric density estimation

We consider an I.I.D. random variable, $X_i, i = 1, \ldots, N$, with a distribution $f_X(x)$. The probability that $X$ will fall in a given region, $\Phi$, is given by :

$$P = \int_{\Phi} f_X(t)dt \tag{2.32}$$

Suppose that $N$ samples are drawn from the distribution $f_X(x)$, the probability that $k$ samples ($k \leq N$) are enclosed in the region $\Phi$ is given by the binomial distribution :

$$P(n = k) = \begin{pmatrix} N \\ k \end{pmatrix} P^k (1 - P)^{N-k} \tag{2.33}$$

The mean and the variance of the ratio $\hat{n}/N$ can be expressed as :

$$E[\frac{\hat{n}}{N}] = P \qquad Var[\frac{\hat{n}}{N}] = E[(\frac{\hat{n}}{N} - P)^2] = \frac{P(1-P)}{N} \tag{2.34}$$

It can be shown from Eq. 2.34 that the distribution becomes sharper when $N \to \infty$. Therefore, we can expect that a good estimate of the probability can be obtained from the mean fraction of the points enclosed in the region $\Phi$.

$$P \approx \frac{\hat{n}}{N} \tag{2.35}$$

Assume that the volume enclosed by the region $\Phi$ is small enough, we can write :

$$P = \int_{\Phi} f_X(t)dt \approx f_X(x) \times \text{Volume}_{\Phi} \tag{2.36}$$

From Eq. 2.35 and Eq. 2.36, we obtain :

$$f_X(x) \approx \frac{\hat{n}}{N \times \text{Volume}_{\Phi}} \tag{2.37}$$

From Eq. 2.37, it can be shown that the estimate becomes more accurate as $N$ is large enough and the volume of $\Phi$, $\text{Volume}_{\Phi}$, is small enough. This means that we have to find a compromise for the volume of $\Phi$ :

- the volume of $\Phi$ must be large enough to include enough samples within $\Phi$ ;
- the volume of $\Phi$ also must be small enough to support the assumption given by Eq. 2.36.

Moreover, two basic approaches can be adopted while applying Eq. 2.37 to practical density estimation problems :

- we can "fix" $k$ and determine $\text{Volume}_{\Phi}$ from the data. This leads to k-Nearest Neighbor (kNN) approach [TS92, Das91] ;
- we can also "fix" $\text{Volume}_{\Phi}$ and determine $k$ from the data. This leads to Kernel Density Estimation (KDE) approach [WJ95, Sil86].

#### 2.2.3.4 Introduction to Kernel Density Estimation

The KDE consists in data smoothing compared to the histogram. In some fields such as signal processing it is also termed the Parzen Window method or the naïve estimation, after Murray Rosenblatt and Emanuel Parzen [Ros56, Par62]. Some software packages for Kernel estimation can be found, such as the KEYS (a.k.a., Kernel Estimating Your Shapes) package [Cra01].

##### 2.2.3.4.1 Naïve estimator : Parzen window

Assume that the region $\Phi$ that encloses the $k$ samples is a hypercube with sides of length $h$ centered at $x$, and then its volume is given by :

$$\text{Volume}_{\Phi} = h^D \tag{2.38}$$

where $D$ is the number of dimensions.

The normalized Parzen window is defined as :

$$K_{\text{Parzen}}(x) = \begin{cases} 1 & \text{if } |x_d| < 0.5, \forall d = 1, \ldots, D, \\ 0 & \text{otherwise.} \end{cases} \tag{2.39}$$

In general, this Kernel function, which corresponds to a unit hypercube centered at the origin, is known as a Parzen window.

Assume that $X_i; i = 1, \ldots, N$ are drawn independently from the true density $f_X(x)$, to verify if $x$ is inside the hypercube centered at $X_i$, we can write :

$$K_{\text{Parzen}}(\frac{x - X_i}{h}) = \begin{cases} 1 & \text{if the hypercube of } X_i \text{ enclose } x, \\ 0 & \text{otherwise.} \end{cases} \tag{2.40}$$

The total number of samples inside the hypercube is then :

$$\hat{n} = \sum_{i=1}^{N} K_{\text{Parzen}}(\frac{x - X_i}{h}) \tag{2.41}$$

Merging with Eq. 2.37 we obtain the density estimate as :

$$f_{\text{Parzen}}(x) = \frac{1}{Nh^D} \sum_{i=1}^{N} K_{\text{Parzen}}(\frac{x - X_i}{h}) \tag{2.42}$$

#### 2.2.3.4.2 Smooth Kernels

Basically, the Parzen window has the following drawbacks :

– density estimates have discontinuities ;

– it weights equally all points $(X_i)_{i=1,...,N}$, regardless of their distance to the estimation point.

For these reasons, the Parzen window is commonly replaced with a smooth Kernel function $K(x)$ normalized to 1 :

$$\int_{\Phi} K(x)dx = 1 \tag{2.43}$$

When $D = 1$, this leads to the density estimate as :

$$\hat{f}_X(x) = f_{KDE}(x) = \frac{1}{Nh}\sum_{i=1}^{N} K(\frac{x - X_i}{h}) \tag{2.44}$$

Usually, $K(x)$ is a symmetric and uni-modal PDF :
– center of Kernel is placed right over each data point ;
– contribution from each point is summed to overall estimate.

Therefore, any function having the following properties can be used as a Kernel :

$$\begin{cases} (a) & \int K(x)dx = 1 \\ (b) & \int xK(x)dx = 0 \\ (b) & \int x^2K(x)dx < \infty \end{cases} \Rightarrow \; Kernel \tag{2.45}$$

In our works, it is assumed to be an even and regular function with zero mean and unit variance, such as the Gaussian Kernel :

$$K_D(x) = \frac{1}{(2\pi)^{\frac{D}{2}}} e^{-\frac{1}{2}x^T x} \quad K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} \; when \; D = 1 \tag{2.46}$$

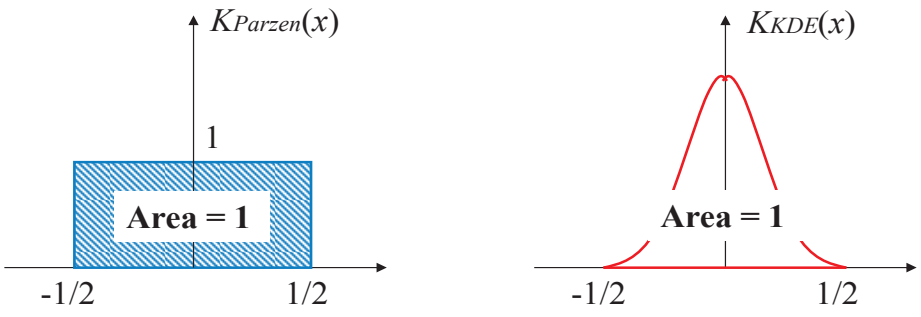


***Figure 2.4*** — Parzen window vs. Gaussian Kernel

Next, we compute the expectation of the PDF estimate $f_{KDE}(x)$.

$$\begin{aligned} E[f_{KDE}(x)] &= \frac{1}{Nh}\sum_{i=1}^{N} E[K(\frac{x-X_i}{h})] \\ &= \frac{1}{h}\int K(\frac{x-t}{h}) f_X(t)dt = \frac{1}{h} f_X(x) * K(\frac{x}{h}) \end{aligned} \tag{2.47}$$

It can be shown that the expectation of the estimate $f_{KDE}(x)$ is a convolution of the true density $f_X(x)$ with the Kernel function. Thus, the Kernel width $h$ plays the role of a smoothing parameter : the smoother or the bandwidth of the estimate $f_{KDE}(x)$.

A Kernel is also a standardized weighting function, namely the weighting function with $h = 1$.

$$\text{weight}(x, h) = \frac{1}{h}K(\frac{x}{h}) \tag{2.48}$$

The Kernel function determines the shape of the weighting function, and the smoothing parameter $h$ determines the amount of smoothing. The two components determine the properties of the estimate $f_{KDE}(x)$.

In an ideal condition, when $h \to 0$, the weighting function $\frac{1}{h}K(\frac{x}{h})$ approaches a delta function and the PDF estimate $f_{KDE}(x)$ approaches the true density $f_X(x)$. However, in practice we only have a finite number of samples, so the bandwidth $h$ cannot be made arbitrarily small, or the PDF estimate $f_{KDE}(x)$ would degenerate to a set of impulses.

The properties of Kernel estimators, the bandwidth selection and the performance of Kernel-based PDF estimation will be detailed in Chapter 3.

### 2.2.4 Semi-parametric density estimation

So far we have been discussing the parametric estimation and the non-parametric density estimation techniques. The non-parametric density estimation is widely used in practice since there is no assumption about the form of the unknown density function, and the estimation is entirely based on the soft data points. However, the number of parameters grows with the size of the dataset, so the models can quickly become unwieldy and long computing time is required.

One solution to overcome this consists in using a hybrid approach : semi-parametric density estimation [Far90], which consists in assigning functions for a set of data points rather than fitting one Kernel function for each data point.

Using the semi-parametric density estimation, we can control the number of components, and then pick a compromise between the efficiency of parametric methods and the flexibility of non-parametric methods.

#### 2.2.4.1 Introduction to Gaussian Mixture Model

Mixture model is widely used as semi-parametric model for PDF estimation. This method output a weighted sum of their parametric mixture components. The parameters comprise :

- mixture coefficients ;
- and all the parameters of the individual components.

Of all mixture models, the Gaussian Mixture Model is the most widely used for data clustering and pattern recognition in signal processing and signal analysis domain [MP04, RW84, CS09, WL05]. In this case, the studied distribution can be expressed as a weighted sum of several Gaussian distributions with different means and different variances.

Assume that $X$ is the dataset with $N$ samples and $K$ is the number of Gaussian components. The unknown PDF is a mixture of $K$ Gaussians as follows [TSM+85] :

$$f_{GM}(x) = f_{X,N}(x) = \sum_{k=1}^{K} \alpha_k f_k(x; \mu_k, \sigma_k^2) \tag{2.49}$$

where :

- $\alpha_k$ is the population fraction in $k$, in other words, it represents the a priori probability of the $k^{th}$ component for the Gaussian mixture ;

$$\sum_{k=1}^{K} \alpha_k = 1 \tag{2.50}$$

- $f_k(x; \mu_k, \sigma_k^2)$ is a Gaussian PDF with mean $\mu_k$ and variance $\sigma_k^2$.

This can be seen as :

$$f_{X,N}(x) = \sum_{k=1}^{K} [P(k^{th}\ component) \times f(x|k^{th}\ component)] \tag{2.51}$$

In general, the unknown parameters of the GM are represented by :

$$\theta = (\alpha_k, \mu_k, \sigma_k^2)_{1 \leq k \leq K}$$

#### 2.2.4.2 Difficulties of Mixture Models

The Gaussian Mixture Model-based PDF estimation technique consists in finding the above unknown parameters with a given number of data samples for K components. However, these parameters cannot be directly determined from the N data samples.

In fact, the difficulty in learning a mixture model is to know which mixture component should be associated to which data. Imagine the data points are clustered into several groups :

– to assign data points to their clusters, we need to have each mixture component fitted to its cluster ;
– to fit each component to its cluster, we need to know which data point(s) belong to which cluster.

In data clustering, the data samples $X_i$ are incomplete data because of missing the cluster assignment information. We shall introduce another parameter, $Z_i$, which is the missing data and should be combined with the data points $X_i$ to build the complete dataset $\{X_i, Z_i\}_{i=1,\ldots,N}$. The value of the missing data is set to $k$ if the data $X_i$ is generated by the $k^{th}$ component of the mixture model.

In Chapter 4, we will present how to determine the unknown parameter $\theta$ by using the incomplete data $X_i$ along with the missing data $Z_i$.

## 2.3 BER calculation with PDF estimate

Once we obtain the PDF estimate of soft observations, the BER estimate can be calculated in an analytical fashion.

Let us consider a simple example. Let $(b_i)_{1\leq i\leq N} \in \{-1, +1\}$ a set of $N$ I.I.D. transmitted data, let $(X_i)_{1\leq i\leq N}$ be the corresponding output of the sign decision having the same probability density function, $f_X(x)$. According to Eq. 2.4, the BER is given by :

$$\begin{aligned} p_e &= P[\hat{b}_i \neq b_i] \\ &= P[X < 0, b_i = +1] + P[X > 0, b_i = -1] \\ &= P[X < 0 | b_i = +1]P[b_i = +1] + P[X > 0 | b_i = -1]P[b_i = -1] \end{aligned}$$

If the $(b_i)_{1\leq i\leq N}$ are assumed to be identically distributed with $P[b_i = \pm 1] = 1/2$, we have :

$$p_e = \frac{1}{2}P[X < 0 | b_i = +1] + \frac{1}{2}P[X > 0 | b_i = -1] \tag{2.52}$$

Let $f_X^{b_+}(\cdot)$ (resp., $f_X^{b_-}(\cdot)$) be the conditional PDF of $X$ such as $b_i = +1$ (resp., $b_i = -1$), Eq. 2.52 can be rewritten as :

$$p_e = \frac{1}{2}\int_{-\infty}^{0} f_X^{b_+}(x)dx + \frac{1}{2}\int_{0}^{+\infty} f_X^{b_-}(x)dx \tag{2.53}$$

As shown in Fig. 2.5, the BER can be expressed as the area dimension of the intersection PDF curves.

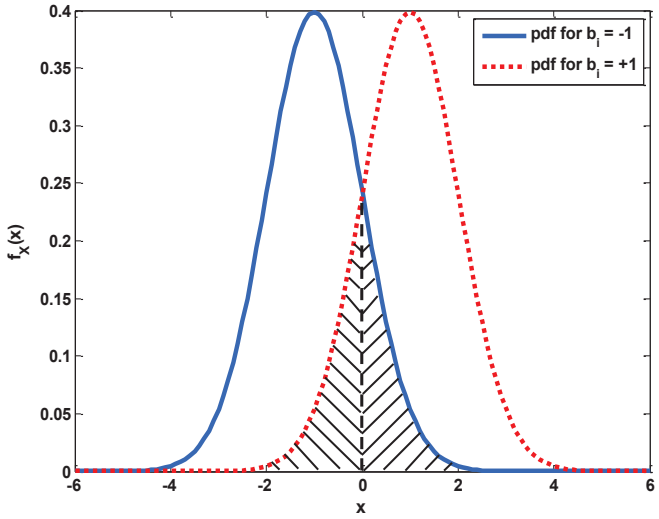


***Figure 2.5*** — BER estimation base on conditional PDF of data samples

Generally, the $(b_i)_{1\leq i\leq N}$ are assumed to be identically distributed with $P[b_i = +1] = \pi_+$ and $P[b_i = -1] = \pi_-$, where $\pi_+ + \pi_- = 1$. Since the data set is independent, the PDF of $X$ is a mixture of the two conditional PDF :

$$f_X(x) = \pi_+ f_X^{b_+}(x) + \pi_- f_X^{b_-}(x) \tag{2.54}$$

Thus, the BER can be rewritten as :

$$p_e = \pi_+ \int_{-\infty}^{0} f_X^{b_+}(x)dx + \pi_- \int_{0}^{+\infty} f_X^{b_-}(x)dx \tag{2.55}$$

If we know the PDF functions $f_X^{b_+}(x)$ and $f_X^{b_-}(x)$, the BER can be easily computed analytically by using parametric PDF estimation method, which means that we only need to estimate the unknown parameters of the PDFs $f_X^{b_+}(x)$ and $f_X^{b_-}(x)$.

### 2.3.1 Theoretical BER : BER estimation based on parametric PDF estimation

Firstly, we shall study the case of parametric PDF estimation. Let us consider the simplest situation : the channel model is AWGN and a sequence of $N$ bits BPSK ($+1$ or $-1$), $(b_i)_{1\leq i\leq N}$, is transmitted.

The AWGN channel model is given by :

$$AWGN \sim N(0, \sigma^2) \tag{2.56}$$

where $\sigma$ is the standard deviation which depends on the applied SNR.

Let $A$ be the amplitude of the transmitted signal, we have :

$$\sigma = \sqrt{\frac{A^2}{2 \times 10^{\frac{SNR}{10}}}} \tag{2.57}$$

where SNR is the signal-to-noise ratio in $dB$.

If the transmitted bits, $(b_i)_{1\leq i\leq N}$, are equal to $-1$, the distribution function of the received signals is :

$$X \sim N(-1, \sigma^2) = N(-1, \frac{A^2}{2 \times 10^{\frac{SNR}{10}}}) \tag{2.58}$$

Obviously, in order to obtain the parametric PDF estimate, we need to know the transmitted data and the SNR estimates.

Assume that we know the channel information and the transmitted bits, the parametric PDF estimation method can be used to compute the theoretical BER values, which can be expressed as :

$$BER_{\text{theoretical}} = \frac{1}{2}\text{erfc}(\frac{1}{\sqrt{2}\sigma}) \tag{2.59}$$

where $\text{erfc}(\cdot)$ denotes the complementary error function defined as [And85, Gre03] :

$$\begin{aligned} \text{erf}(x) &= \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \\ \text{erfc}(x) &= 1 - \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_x^{+\infty} e^{-t^2} dt \end{aligned} \tag{2.60}$$

Fig. 2.6 shows the theoretical BER and the BER estimate obtained by using Monte-Carlo simulation for BPSK system. It can be shown that the parametric PDF estimator provides reliable theoretical BER values with knowledge about the form of soft output $X_i$, the channel model and the transmitted data.

Moreover, we observed the weakness of the conventional Monte-Carlo technique. The MC-based BER estimates match closely to the theoretical BERs for small values of SNR (from 0 $dB$ to 4 $dB$). However, for high SNR ($> 4$ $dB$), the MC simulation cannot return accurate result since 1000 data samples are not sufficient to have a good precision. As an example, we have to count at least 10 errors for each value of SNR, for $SNR = 7$ $dB$ the theoretical value of BER is $8.0 \cdot 10^{-4}$, therefore, the Monte-Carlo simulation needs at least $1.25 \cdot 10^5$ samples for similar precision.

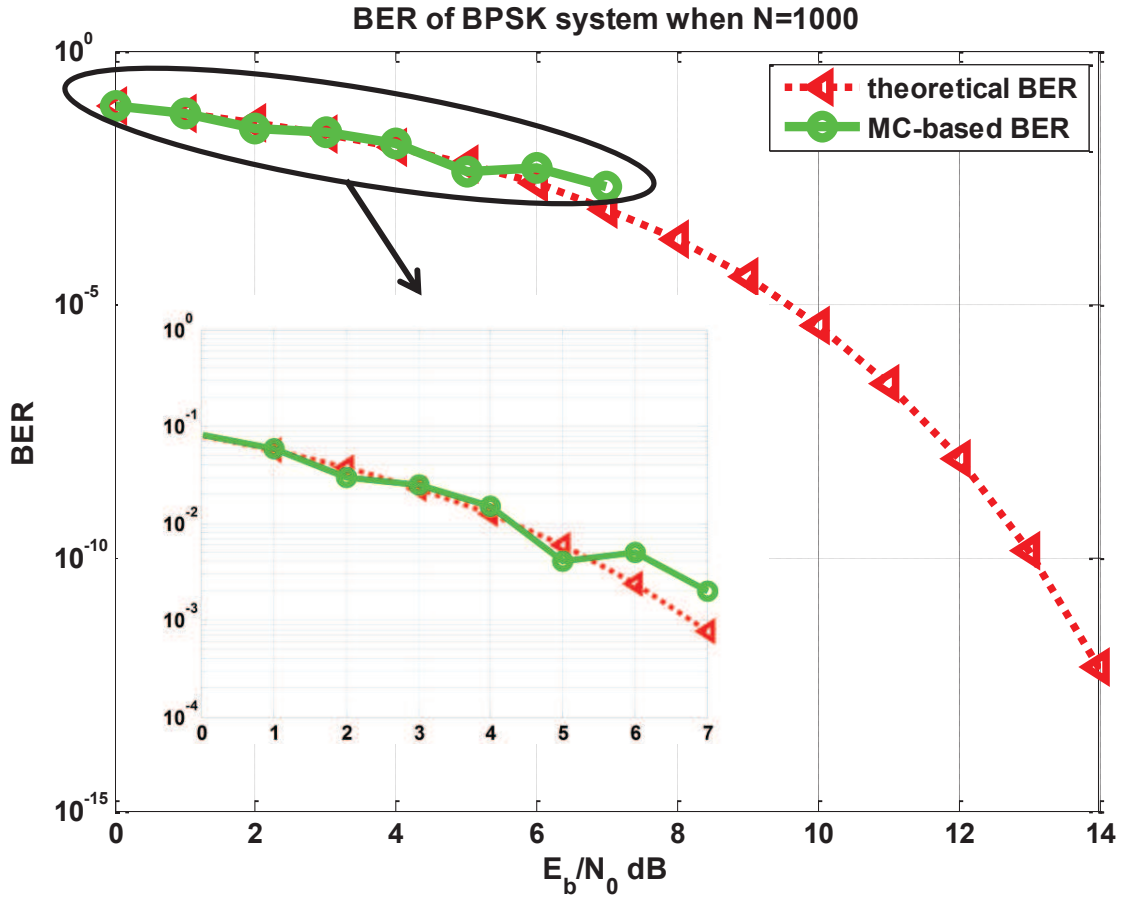


***Figure 2.6*** — Theoretical BER and Monte-Carlo based BER estimate for BPSK system when $N = 1000$

### 2.3.2 Practical situation : necessity of non-parametric or semi-parametric PDF estimation

Using parametric PDF estimation method requires the knowledge about form of the unknown PDF, whereas in practice it could be very difficult to find the right parametric model for the received signal distribution.

In general, the PDF of the output $X_i$ depends on the type of the channel and the system scheme :

- the channel model could be AWGN or other distributions, e.g., Rayleigh or Rice fading channel ;
- the transmitter could use any kinds of transmission scheme such as CDMA, FDMA, TDMA, etc. ;
- the receiver could use iterative techniques such as turbo codes for MIMO (Multiple Input Multiple Output) systems.

As an example, Fig. 2.7 (a) shows the normalized histograms of the soft output $X_i$ for BPSK system with AWGN and Rayleigh channel when $N = 1000$ and $SNR = 6\ dB$. For simplicity reasons, we only consider the PDF of the bits $-1$.

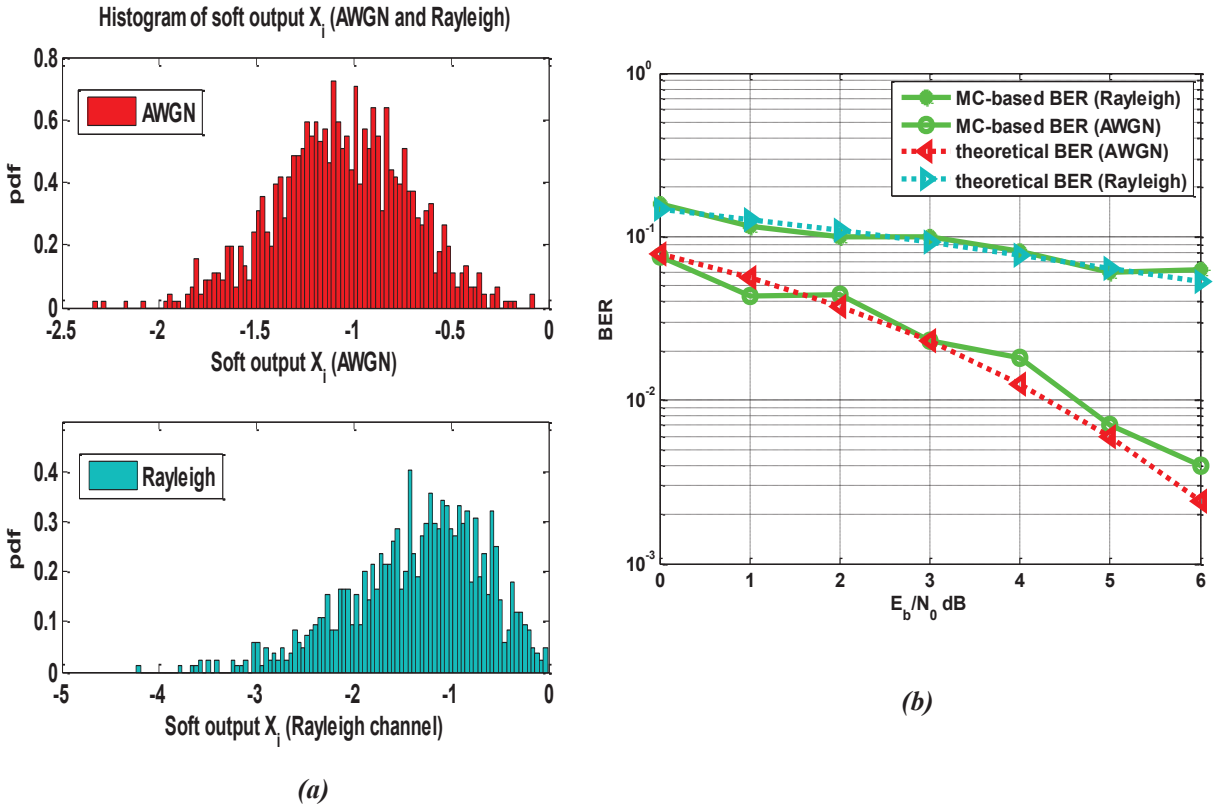


***Figure 2.7*** — (a). Normalized histogram of soft output $X_i$ ; (b). Monte-Carlo-based BER estimates for BPSK system for AWGN and Rayleigh channel ($N = 1000$, $SNR = 6\ dB$)

It can be shown that, for AWGN channel, the histogram (and the PDF) of the soft output is Gaussian, whereas for Rayleigh channel, the histogram (PDF) is vastly different. This causes a large difference between the Monte-Carlo-based BER estimates, as shown in Fig. 2.7 (b).

Fortunately, the Rayleigh distribution is well known, so that we still can utilize the parametric method to estimate the BER. However, this is not the case in practice :

- it is difficult to find the channel model, e.g. for indoor wireless communication system, the channel condition is difficult to be determined since it is affected by many factors, such as obstacles, moving terminals, etc. ;
- hence, the channel estimation could be imprecise and the performance of BER estimation will be degraded, e.g., using the LLR-based methods presented in section 2.1.5, we may have a bad precision of BER since the LLR distribution strongly depends on the accuracy of the SNR estimate.

Therefore, we suggest using the non-parametric and semi-parametric methods which only focus on the soft output distribution and "ignores" channel information.

## 2.4 Conclusion

We have firstly presented the conventional Monte-Carlo simulation, which demands very long computing time for small values of BER, and then we have discussed three specific techniques, Importance Sampling method, Tail Extrapolation method and Quasi-Analytical estimation, but yet the performance of these methods is strongly determined by the assumptions of parameters based on the behavior of real system. As for the BER estimation methods based on LLR distribution, the performance of the BER estimate depends strongly on the SNR uncerptainty.

In this report, instead of using the previous methods, we will suggest some new techniques based on the estimation of PDF of soft observed samples right before the hard decision. These PDF depends on the system scheme, the noise, the channel model, and are either difficult to know or arbitrarily chosen, which means that in practice the parametric PDF estimation technique cannot be used. Two PDF estimation techniques, non-parametric method and semi-parametric method, have been presented in this chapter. By using the PDFs estimates of receiver's soft observations, the BER estimate can be analytically calculated. In other words, the soft BER estimation is equivalent to estimating the PDFs of conditional soft observations.

The non-parametric method and the semi-parametric method can provide accurate PDF estimates for any digital communication system. The soft estimator could have no information about system scheme and channel model. Moreover, it is also possible that we do not need to know the transmitted data.

In Chapter 3, we will present the BER estimation technique based on non-parametric estimation using Kernel estimation. In our works, the Gaussian Kernel was used.

In Chapter 4, we will present the BER estimation technique based on semi-parametric estimation. In our works, we use the Gaussian Mixture Model to estimate the PDF.

CHAPTER 3

# Bit Error Rate estimation based on Kernel method

As discussed in Chapter 2, the conventional and modified Monte-Carlo techniques require the knowledge about transmitted data, system scheme and channel model. Moreover, excessively long computing time for small values of BER (high values of SNR) limits the performance of estimator in practical situation. Hence, the soft BER estimation techniques based on PDF estimation of receiver's soft outputs can be considered as an alternative to overcome these drawbacks.

In this chapter, we shall focus on the non-parametric Kernel-based PDF and BER estimation technique. For simplicity, we will consider the supervised case, which means that we still assume that the estimator perfectly knows transmitted data.

## 3.1 Properties of Kernel-based PDF estimator

In section 2.2.3.4, we have given a brief introduction to the Kernel density estimation technique. To understand the accuracy of estimator and evaluate the performances of KDE, we will present in this subsection the general properties of Kernel estimators.

An efficient way to quantify the accuracy of a density estimator is to measure the Mean Squared Error (MSE).

$$\begin{aligned} MSE(f_{KDE}(x)) &= E[(f_{KDE}(x) - f_X(x))^2] \\ &= (E[f_{KDE}(x)] - f_X(x))^2 + E[(f_{KDE}(x) - E[f_{KDE}(x)])^2] \\ &= Bias^2(f_{KDE}(x)) + Var(f_{KDE}(x)) \end{aligned} \tag{3.1}$$

A measure of the global accuracy of $f_{KDE}(x)$ is the Integrated Mean Squared Error (IMSE) :

$$\begin{aligned} IMSE(f_{KDE}(x)) &= E[\int_{-\infty}^{+\infty} (f_{KDE}(x) - f_X(x))^2 dx] = \int_{-\infty}^{+\infty} MSE(f_{KDE}(x))dx \\ &= \int_{-\infty}^{+\infty} Bias^2(f_{KDE}(x))dx + \int_{-\infty}^{+\infty} Var(f_{KDE}(x))dx \end{aligned} \tag{3.2}$$

It can be shown that there are two components that determine the MSE and IMSE : the bias and the variance.

### 3.1.1 Bias and variance of Kernel estimator

The expectation of Kernel estimator is given by Eq. 2.44. Note that $z = \frac{x-t}{h}$, and then $t = x - hz$ yields :

$$E[f_{KDE}(x)] = \int_{-\infty}^{+\infty} K(z) f_X(x - hz) dz$$

Expanding $f_X(x - hz)$ in a Taylor series, we obtain :

$$f_X(x - hz) = f_X(x) - hz f_X^{'}(x) + \frac{1}{2}(hz)^2 f_X^{''}(x) + o(h^2)$$

$o(h^2)$ can be ignored since it represents terms that converge to zero faster than $h^2$ as $h$ approaches zero.

Considering the normalization of Kernel, we can write :

$$\begin{aligned} E[f_{KDE}(x)] &= f_X(x) \int_{-\infty}^{+\infty} K(z) dz - h f_X^{'}(x) \int_{-\infty}^{+\infty} z K(z) dz + \frac{h^2}{2} f_X^{''}(x) \int_{-\infty}^{+\infty} z^2 K(z) dz \\ &= f_X(x) + \frac{h^2}{2} \upsilon_2 f_X^{''}(x) \end{aligned} \tag{3.3}$$

where $\upsilon_2$ represents the "variance" of the Kernel, $f_X^{''}(x)$ represents the second derivative of the density at the sample $x$.

Thus, we get :

$$Bias(f_{KDE}(x)) \approx \frac{h^2}{2} \upsilon_2 f_X^{''}(x) \tag{3.4}$$

Note that if $h \to 0$, the bias tends to zero.

Assume that the $X_i; i = 1, \ldots, N$ are independently distributed, the variance of Kernel estimate is given by :

$$Var(f_{KDE}(x)) = \frac{1}{(Nh)^2} \sum_{i=1}^{N} Var(K(\frac{x - X_i}{h})) \tag{3.5}$$

We can write :

$$Var(K(\frac{x - X_i}{h})) = E\Big[\Big(K(\frac{x - X_i}{h})\Big)^2\Big] - \Big(E\Big[K(\frac{x - X_i}{h})\Big]\Big)^2$$

Thus,

$$\begin{aligned} Var(f_{KDE}(x)) &= \frac{1}{Nh^2} \int (K(\frac{x - t}{h}))^2 f_X(t) dt - \frac{1}{Nh^2} (\int K(\frac{x - t}{h}) f_X(t) dt)^2 \\ &= \frac{1}{Nh^2} \int (K(\frac{x - t}{h}))^2 f_X(t) dt - \frac{1}{N} (f_X(x) + Bias(f_{KDE}(x)))^2 \end{aligned}$$

Substituting $z = \frac{x-t}{h}$ and applying Taylor approximation yields :

$$Var(f_{KDE}(x)) = \frac{1}{Nh}\int (K(z))^2 f_X(x - hz)dz - \frac{1}{N}(f_X(x) + o(h^2))^2$$

Note that if $h \to 0$ and $N \to \infty$, the above expression becomes approximately :

$$Var(f_{KDE}(x)) \approx \frac{1}{Nh} f_X(x) \int (K(z))^2 dz \tag{3.6}$$

It can be shown that the variance decreases as $Nh$ increases, and that the estimator is asymptotically unbiased as $h \to 0$ and $N \to \infty$ [SHG94, SGH97].

### 3.1.2 MSE and IMSE of Kernel estimator

The previous results for bias and variance of $f_{KDE}(x)$ lead to :

$$MSE(f_{KDE}(x)) \approx \frac{1}{Nh} f_X(x) M(K) + \frac{1}{4} h^4 \upsilon_2^2 (f_X''(x))^2 \tag{3.7}$$

and

$$IMSE(f_{KDE}(x)) \approx \frac{1}{Nh} M(K) + \frac{1}{4} h^4 \upsilon_2^2 J(f_X) \tag{3.8}$$

where :

$$\begin{cases} \upsilon_2 & = \int z^2 K(z)dz \\ M(K) & = \int (K(z))^2 dz \\ J(f_X) & = \int (f_X''(x))^2 dx \end{cases} \tag{3.9}$$

### 3.1.3 Kernel selection

From Eq. 3.8, we can observe that the IMSE can also be minimized with respect to the Kernel function. The Epanechnikov Kernel gives the lowest IMSE [Han09].

$$K_{Ep}(x) = \begin{cases} \frac{3}{4\sqrt{5}}(1 - \frac{1}{5}x^2) & \text{for } |x| < \sqrt{5}, \\ 0 & \text{otherwise.} \end{cases}$$

This result can be used to examine the impact of Kernel choice on the optimal IMSE. The efficiency for Epanechnikov Kernel is set to 1. The efficiency of a Kernel function, $K(x)$, compared to the optimal Epanechnikov Kernel, $K_{Ep}(x)$, is defined as [Han09] :

$$Efficiency(K(x)) = \Big(\frac{IMSE_{opt,Ep}(f_{K_{Ep}}(x))}{IMSE_{opt,K}(f_K(x))}\Big)^{\frac{5}{4}} = \Big(\frac{\upsilon_2^2 (M(K))^4 \text{ using } K_{Ep}}{\upsilon_2^2 (M(K))^4 \text{ using } K}\Big)^{\frac{1}{4}} \tag{3.10}$$

Tab. 3.1 shows the efficiencies for a number of well-known Kernels.

| Kernel | K(x) | Efficiency |
|---|---|---|
| Epanechnikov | $\begin{cases} \frac{3}{4\sqrt{5}}(1-\frac{1}{5}x^2) & \text{for } \lvert x\rvert < \sqrt{5}, \\ 0 & \text{otherwise.} \end{cases}$ | 1 |
| Triangular | $\begin{cases} 1-\lvert x\rvert & \text{for } \lvert x\rvert < 1, \\ 0 & \text{otherwise.} \end{cases}$ | 0.993 |
| Biweight | $\begin{cases} \frac{15}{16}(1-t^2)^2 & \text{for } \lvert x\rvert < 1, \\ 0 & \text{otherwise.} \end{cases}$ | 0.994 |
| Gaussian | $\frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}x^2}$ | 0.951 |
| Rectangular | $\begin{cases} \frac{1}{2} & \text{for } \lvert x\rvert < 1, \\ 0 & \text{otherwise.} \end{cases}$ | 0.930 |

**Table *3.1*** — Well-known Kernels and their efficiencies

Throughout this report, we suggest the use of the most popular Gaussian Kernel whose expression is given by Eq. 2.46.

### 3.1.4 Bandwidth (smoothing parameter) selection

As presented in section 2.2.3.4.2, the bandwidth $h$ controls the smoothness (or roughness) of a density estimate :

- a large bandwidth will over-smooth the density estimate and mask the structure of the data ;
- a small bandwidth will under-smooth the density estimate that is spiky and very hard to interpret.

As an example, we have simulated with different bandwidths the Kernel density estimates of the Gaussian normal distribution and two piecewise uniform distributions, as shown in Fig. 3.1.

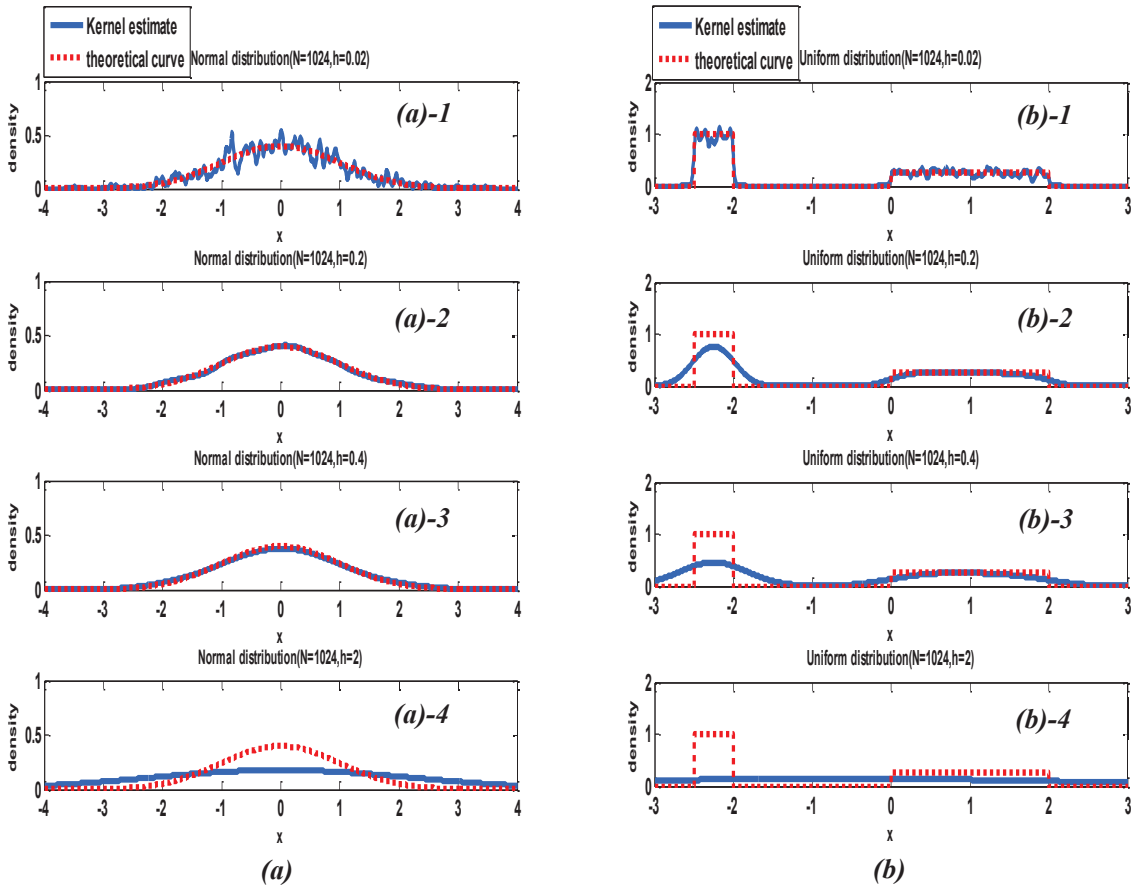


***Figure 3.1*** — (a). KDE of the normal distribution and (b). KDE of one distribution with two piecewise uniforms with different bandwidths ($h$ = 0.02, 0.2, 0.4 and 2), number of data points $N = 1024$

Fig. 3.1 (a)-1 and Fig. 3.1 (b)-1 show the under-smoothing of the density estimates due to small bandwidth.

When the bandwidth $h$ is equal to 0.2, the estimated PDF curve shown in Fig. 3.1 (a)-2 fits closely to the theoretical curve. The result shown in Fig. 3.1 (a)-3 is also not bad, only with a little distortion near the top of the curve. Fig. 3.1 (a)-4, (b)-2, (b)-3 and (b)-4 show that the bandwidth selection bears danger of over-smoothing since the selected bandwidth is too big. Selection of the bandwidth of Kernel estimator is a subject of considerable research. There are several popular methods based on the properties of the estimator.

#### 3.1.4.1 Subjective selection

The simplest solution consists in using different bandwidths and selecting one that looks right for the type of data under investigation.

As shown in Fig. 3.1, different distributions or even same distribution with different parameters may demand different bandwidths :

– for Gaussian distribution $X \sim N(0,1)$, the obtained estimate of Fig. 3.1 (a)-2 ($h = 0.2$) is better than the one of Fig. 3.1 (a)-1 ($h = 0.02$) and the one of Fig. 3.1 (a)-3 ($h = 0.4$) ;

– for uniform distribution, the case of Fig. 3.1 (b)-1 ($h = 0.02$) shows a better estimate than Fig. 3.1 (b)-2 ($h = 0.2$).

However, the bandwidth selection also depends on other parameters, such as the number of data points. We consider the same example in Fig. 3.1 but with fewer samples ($N = 128$).

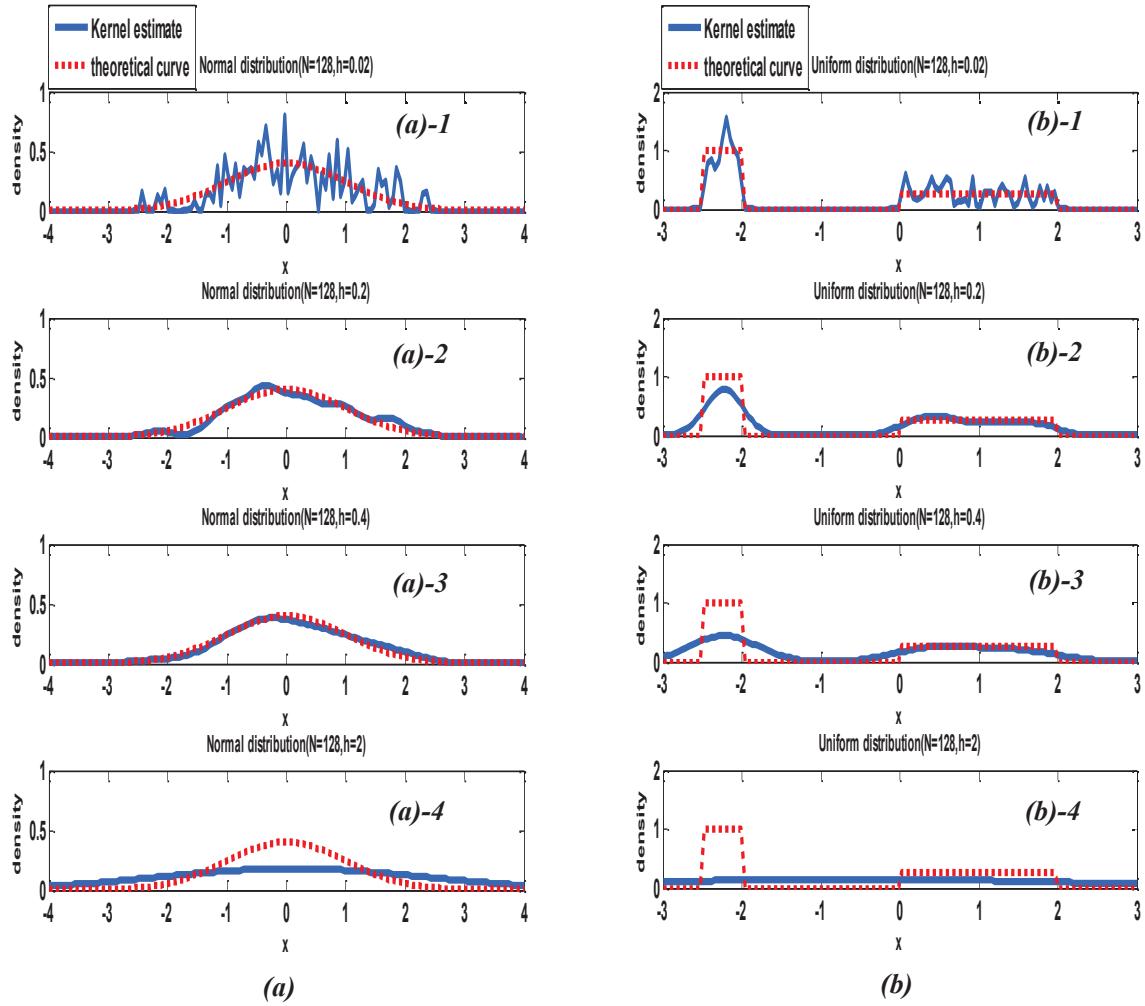


***Figure 3.2*** — (a). KDE of the normal distribution and (b). KDE of one distribution with two piecewise uniforms with different bandwidths ($h = 0.02$, 0.2, 0.4 and 2), number of data points $N = 128$

Comparing with the previous results in Fig. 3.1, the estimates for both two distributions become worse. Moreover, it can be shown that the estimated curve when $h = 0.4$ (Fig. 3.2 (a)-3) has fewer fluctuations than the one of Fig. 3.2 (a)-2, this means that the estimated result when $h = 0.2$ is no longer the optimal among the four obtained curves.

Furthermore, it can be shown that the obtained estimate of Fig. 3.2 (b)-1 becomes worse compared with the one of Fig. 3.1 (b)-1 when $h = 0.02$, however, the estimated curve of Fig. 3.2 (b)-2 ($h = 0.2$) is similar to the one of Fig. 3.1 (b)-2. This seems unusual since theoretically fewer data samples leads to a worse PDF estimation. One explanation is that for any data number and any distribution there is an optimal value of the bandwidth $h$ ; when $N = 128$, the bandwidth of Fig. 3.2 (b)-2 ($h = 0.2$) is close to the optimal value.

#### 3.1.4.2 Selection with reference to some given distribution : optimal smoothing parameter

It was shown from Eq. 3.8 that the IMSE of Kernel-based PDF estimate changes as a function of the bandwidth $h$. Let the soft output right before the hard decision, $(X_i)_{1\leq i\leq N}$, be random variables having the same PDF, $f_X(x)$. The Kernel-based PDF estimate, $f_{KDE}(x)$, is noted as $\hat{f}_X(x)$.

- for very small values of $h$ the first term in Eq. 3.8 becomes large, if $h$ tends towards 0, when $N$ tends towards $+\infty$, we have [STG09] :

$$E[\hat{f}_X(x)] \to f_X(x) \tag{3.11}$$

  the estimation of the PDF is asymptotically unbiased, but the estimate is under-smooth ;
- as $h$ gets larger the second term in Eq. 3.8 increases and the estimate becomes over-smooth.

**(a) General Kernel**

There is an optimal bandwidth which minimizes the IMSE. The first derivative is given by [TAS07] :

$$\frac{d(IMSE(\hat{f}_X(x)))}{dh} = h^3 \upsilon_2^2 J(f_X) - \frac{1}{Nh^2} M(K)$$

Setting this equal to zero yields the optimal bandwidth, given by :

$$h = \Big(\frac{M(K)}{N\upsilon_2^2 J(f_X)}\Big)^{\frac{1}{5}} \tag{3.12}$$

Substituting Eq. 3.12 for $h$ in Eq. 3.8 gives the minimal IMSE for the given PDF and Kernel function.

$$IMSE_{opt}(\hat{f}_X(x)) = \frac{5}{4}\Big(\frac{J(f_X)(M(K))^4\upsilon_2^2}{N^4}\Big)^{\frac{1}{5}} \tag{3.13}$$

Note that the optimal $h$ depends on the sample size, $N$, the unknown PDF, $f_X(x)$ and the Kernel function, $K(x)$. To derive the optimal bandwidth, we have to compute the value of $M(K)$, $\upsilon_2^2$ and $J(f_X)$ which depends on the unknown PDF, $f_X$.

Replacing $J(f_X)$ by its estimate $J(\hat{f}_X)$, we get :

$$IMSE \approx \frac{1}{Nh} M(K) + \frac{1}{4} h^4 \upsilon_2^2 J(\hat{f}_X) \tag{3.14}$$

where $M(K)$ and $J(\hat{f}_X)$ are given by Eq. 3.9.

**(b) Gaussian Kernel**

For the Gaussian Kernel, we have :

$$\begin{cases} K''(x) = (x^2 - 1)K(x) \\ \upsilon_2 = 1 \end{cases} \tag{3.15}$$

Using Eq. 2.44 and Eq. 3.15, we have [STG09] :

$$\begin{aligned} J(\hat{f}_X) = & \frac{1}{N^2 h^6} \sum_{i=1}^{N} \sum_{j=1}^{N} \Bigg\{ \int_{-\infty}^{+\infty} \Big[ \Big( \frac{x - X_i}{h} \Big)^2 - 1 \Big] \Big[ \Big( \frac{x - X_j}{h} \Big)^2 - 1 \Big] \\ & K\Big( \frac{2x - (X_i + X_j)}{\sqrt{2}h} \Big) K\Big( \frac{X_i - X_j}{\sqrt{2}h} \Big) dx \Bigg\} \end{aligned} \tag{3.16}$$

Let us note that :

$$\begin{cases} m = \frac{2x - (X_i + X_j)}{\sqrt{2}h} \\ n_{i,j} = \frac{X_i - X_j}{2h} \end{cases}$$

Then :

$$\Big[ \Big( \frac{x - X_i}{h} \Big)^2 - 1 \Big] \Big[ \Big( \frac{x - X_j}{h} \Big)^2 - 1 \Big] = \frac{m^4}{4} + m^2(2n_{i,j}^2 - 1) + (2n_{i,j}^2 - 1)^2 \tag{3.17}$$

Using Eq. 3.16 and Eq. 3.17, we have :

$$J(\hat{f}_X) = \frac{1}{N^2 h^5 \sqrt{2}} \sum_{i=1}^{N} \sum_{j=1}^{N} \Bigg\{ K(\sqrt{2} n_{i,j}) \int_{-\infty}^{+\infty} \Big[ \frac{m^4}{4} + m^2(2n_{i,j}^2 - 1) + (2n_{i,j}^2 - 1)^2 \Big] K(m) dm \Bigg\} \tag{3.18}$$

Since the Kernel function $K(\cdot) \sim N(0, 1)$, we obtain :

$$\begin{cases} \int m^2 K(m) dm = 1 \\ \int m^4 K(m) dm = 3 \end{cases}$$

Therefore, Eq. 3.18 can be rewritten as :

$$J(\hat{f}_X) = \frac{1}{N^2 h^5 \sqrt{2}} \sum_{i=1}^{N} \sum_{j=1}^{N} \left\{ K\Big(\frac{X_i - X_j}{\sqrt{2}h}\Big) \Big[ \Big(\frac{X_i - X_j}{\sqrt{2}h}\Big)^4 + \frac{3}{4} \Big] \right\} \tag{3.19}$$

Also, for a zero mean and unit variance Gaussian Kernel, we have :

$$M(K) = \int_{-\infty}^{+\infty} K^2(x)dx = \frac{1}{2\sqrt{\pi}} \tag{3.20}$$

If $f_X(x) \sim N(\mu, \sigma^2)$, we have :

$$J(\hat{f}_X) = \frac{3}{8\sqrt{\pi}\sigma^5} \tag{3.21}$$

Using Eq. 3.20 and Eq. 3.21 at the same time to resolve Eq. 3.14, we can compute the IMSE, and then derive the optimal smoothing parameter for a normal distribution by minimizing the IMSE. We get :

$$h = \Big(\frac{4}{3N}\Big)^{\frac{1}{5}} \sigma \tag{3.22}$$

**(c) Numerical illustration**

To be more specific, we will compute the MSE and the IMSE of KDE for the normal distribution with the same configurations as given by Fig. 3.1 (a) and Fig. 3.2 (a).

The Kernel function is a zero mean and unit variance Gaussian distribution and we utilize directly the normal PDF as the distribution $f_X$. Hence, we get :

$$\begin{cases} M(K) = \frac{1}{2\sqrt{\pi}} \\ \upsilon_2 = \frac{1}{\sqrt{2\pi}} \int z^2 \exp\Big(-\frac{z^2}{2}\Big) dz = 1 \end{cases} \tag{3.23}$$

The $2^{nd}$ derivative of $f_X(x)$ is given by :

$$f_X''(x) = \frac{1}{\sqrt{2\pi}} (x^2 - 1) \exp\Big(-\frac{x^2}{2}\Big)$$

Thus, we get :

$$MSE = \frac{1}{2\sqrt{2}Nh\pi} \exp\Big(-\frac{x^2}{2}\Big) + \frac{h^4}{4\sqrt{2\pi}} \Big[ (x^2 - 1) \exp\Big(-\frac{x^2}{2}\Big) \Big]^2 \tag{3.24}$$

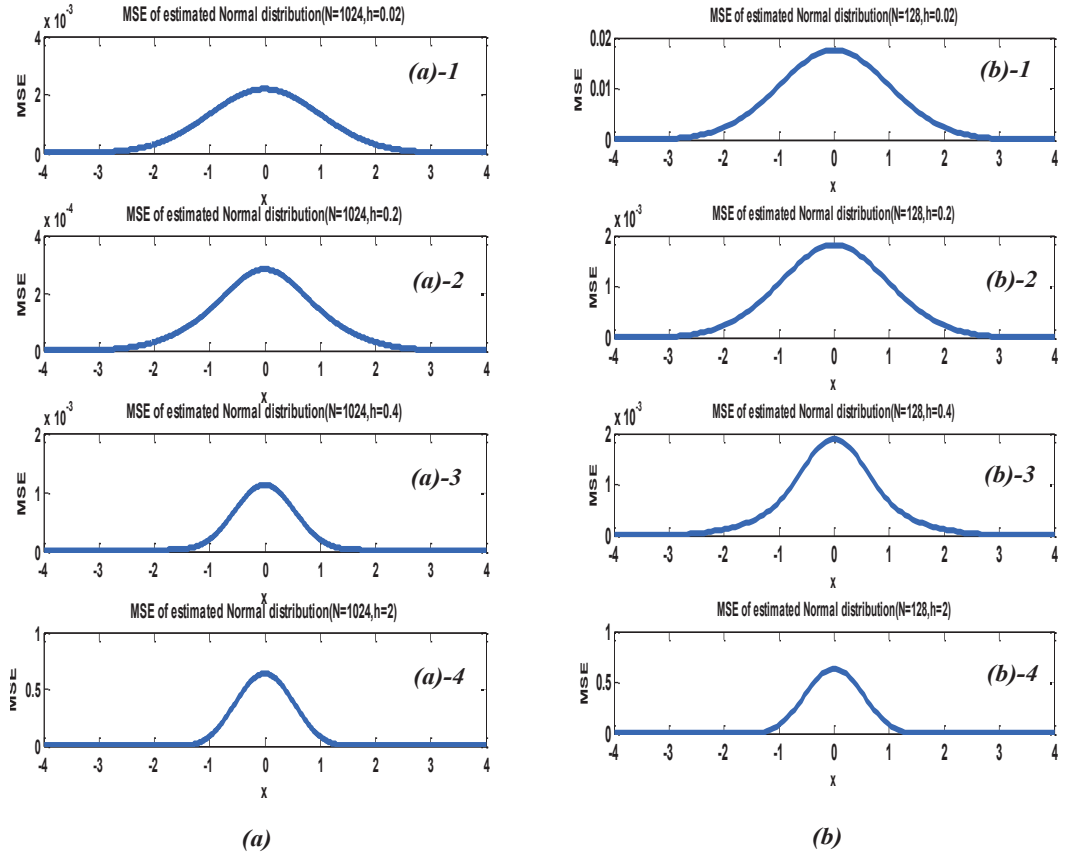


***Figure 3.3*** — MSE of the KDE for normal distribution with different bandwidth ($h = 0.02,\ 0.2,\ 0.4$ and 2) and different number of data points (a). $N = 1024$ (b). $N = 128$

Fig. 3.3 shows the MSE of the KDE with different bandwidth ($h = 0.02,\ 0.2,\ 0.4,\ 2$) and different number of data points ($N = 1024$ and 128). Compared with Fig. 3.1 (a) and Fig. 3.2 (a), we can easily find the general relationship between the MSE and the estimated density curves :

- in the case of 1024 data points, the value of MSE is minimum when $h = 0.2$, this means that the simulated curves shall be better fit to the true normal distribution than other configurations, as shown in Fig. 3.1 (a)-2. The optimal bandwidth computed by using Eq. 3.22 is about 0.265, which is close to 0.2 ;
- in the case of 128 data points, the value of MSE is minimum when $h = 0.4$, as shown in Fig. 3.2 (a)-3. This corresponds to the previous discuss in the above subsection. The optimal bandwidth is about 0.401, which is very close to 0.4 ;
- when $h = 2$, the values of MSE are maximum for both two configurations, the simulated curves are "very stable", as shown in Fig. 3.1 (a)-4 and Fig. 3.2 (a)-4.

Furthermore, we can also compute the IMSE. Using Eq. 3.23, we get :

$$IMSE = \frac{1}{2\sqrt{\pi}Nh} + \frac{3h^4}{32\sqrt{\pi}} \tag{3.25}$$

Tab. 3.2 gives the values of IMSE for different configurations of the KDE model.

| | $h = 0.02$ | $h = 0.2$ | $h = 0.4$ | $h = 2$ |
|---|---|---|---|---|
| $N = 1024$, IMSE : | 0.0137 | 0.0014 (min) | 0.0020 | 0.8464 |
| $N = 128$, IMSE : | 0.1101 | 0.0111 | 0.0068 (min) | 0.8473 |

**Table 3.2** — IMSE of KDE for normal distribution with different configurations

It can be shown that the obtained IMSE with $N = 1024$ are generally better than the IMSE with 128 samples. In fact, from Eq. 3.7 and Eq. 3.8, we can observed that the MSE and the IMSE are inversely proportion to the number of data points. This means that using more data samples is always helpful to Kernel PDF estimation.

It is also important to note that Eq. 3.22 only gives the optimal value of the bandwidth when the unknown PDF is Gaussian. For other distributions, this value could only be quasi-optimal. In our works, it is used as an initial condition for the iterative procedure of our simulations to compute the optimal smoothing parameter for non-Gaussian distribution (cf. section 3.2.2).

## 3.2 BER estimation based on Kernel method

### 3.2.1 PDF estimation based on Kernel method

Eq. 2.54 gives the expression of the output observations PDF which is a mixture of the two conditional PDFs.

The following notation is used. The soft output (right before the hard decision), $(X_i)_{1\leq i\leq N}$, are random variables having the same PDF, $f_X(x)$. We assume that we know the exact partitions of the observations $(X_i)_{1\leq i\leq N}$ into two classes $C_+$ and $C_-$ which respectively contains the observed output corresponding to the transmitted information bit $b_i = +1$ (resp., $b_i = -1$). Let $N_+$ (resp., $N_-$) be the cardinality of $C_+$ (resp., $C_-$), with $N = N_+ + N_-$.

The PDF estimation based on Kernel method is realized by using Eq. 2.38. The estimation of the conditional PDF, $f_X^{b_+}(x)$ (resp., $f_X^{b_-}(x)$), can be given by :

$$\begin{aligned}\hat{f}^{b_+}_{X,N_+}(x) &= \frac{1}{N_+ h_{N_+}} \sum_{X_i \in C_+} K\Big(\frac{x - X_i}{h_{N_+}}\Big) \\ \hat{f}^{b_-}_{X,N_-}(x) &= \frac{1}{N_- h_{N_-}} \sum_{X_i \in C_-} K\Big(\frac{x - X_i}{h_{N_-}}\Big)\end{aligned} \tag{3.26}$$

where $K(\cdot)$ is the Kernel function. As presented in section 3.1.3, in our works we use the Gaussian Kernel to estimate the PDF of soft observations. $h_{N_+}$ (resp., $h_{N_-}$) is the smoothing parameter which depends on the length of the observed samples, $N_+$ (resp., $N_-$).

Eq. 3.22 gives the general expression of the optimal smoothing parameter for Gaussian Kernel and normal distribution. For the system with two classes $C_+$ and $C_-$, we

can directly use the same expression :

$$\begin{cases} h_{N_+} = \left(\frac{4}{3N_+}\right)^{1/5} \sigma_+ \\ h_{N_-} = \left(\frac{4}{3N_-}\right)^{1/5} \sigma_- \end{cases} \tag{3.27}$$

### 3.2.2 Smoothing parameters optimization in practical situation

However, in the case of unknown PDF or unsupervised estimation (cf. Chapter 5) with Kernel-based method, it is impossible to derive the optimal IMSE smoothing parameters, and only approximations are generally used. Thus, the optimal smoothing parameters given by Eq. 3.27 are not exact. In this section, we will show how to use the Maximum Likelihood criterion for the computation of optimal smoothing parameter $h_{N_+}$ (resp., $h_{N_-}$) in an iterative way.

Suppose there is an independent and identically distributed sample of output of $N_+$ observations, $(X_i)_{1\leq i\leq N}$, having the same PDF estimate, $\hat{f}^{b_+}_{X,N_+}(\cdot|h_{N_+})$. It is demanded to find some estimates of $h_{N_+}$ which could be as close as possible to the true values which minimize the IMSE. Firstly, we specify the joint density function of all these observations, which can be written as :

$$\begin{aligned} \hat{f}^{b_+}_{X,N_+}(X_1,\ldots,X_{N_+}|h_{N_+}) &= \hat{f}^{b_+}_{X,N_+}(X_1|h_{N_+})\ldots\hat{f}^{b_+}_{X,N_+}(X_{N_+}|h_{N_+}) \\ &= \prod_{i=1}^{N_+} \hat{f}^{b_+}_{X,N_+}(X_i|h_{N_+}) \end{aligned} \tag{3.28}$$

Let us consider the observed values $(X_i)_{1\leq i\leq N}$ to be "fixed parameters" of the above function, whereas $h_{N_+}$ will be the unknown variable which is allowed to vary freely. The log-likelihood function is given by :

$$L(h_{N_+}|X_1,\ldots,X_{N_+}) = \hat{f}^{b_+}_{X,N_+}(X_1,\ldots,X_{N_+}|h_{N_+}) = \prod_{i=1}^{N_+} \hat{f}^{b_+}_{X,N_+}(X_i|h_{N_+}) \tag{3.29}$$

$$\log\Big[L(h_{N_+}|X_1,\ldots,X_{N_+})\Big] = \log\Big[\prod_{i=1}^{N_+} \hat{f}^{b_+}_{X,N_+}(X_i|h_{N_+})\Big] \tag{3.30}$$

We utilize the Maximum Likelihood method to estimate the smoothing parameter by finding a value of $h_{N_+}$ which maximizes the log-likelihood function, which means :

$$h^{\star}_{N_+} = \arg\max_{h_{N_+}}\Big\{\log\Big[L(h_{N_+}|X_1,\ldots,X_{N_+})\Big]\Big\} \tag{3.31}$$

where :

$$\begin{aligned} \log\Big[L(h_{N_+}|X_1,\ldots,X_{N_+})\Big] &= \sum_{i=1}^{N_+} \log\Big[\frac{1}{N_+h_{N_+}} \sum_{j=1}^{N_+} K\Big(\frac{X_i - X_j}{h_{N_+}}\Big)\Big] \\ &= \sum_{i=1}^{N_+} \log\Big[\frac{1}{N_+h_{N_+}}\Big(\sum_{j=1,j\neq i}^{N_+} K\Big(\frac{X_i - X_j}{h_{N_+}}\Big) + \frac{1}{\sqrt{2\pi}}\Big)\Big] \end{aligned} \tag{3.32}$$

As an example, considering a decorrelator-based receiver of CDMA system (cf. section 3.2.2.2), Fig. 3.4 shows the log-likelihood as a function of $h_{N_+}$ and $h_{N_-}$ with 1000 soft observations.

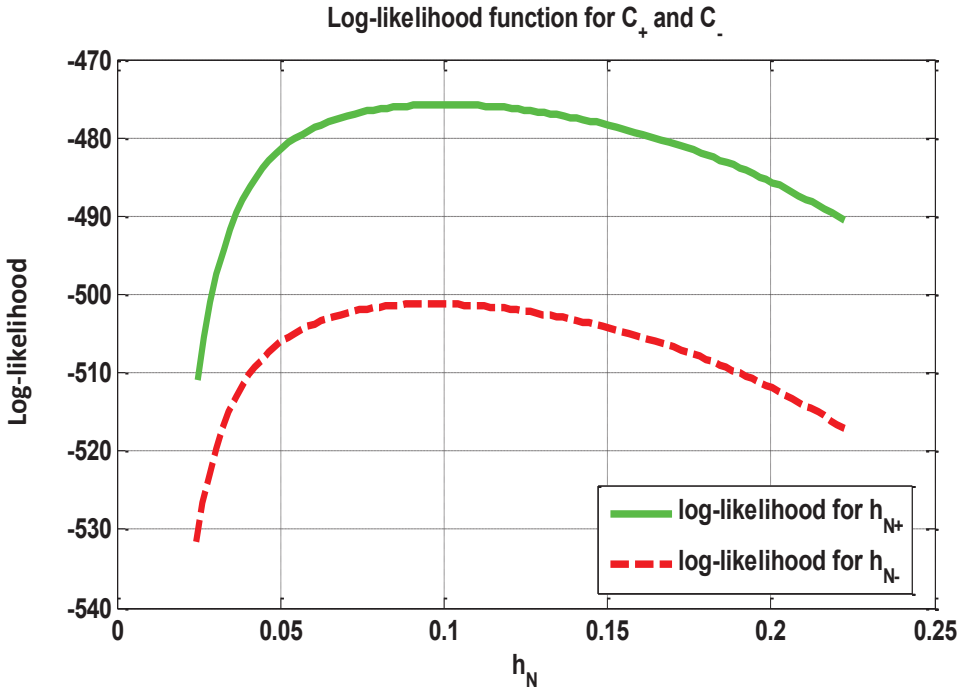


***Figure 3.4*** — Behavior of log-likelihood function for decorrelator-based CDMA system

From 3.4, we observe that there is a maximum value of the log-likelihood function. The optimal value of $h_{N_+}$ (resp., $h_{N_-}$), $h^{\star}_{N_+}$ (resp., $h^{\star}_{N_-}$), can be either directly derived from Eq. 3.32, or computed by using the derivative of the log-likelihood function. It is easier to find the value of $h_{N_+}$ which cancels the derivative of the considered log-likelihood function, which is given by :

$$\begin{aligned}\Big\{\log\Big[L(h_{N_+}|X_1,\ldots,X_{N_+})\Big]\Big\}' &= \frac{d\Big\{\Big[L(h_{N_+}|X_1,\ldots,X_{N_+})\Big]\Big\}}{d(h_{N_+})} \\ &= \frac{d\Big\{\Big[\sum_{i=1}^{N_+}\Big(\log\frac{1}{N_+}+\log\frac{1}{h_{N_+}}+\log\Big(\sum_{j=1}^{N_+}K\big(\frac{X_i-X_j}{h_{N_+}}\big)\Big)\Big)\Big]\Big\}}{d(h_{N_+})} \\ &= -\frac{N_+}{h_{N_+}}+\sum_{i=1}^{N_+}\frac{\sum_{j=1}^{N_+}K\big(\frac{X_i-X_j}{h_{N_+}}\big)\big(\frac{(X_i-X_j)^2}{h^3_{N_+}}\big)}{\sum_{j=1,j\neq i}^{N_+}K\big(\frac{X_i-X_j}{h_{N_+}}\big)}\end{aligned} \tag{3.33}$$

**Remark** : for normal Kernel, $K(\frac{X_i-X_j}{h_{N_+}})=\frac{1}{\sqrt{2\pi}}$ if $i=j$.

Simulations prove that the derivative of the considered likelihood function is strictly monotonous as a function of the smoothing parameter, as shown in Fig. 3.5.

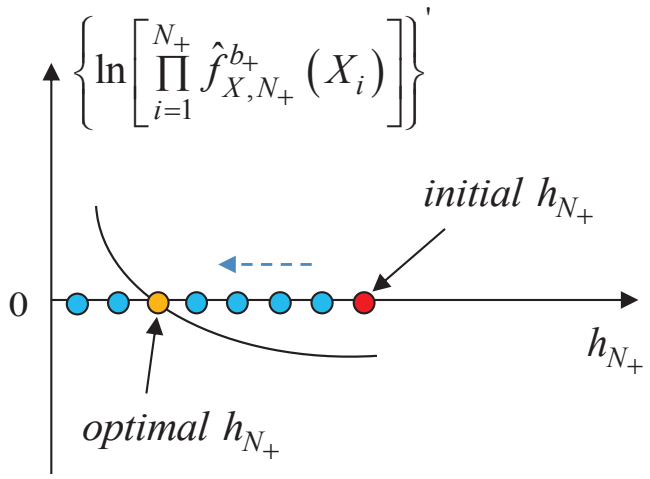


***Figure 3.5*** — Relationship between the log-likelihood function and the smoothing parameter

The zeros of Eq. 3.32 and Eq. 3.33 would be difficult to be solved analytically ; the simulation run time is quite long. We would like to suggest two other methods to compute the optimal smoothing parameter with fewer simulation run time.

#### 3.2.2.1 Curve fitting method

By using the initial value of $h_{N_+}$ given by Eq. 3.27, we introduce the curve fitting method to simplify the computation of the optimal smoothing parameter.

Curve fitting is the process of constructing a curve, which has the best fit to a series data points. Furthermore, the obtain curve is presented by a mathematical expression which is simpler than the considered function (i.e. Eq. 3.32 and Eq. 3.32).

There are many methods used to curve fitting, such as exponential model and Gaussian model. In our works, we have chosen the polynomial model which is the most popular method. Fig. 3.6 presents the flow chart of this method to derive the optimal $h_{N_+}$.

It is clear that if the real root of the obtained polynomial expression is not unique, there will be several intersection points of X-axis (the axis of $h_{N_+}$), which means that the obtained polynomial function is not monotonous. In this situation, we need to increase the order $P$ of the polynomial equation.

We have simulated the performance of the polynomial method for curve fitting of the derivative of log-likelihood function. For the same decorrelator-based CDMA system used in Fig. 3.4, curve fitting result of this derivative function is presented in Fig. 3.7 when order $P$ is equal to 5. We can observe that the dotted line has at least three intersection points with $h_{N_+}$-axis since the fitting curve is not monotonous. Therefore, the "optimal" smoothing parameter is not unique, which means that we have to use a higher-ordered polynomial model.

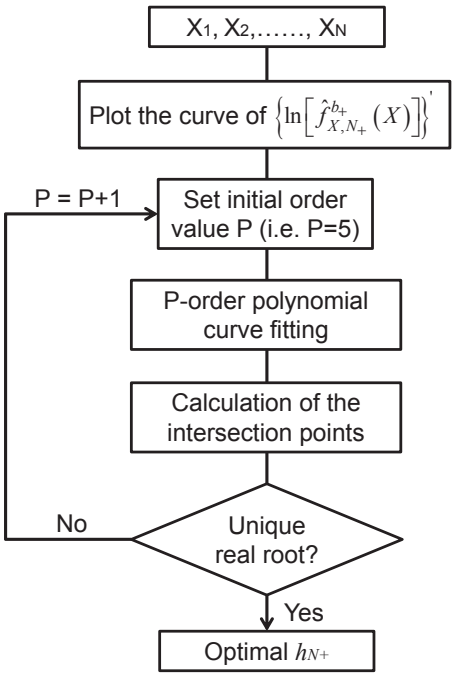


***Figure 3.6*** — Flow chart for smoothing parameter optimization by using the curve fitting method

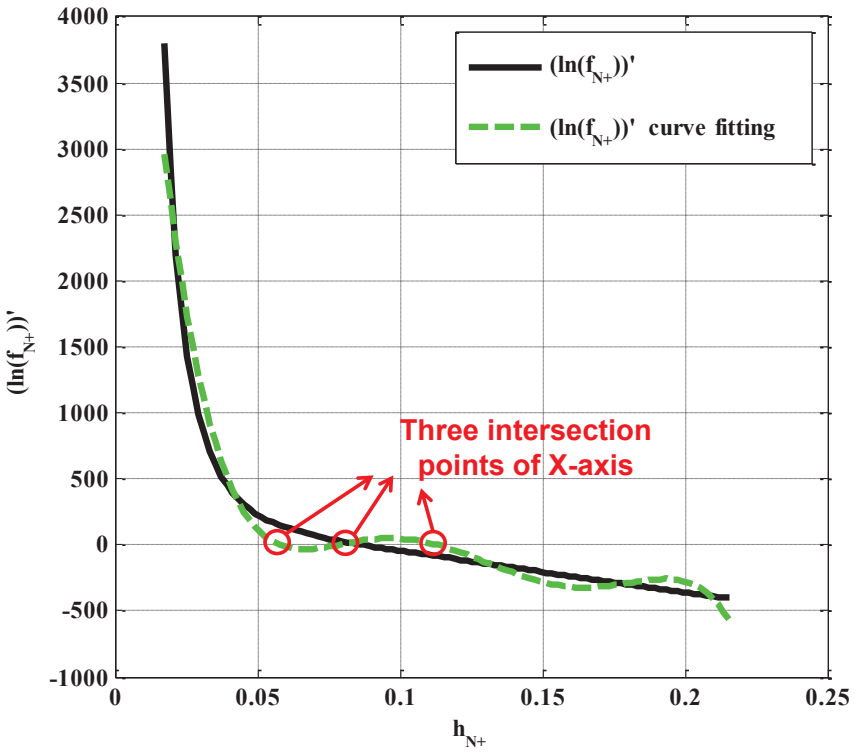


***Figure 3.7*** — Derivative of the log-likelihood by curve fitting method($P = 5$)

To apply this polynomial curve fitting method, we need a large number of outputs at the receiver to correctly obtain the original mathematical expression of the considered function. Another drawback is that we also need a high-ordered polynomial model in order to approximate and accurately fit the true function, which increases the complexity and computation time.

Fig. 3.8 shows the simulation result for curve fitting of derivative function when $P = 7$. It is shown that the obtained root is unique.

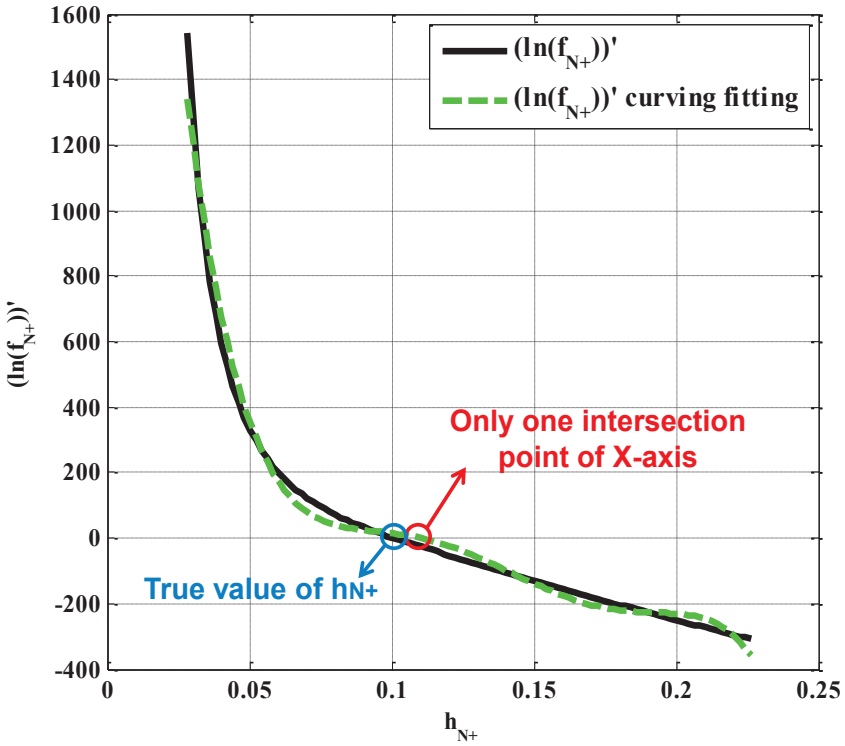


***Figure 3.8*** — Derivative of the log-likelihood by curve fitting method($P = 7$)

However, this does not mean that we can always obtain an accurate estimate of $h^{\star}_{N_+}$ or $h^{\star}_{N_-}$. As shown in Fig. 3.8, we can observe that there is still a difference (about 0.01) between the true value and the obtained value of $h^{\star}_{N_+}$. In this situation, we must increase the fitting order again, or we choose another type of fitting model. Moreover, increasing the order $P$ may lead to "over fitting" of the derivative of log-likelihood function.

Fortunately, for most cases that we have tested, the curve fitting method with polynomial model works quite well. The problem only shows up when the true log-likelihood function and its derivative have a great turning point, as shown in Fig. 3.4 and Fig. 3.7.

#### 3.2.2.2 Newton's method

We suggested another method called Newton's method to find the optimal smoothing parameter. Fig. 3.9 illustrates the flow chart of this method for the computation of $h^{\star}_{N_+}$ by using the initial smoothing parameter $h_{N_+}$.

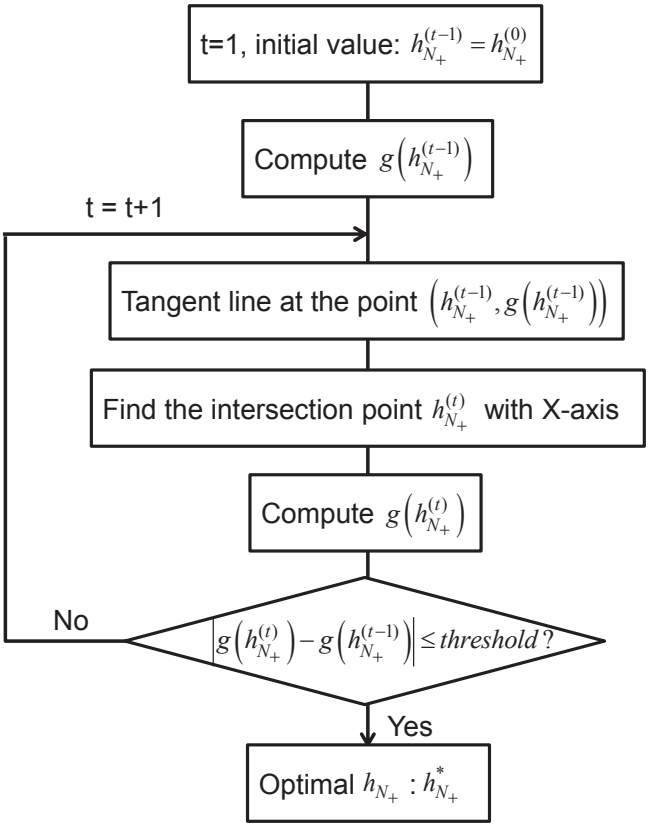


***Figure 3.9*** — Flow chart for smoothing parameter optimization by using Newton's method

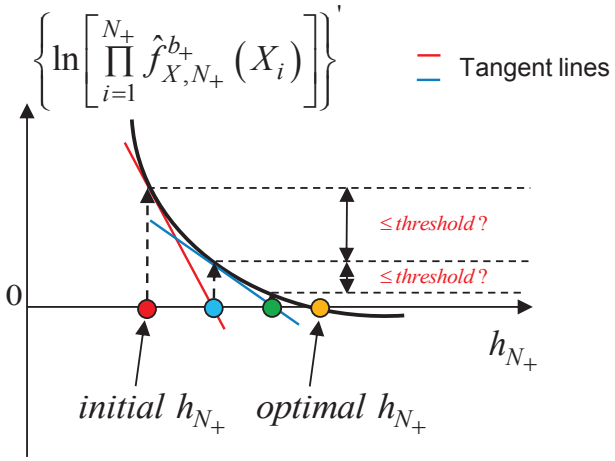


***Figure 3.10*** — Principal of Newton's method

Fig. 3.10 shows the principle of Newton's method. Firstly, with the initial value of the smoothing parameter $h_{N_+}$ (red point in Fig. 3.10), we calculate the corresponding value of the considered derivative function by using Eq. 3.33. Then we can find the tangent line (red line in Fig. 3.10) of this point and obtain the first intersection point of X-axis (blue point in Fig. 3.10). This first intersection is considered as a new updated value of $h_{N_+}$. By repeating this process, the second intersection point (blue point in Fig. 3.10) can be found, which is much closer to the optimal value of $h_{N_+}$. After several times of iterative calculations, with the threshold condition, we can finally find a $h_{N_+}$ (green point in Fig. 3.10) which is very close to the real optimal value.

Let $g(\cdot)$ be the derivative of the log-likelihood function. To find the unique root of the $g(\cdot)$ function, Newton's algorithm can be implemented as follows [Kel03].

Let the initial condition be :

$$h_{N_+}^{(0)} = \Big(\frac{4}{3N_+}\Big)^{\frac{1}{5}}\sigma_+ \tag{3.34}$$

At each iteration $t$, a new value of $h_{N_+}^{(t)}$ is given by using the previous value $h_{N_+}^{(t-1)}$ :

$$h_{N_+}^{(t)} = h_{N_+}^{(t-1)} - \frac{g(h_{N_+}^{(t-1)})}{g^{'}(h_{N_+}^{(t-1)})} \tag{3.35}$$

where :

$$g(h_{N_+}^{(t-1)}) = \Big[\log\Big(\prod_{i=1}^{N_+}\hat{f}_{X,N_+}^{b_+}(X_i)\Big)\Big]^{'}_{(t-1)} \tag{3.36}$$

The difference between the obtained values of derivative function of two sequential calculations is then compared with the threshold.

$$\left|\Big[\log\Big(\prod_{i=1}^{N_+}\hat{f}_{X,N_+}^{b_+}(X_i)\Big)\Big]^{'}_{(t)} - \Big[\log\Big(\prod_{i=1}^{N_+}\hat{f}_{X,N_+}^{b_+}(X_i)\Big)\Big]^{'}_{(t-1)}\right| \leq \text{threshold} \tag{3.37}$$

In our works, the threshold is set to $10^{-4}$. If the obtained difference in Eq. 3.37 after $t$ times iterations is sufficient small, the $h_{N_+}^{(t)}$ can be considered as the "optimal" smoothing parameter.

### 3.2.3 BER calculation with Kernel-based PDF estimates

Using Eq. 3.26, we can evaluate the expression of Eq. 2.55.

We use the following change of variable : $t = \frac{x - X_i}{h_{N_+}}$.

For the Gaussian Kernel, we have :

$$\begin{aligned}\int_{-\infty}^{0} \hat{f}_{X,N_+}^{b_+}(x)dx &= \sum_{X_i \in C_+} \int_{-\infty}^{-\frac{X_i}{h_{N_+}}} \frac{1}{N_+} K(t)dt \\ &= \frac{1}{N_+} \sum_{X_i \in C_+} \int_{\frac{X_i}{h_{N_+}}}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \\ &= \frac{1}{N_+} \sum_{X_i \in C_+} Q\Big(\frac{X_i}{h_{N_+}}\Big)\end{aligned} \tag{3.38}$$

and

$$\begin{aligned}\int_{0}^{+\infty} \hat{f}_{X,N_-}^{b_-}(x)dx &= \sum_{X_i \in C_-} \int_{-\frac{X_i}{h_{N_-}}}^{+\infty} \frac{1}{N_-} K(t)dt \\ &= \frac{1}{N_-} \sum_{X_i \in C_-} \int_{-\frac{X_i}{h_{N_-}}}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \\ &= \frac{1}{N_-} \sum_{X_i \in C_-} Q\Big(-\frac{X_i}{h_{N_-}}\Big)\end{aligned} \tag{3.39}$$

where $Q(\cdot)$ denotes the complementary cumulative Gaussian distribution function, as presented in section 2.2.2.1. $Q(\cdot)$ can also be expressed by the erfc function as follows :

$$Q(x) = \frac{1}{2}\mathrm{erfc}\Big(\frac{x}{\sqrt{2}}\Big) \tag{3.40}$$

Using Eq. 3.38 and Eq. 3.39, we can evaluate the expression of the soft BER estimate for the Gaussian Kernel :

$$\hat{p}_{e,N} = \frac{\pi_+}{N_+} \sum_{X_i \in C_+} Q\Big(\frac{X_i}{h_{N_+}}\Big) + \frac{\pi_-}{N_-} \sum_{X_i \in C_-} Q\Big(-\frac{X_i}{h_{N_-}}\Big) \tag{3.41}$$

For Gaussian distribution, the smoothing parameters $h_{N_+}$ and $h_{N_+}$ are computed by using Eq. 3.27. For non-Gaussian PDF, they should be computed in an iterative way by using the proposed methods as presented in section 3.2.2.

### 3.2.4 MSE of Kernel-based soft BER estimator

We have given the expression of the BER by Eq. 3.41, in this section we shall study the convergence of the suggested BER estimator in the sense of MSE.

The MSE of the soft BER estimation can be written as :

$$\begin{aligned} MSE(\hat{p}_e(x)) &= MSE(\hat{p}_{e,N}) = E[\hat{p}_e(x) - p_e]^2 \\ &= \big(E[\hat{p}_e(x)] - p_e\big)^2 + E\big[\hat{p}_e(x) - E[\hat{p}_e(x)]\big]^2 \\ &= Bias^2(\hat{p}_e(x)) + Var(\hat{p}_e(x)) \end{aligned} \tag{3.42}$$

where $p_e$ is the true BER.

In order to study the convergence of the MSE, we have to find the bias and the variance of the soft BER estimator. According to the results shown in [STG09], we can obtain the following theorems.

Assume that the conditional PDF, $f_X^{b_+}(x)$ (resp., $f_X^{b_-}(x)$), is a second derivative PDF function, that $h_{N_+}$ (resp., $h_{N_-}$) $\to 0$ as $N \to 0$. Then the soft BER estimation is asymptotically unbiased, i.e.,

$$\lim_{N\to\infty} E[\hat{p}_{e,N}] = p_e \tag{3.43}$$

Assume that $f_X^{b_+}(x)$ (resp., $f_X^{b_-}(x)$) is a second derivative PDF function, that $h_{N_+}$ (resp., $h_{N_-}$) $\to 0$ as $N \to 0$. Then variance of the soft BER estimation tends to zero as $N$ tends to $+\infty$, i.e.,

$$\lim_{N\to\infty} E\big[\hat{p}_{e,N} - E[\hat{p}_{e,N}]\big]^2 = 0 \tag{3.44}$$

Using the two above theorems, it can be shown that the MSE of the soft BER estimation tends to 0 as $N$ tends to $+\infty$.

Assume that $f_X^{b_+}(x)$ (resp., $f_X^{b_-}(x)$) is a second derivative PDF function, that $h_{N_+}$ (resp., $h_{N_-}$) $\to 0$ as $N \to 0$. Then MSE of the soft BER estimation tends to zero as $N$ tends to $+\infty$, i.e.,

$$\lim_{N\to\infty} E[\hat{p}_{e,N} - p_e]^2 = \lim_{N\to\infty} MSE(\hat{p}_{e,N}) = 0 \tag{3.45}$$

## 3.3 Simulation results of BER estimation based on Kernel method

To evaluate the performance of the proposed non parametric BER estimation based on Kernel method, we have considered four different frameworks :

- BPSK sequence over AWGN and Rayleigh channels ;
- CDMA system ;
- Turbo coding system ;
- LDPC coding system.

### 3.3.1 Sequence of BPSK symbol over AWGN and Rayleigh channels

Let $(b_i)_{1\leq i\leq N} \in \{-1, +1\}$ be a sequence of $N$ BPSK signal, transmitted over an AWGN channel without using any coding technique and any kinds of transmission scheme. To simplify the simulation, we only consider the cardinality of $C_-$, with $N = N_-$, which means that the transmitted bits $(b_i)_{1\leq i\leq N}$ are equal to -1. The conditional PDF estimation of $f_X^{b_-}(x)$ based on Kernel method is realized by using Eq. 3.26.

Fig. 3.11 shows the simulation results of the BER estimation for AWGN channel when $N = 2000$. We compare the Kernel-based BER estimates of three cases :

- without using the proposed methods for smooth parameter optimization but using the initial values given by Eq. 3.27 ;
- using Newton's method for smooth parameter optimization ;
- using curve fitting method for smooth parameter optimization, the chosen order $P$ of polynomial model is equal to 10 (with unique root) ;

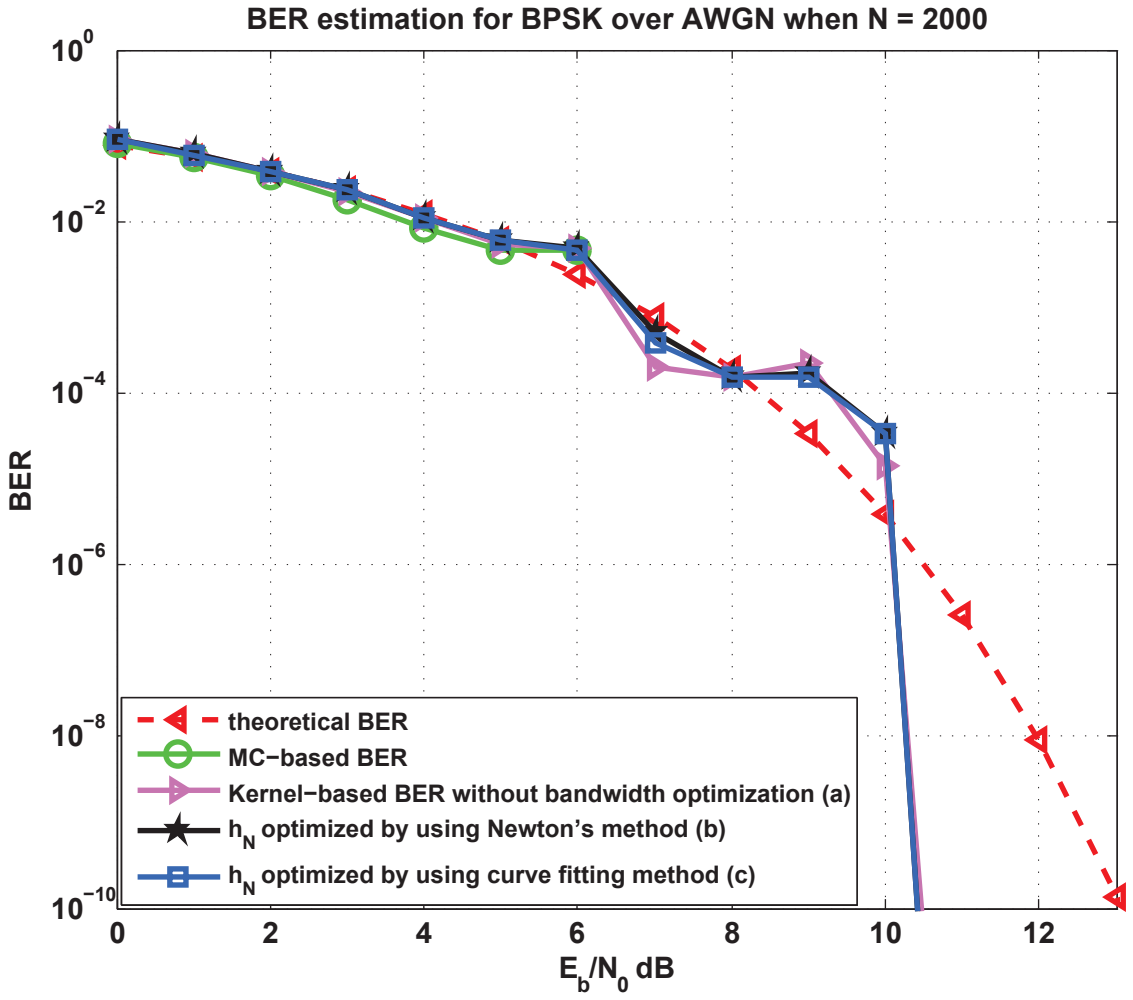


***Figure 3.11*** — BER estimate based on Kernel method for BPSK system over AWGN channel when $N = 2000$

The Monte-Carlo simulation works well for small values of SNR. When $SNR > 6\ dB$, the Monte-Carlo method cannot return non-zero value for BER estimation. Whereas, the Kernel-based BER estimates work quasi-well (not very close to the theoretical values) till $SNR = 10\ dB$.

Moreover, we can clearly observe that the BER estimated by using the optimized smooth parameters has similar precision to the one without bandwidth optimization.

TAB 3.3 shows the value of non-optimized and optimized smooth parameter for different SNR. The corresponding BER estimates are also given and compared with the theoretical values.

| SNR | 1 $dB$ | 3 $dB$ | 5 $dB$ | 7 $dB$ | 9 $dB$ |
|---|---|---|---|---|---|
| Theoretical BER | 0.0563 | 0.0229 | $5.954 \cdot 10^{-3}$ | $7.727 \cdot 10^{-4}$ | $3.362 \cdot 10^{-5}$ |
| $h_{N_-}$ given by Eq. 3.27 | 0.1450 | 0.1140 | 0.0901 | 0.0720 | 0.0597 |
| BER estimate (a) | 0.0608 | 0.0224 | $5.565 \cdot 10^{-3}$ | $2.033 \cdot 10^{-4}$ | $1.529 \cdot 10^{-4}$ |
| $h_{N_-,\text{Newton's method}}$ | 0.1098 | 0.1242 | 0.1091 | 0.0900 | 0.0476 |
| BER estimate (b) | 0.0589 | 0.0231 | $5.967 \cdot 10^{-3}$ | $3.767 \cdot 10^{-4}$ | $9.516 \cdot 10^{-5}$ |
| $h_{N_-,\text{curve fitting}}$ | 0.1363 | 0.1245 | 0.1094 | 0.0902 | 0.0518 |
| BER estimate (c) | 0.0602 | 0.0231 | $5.973 \cdot 10^{-3}$ | $3.749 \cdot 10^{-4}$ | $1.245 \cdot 10^{-4}$ |

**Table *3.3*** — Comparison of theoretical BER and BER estimates computed by using (a). non-optimized smooth parameter, (b). Newton's method-based optimized smooth parameter, (c). curve fitting-based optimized smooth parameter

It can be shown that :

- the BER estimates using initial values of smoothing parameters and based on the curve fitting or Newton's method are all close to the theoretical values. In fact, because of AWGN channel, the initial values of $h_{N_+}$ and $h_{N_-}$ are considered as the optimal one ;
- as SNR get higher, the BER estimates become imprecise, e.g. for $SNR = 9\ dB$, the BER estimates are about $3 \sim 5$ times bigger than the theoretical value.

When $SNR > 10\ dB$, the values of BER estimates decline sharply, even for the one using optimized smooth parameter. In theory, this is caused by the limitation of length of database (we only have 2000 samples). However, while taking 20,000 data samples, the performance of BER estimation is still similar to the previous case.

To explain further, we consider the normalized histogram and the PDF estimate (obtained by using Newton's method-based optimized smooth parameter, when $N = 2000$ and $SNR = 10\ dB$) of the soft output of BPSK receiver, as shown in FIG 3.12.

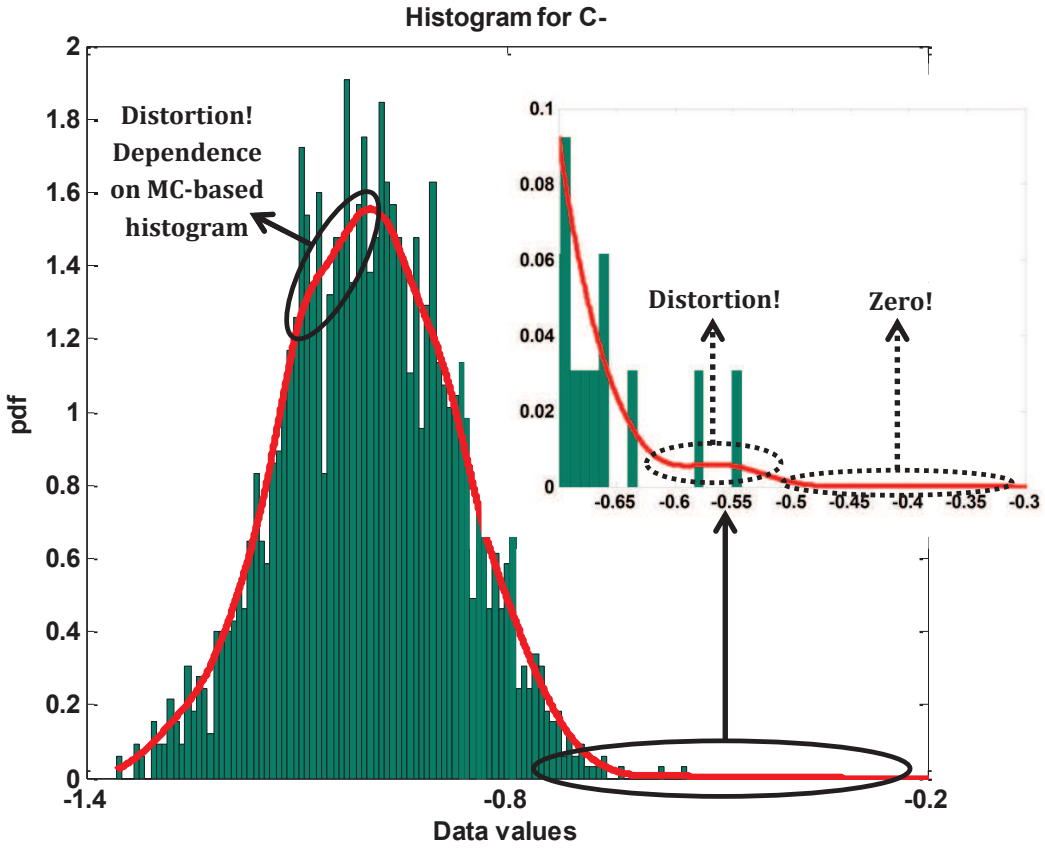


***Figure 3.12*** — Histogram and Kernel-based PDF estimate of soft output of BPSK receiver when $N = 2000$ and $SNR = 10\ dB$

It can be shown that there is a conformance between the Kernel-based PDF estimate (red curve) and the Monte-Carlo-based histogram. In theory, the true PDF is Gaussian since the BPSK signal is transmitted over AWGN channel, whereas we found an obvious "distortion" near the top of the kernel-based PDF curve. In fact, this distortion is completely caused by small value of number count.

Furthermore, it is important to note that the BER estimate depends on the accuracy of PDF estimation in the "error area" (tail of PDF). For our case, this corresponds to the area where the value of soft output $X_i$ is greater than zero. However, in the zoomed-in figure, we found another distortion at the position of the PDF curve edge, because the studied SNR is sufficient high so that there are only very limited number of samples far from the center of distribution. Also, we found that the density declined to zero due to lack of sample count. For this reason, the BER estimate declined sharply to very small value when SNR is high.

If we focus on the "average" observations by performing several trials of simulation, the histogram could be exactly Gaussian without oscillation and the distortions could disappear. However, this does not make any sense since this is equivalent to increase the size of dataset, whereas, in practice, we may have only few data samples as for many practical systems it is required to estimate the real-time BER with only limited number of data frames.

In a word, the dependence of Kernel-based PDF estimate on Monte-Carlo simulation may present bad performance. For Gaussian distribution, we suggest use Gaussian Mixture Model for PDF and BER estimation. In Chapter 4, we can find that the Gaussian Mixture Model-based PDF estimate is very close to the true PDF, and the BER estimation has a much better precision compared with the Kernel-based one.

We have also performed the Kernel-based BER estimation for Rayleigh channel with different size of dataset. We compared the BER estimates based on smoothing parameters optimization using Newton's method with the BER estimates computed with the Gaussian initial values.

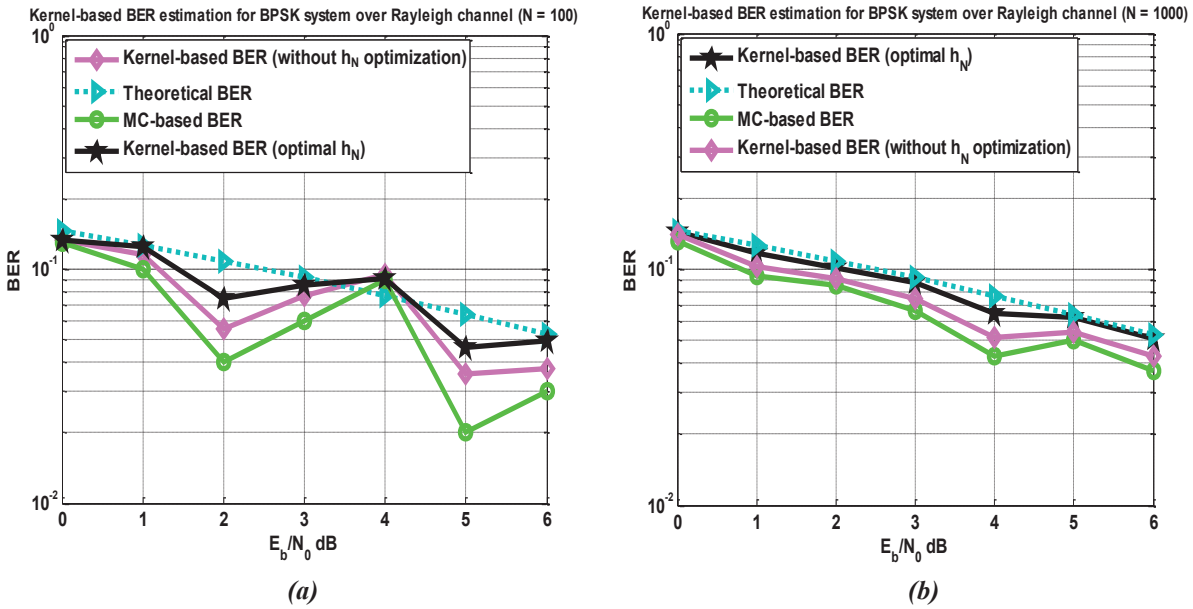


***Figure 3.13*** — Kernel-based BER estimation for BPSK signal over Rayleigh channel when (a). $N = 100$, (b). $N = 1000$

In Fig. 3.13, we can find that for small number of data samples, the Kernel-based BER presents better performance than the one using Monte-Carlo simulation, even for only 100 data points (cf. Fig. 3.13 (a)). Moreover, because of Rayleigh channel, it is obvious that the BER estimates based on smoothing parameter optimization using Newton's method have better precision compared with the one with Gaussian initial values of $h_{N_+}$ and $h_{N_-}$.

Also, the Kernel-based BER estimate changes with the SNR as regularly as the Monte-Carlo-based one does. As previously discussed, this is due to the dependence of Kernel-based PDF estimate on Monte-Carlo simulation.

Fig. 3.14 shows the Kernel-based PDF estimates for the case of Rayleigh channel when $N = 100$ and 1000. The results are given at $SNR = 6\ dB$.

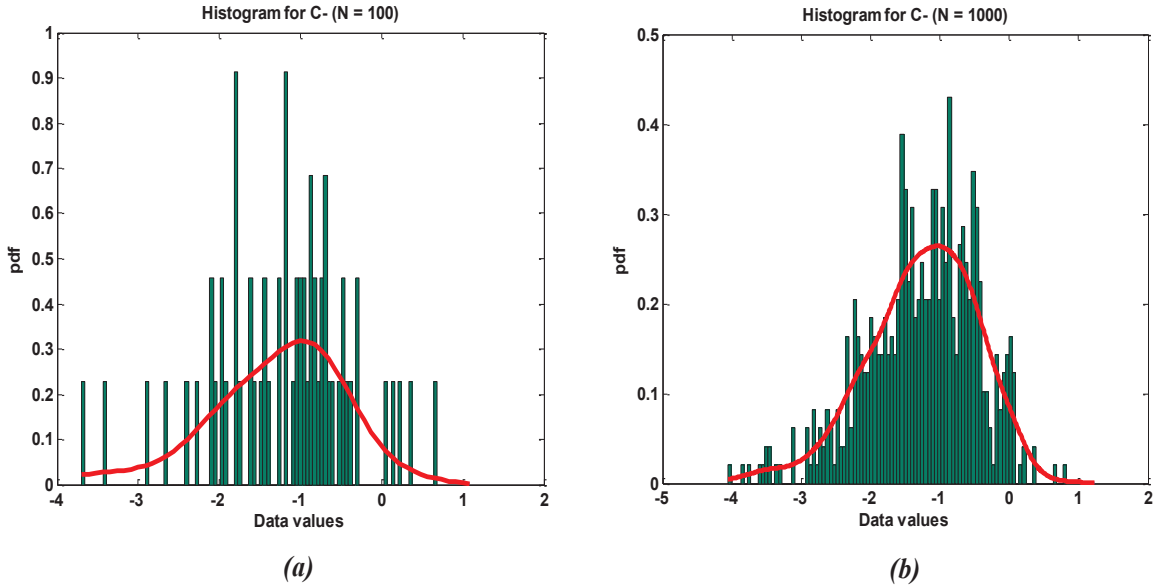


***Figure 3.14*** — Kernel-based PDF estimation for BPSK signal over Rayleigh channel when (a). $N = 100$, (b). $N = 1000$

For small values of SNR, even if taking very few data samples, we can always obtain precise estimation results. As shown in Fig. 3.14 (a), the PDF curve estimated using 100 samples still fits to theoretical Rayleigh distribution, whereas the Monte-Carlo-based histogram is very "confused".

### 3.3.2 CDMA system

In this section, we shall consider a synchronous CDMA system with $K$ users employing normalized spreading codes, using BPSK over an AWGN channel.

At each instant $i$, the received signal vector is given by

$$\boldsymbol{r}_i = \sum_{k=1}^{K} A_k b_i^{(k)} \boldsymbol{s}_k + \boldsymbol{n}_i \tag{3.46}$$

where :

- $\boldsymbol{s}_k$ : spreading code corresponding to user $k$, $\boldsymbol{s}_k \in \{\pm 1/\sqrt{L_{SF}}\}^{L_{SF}}$. In the following, we use the two spreading codes :

$$\begin{cases} \boldsymbol{s}_1 = \frac{1}{\sqrt{7}}[+1,+1,+1,+1,-1,-1,-1]^T \\ \boldsymbol{s}_2 = \frac{1}{\sqrt{7}}[-1,-1,+1,+1,-1,-1,-1]^T \end{cases} \tag{3.47}$$

- $L_{SF}$ : spreading factor ;
- $b_i^{(k)}$ : information bit of user $k$ at instant $i$, $b_i^{(k)} \in \{\pm 1\}$ ;
- $\boldsymbol{n}_i$ : temporally and spatially white Gaussian noise, i.e., $\boldsymbol{n}_i \sim N(\mu, \sigma^2 I_{L_{SF}})$ ;
- $A_k$ : signal amplitude of the users. In the following, we consider the case where the two users have equal power, $A_1 = A_2 = 1$.

In our works, we consider two types of CDMA receiver : the standard receiver and the decorrelator-based receiver.

#### 3.3.2.1 Standard receiver

A sufficient method for demodulating the data bits from the received signal of the $K$ users is to utilize the filter matched to the given spreading codes $\boldsymbol{s}_k$. The output of the filter for the $k^{th}$ user is given by :

$$X_i^{(k)} = \boldsymbol{s}_k^T \boldsymbol{r}_i \qquad k = 1, \dots K \tag{3.48}$$

Fig. 3.15 shows the schema bloc of standard receiver for 2 users ($K = 2$).

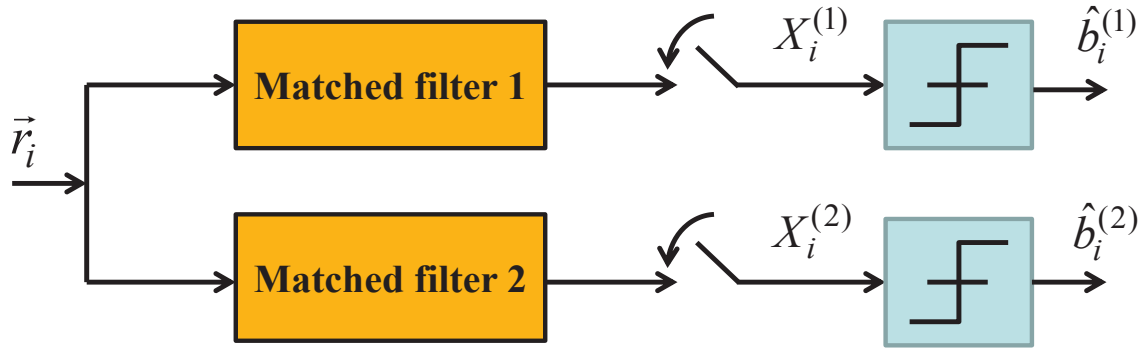


***Figure 3.15*** — Standard receiver

The detection of the $k^{th}$ user at time instant $i$ is given by :

$$\begin{cases} X_i^{(1)} = A_1 b_i^{(1)} + A_2 b_i^{(2)} \rho + n_i^{(1)} \\ X_i^{(2)} = A_2 b_i^{(2)} + A_1 b_i^{(1)} \rho + n_i^{(2)} \end{cases} \tag{3.49}$$

where :

- $\rho$ : normalized cross-correlation between the two spreading codes. For the spreading codes given by Eq. 3.47, $\rho = 0.4286$.
- $n_i^{(k)}$ : Gaussian noise at the output of the detector of $k^{th}$ user, i.e., $n_i^{(k)} \sim N(0, \sigma^2)$.

The decision about information bit $b_i^{(k)}$ is performed by computing the sign of decision statistic.

$$\hat{b}_i^{(k)} = sign(X_i^{(k)}) \qquad k = 1, 2 \tag{3.50}$$

Assume that the a priori probabilities of transmitted bits are identical for both users.

$$\begin{cases} \pi_+ = P[b_i^{(k)} = +1] \\ \pi_- = P[b_i^{(k)} = -1] \end{cases} \qquad \forall\, k, i \tag{3.51}$$

The theoretical BER for user 1 is given by :

$$p_{e1,\text{theoretical}} = 2\pi_+\pi_- Q\Big(\frac{A_1 - A_2\rho}{\sigma}\Big) + (\pi_+^2 + \pi_-^2) Q\Big(\frac{A_1 + A_2\rho}{\sigma}\Big) \tag{3.52}$$

where $Q(\cdot)$ is given by Eq. 2.28.

Fig. 3.16 shows the simulation results of BER estimation for the CDMA system employing standard receiver. The number of soft outputs that serve for estimating the BER is 1000. We consider two cases : single user and 2 users.

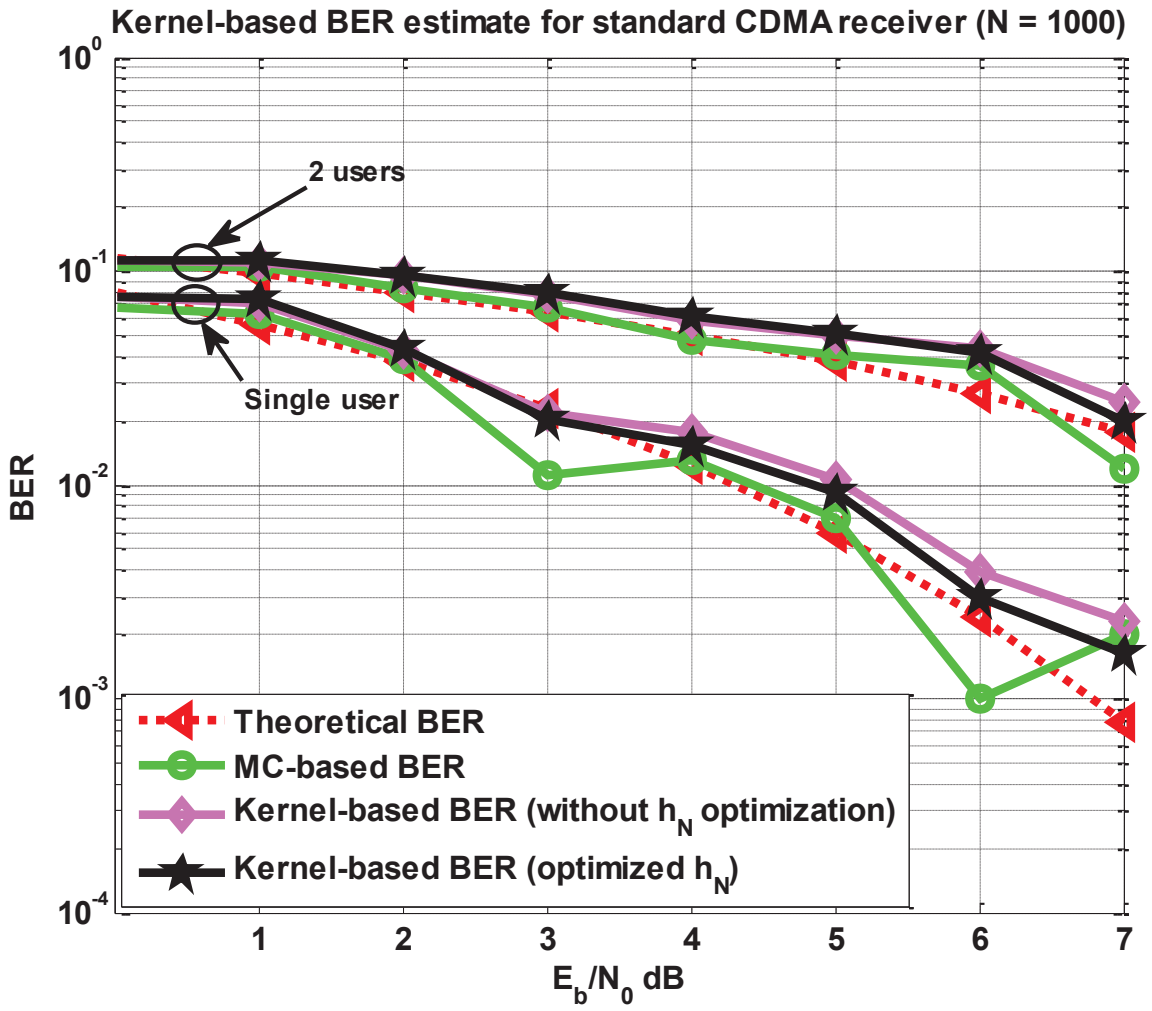


***Figure 3.16*** — Kernel-based BER estimation for standard CDMA receiver ($N = 1000$)

The Kernel method gives reliable BER estimation for given values of SNR. Compared with Monte-Carlo simulation, the Kernel-based results are not always very close to the theoretical values given by Eq. 3.51, e.g., it can be shown that for $SNR = 6\ dB$, the BER estimated by using Monte-Carlo method is closer to the theoretical value compared with the one estimated by Kernel method.

We can also find the dependence of Kernel-based BER estimate on Monte-Carlo simulation. However, for single trial, the BER estimate curve based on Kernel method, especially the one using optimized smooth parameters, is "smoother" than the one based on Monte-Carlo simulation.

Fig. 3.17 illustrates the histograms and the Kernel-based PDF estimates when $SNR = 6\ dB$ and $10\ dB$.

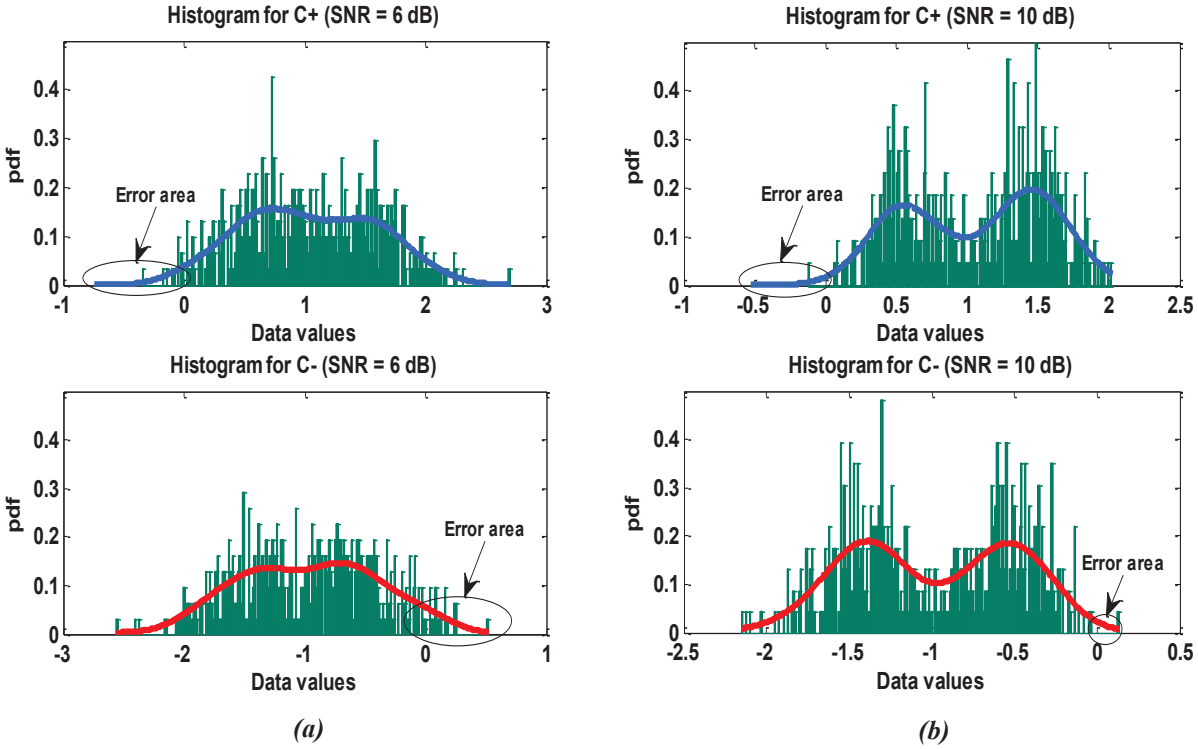


***Figure 3.17*** — Histograms and Kernel-based PDF estimates for standard CDMA receiver when (a). $SNR = 6; dB$, (b). $SNR = 10\ dB$ and $N = 1000$

While increasing the value of SNR, the error areas for $C_+$ and $C_-$ become smaller as errors are not normally seen. For this reason, if the simulation is performed several trials, we may find that the Kernel method is unreliable for high SNR since the estimated PDF functions in the error areas may be "unstable" and may present quasi-significant difference between the results of different trials. However, the "average" PDF and BER estimates could be closer to the theoretical one. Thus, if it is not required to estimate the BER in a real-time fashion, performing several trials of Kernel simulation then computing the mean BER can improve the BER estimates.

### 3.3.2.2 Decorrelator-based receiver

Adding a decorrelator after the filter allows improving the receiver performance.

The device $R^{-1}$ shown in Fig. 3.18 represents the inverse correlation matrix of the two spreading codes given by Eq. 3.47. In our case, $R^{-1}$ is given by :

$$R^{-1} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$$

The input of the decision logic for user 1 is given by :

$$X_i^{(1)} = R^{-1}(1,1)y_i^{(1)} + R^{-1}(1,2)y_i^{(2)} \tag{3.53}$$

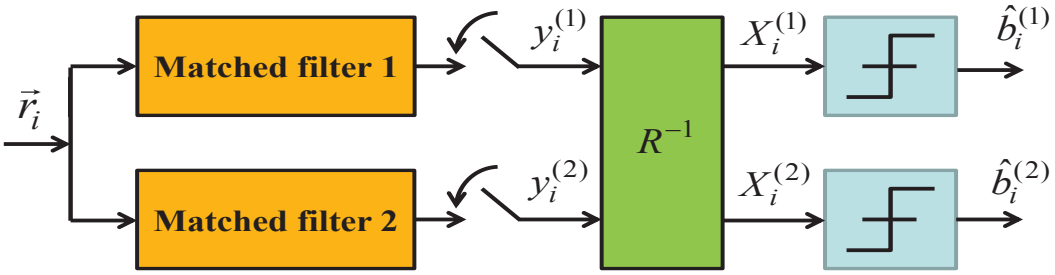


***Figure 3.18*** — Decorrelator-based receiver

The theoretical BER for user 1 can be expressed as :

$$p_{e1,\text{theoretical}} = Q\left(\frac{A_1\sqrt{1-\rho^2}}{\sigma}\right) \tag{3.54}$$

Obviously, this value does not depend on the bits transmitted by user 2.

FIG 3.19 shows the Kernel-based BER estimates for the decorrelator-based receiver while taking 2000 data samples. We have utilized the Newton's method to optimize the smooth parameters.

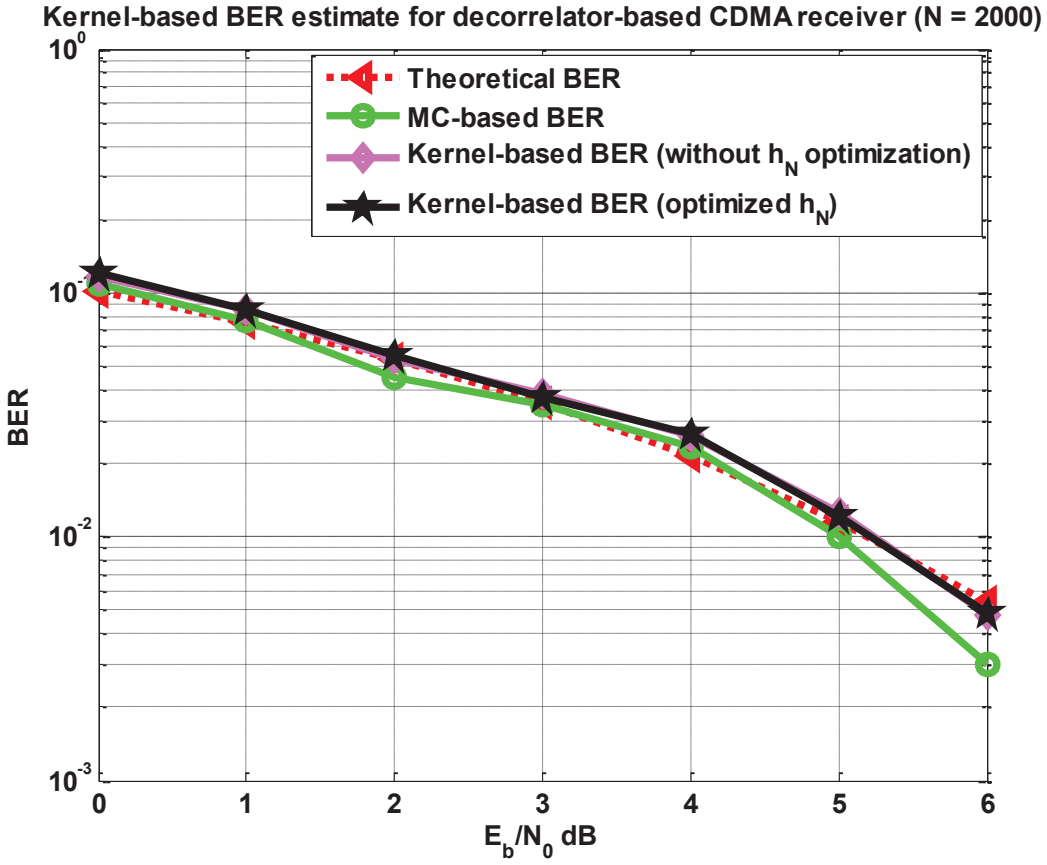


***Figure 3.19*** — Kernel-based BER estimation for decorrelator-based CDMA receiver ($N = 2000$)

It can be shown that :
- for low SNR, the BER performances of different estimation methods are quite similar ;
- for high SNR, the proposed technique based on Kernel method provides reliable BER estimates, with respect to the theoretical curve (obtained by using Eq. 3.54), while the performance of Monte-Carlo method turns to be random because of the very limited number of transmitted data bits.

Moreover, compared with the previous simulation results shown in Fig. 3.16, we can observed a very small "bias" and a clear improvement in "stability" and "smoothness" : the BER curves estimated by using the three methods no longer present dramatic random fluctuations, since the number of data samples is increased.

However, 2000 samples are not sufficient while taking very high SNR, even for the Kernel method. If the SNR gets higher, the Kernel method cannot provide BER estimation in good precision while taking 2000 samples, e.g., when $SNR = 20\ dB$, we have tested the Kernel-based BER estimation for several trials and we found that there are always vast and random fluctuations which are quite similar to the results shown in Fig 3.11.

In Tab. 3.4, three BER values obtained by taking 100 trials and using the corresponding initial and optimal values of the smoothing parameter $h_{N_+}$ are demonstrated for SNR from 1 $dB$ to 6 $dB$. Theoretical BER and estimated BER values by using initial and optimal smoothing parameters (optimized by using Newton's method) were reported. The initial smoothing parameters are given by Eq. 3.27.

| SNR | 1 $dB$ | 2 $dB$ | 3 $dB$ | 4 $dB$ | 5 $dB$ | 6 $dB$ |
|---|---|---|---|---|---|---|
| Theoretical BER | 0.0758 | 0.0539 | 0.0355 | 0.0214 | 0.0115 | 0.0054 |
| Mean BER (MC) | 0.0800 | 0.0549 | 0.0389 | 0.0251 | 0.0120 | 0.0055 |
| Mean BER ($h_{N,\text{initial}}$) | 0.0864 | 0.0605 | 0.0433 | 0.0305 | 0.0160 | 0.0077 |
| Mean BER ($h_{N,\text{optimal}}$) | 0.0821 | 0.0582 | 0.0429 | 0.0300 | 0.0139 | 0.0061 |

**Table *3.4*** — Comparison of theoretical BER and estimated BERs by using optimal and initial values of $h_N$ in the CDMA system

Firstly, we can find that the mean of BER estimation using Monte-Carlo method is quite close to the theoretical BER. This is in line with the theory, in fact, the main advantage of Monte-Carlo simulation is to lead very small bias when there is sufficient trials and data samples.

Secondly, by comprehensive comparison, we arrive at a conclusion that the estimated BER with optimal $h_N$ is closer to the theoretical values than the one with initial $h_N$. This conclusion accords with the simulation results shown in Fig 3.16 : the PDF curves estimated by using optimal smooth parameters could be "smoother" than the one without performing optimization, even if AWGN channel is used (note that Eq. 3.14 gives an approximation of the IMSE). This can be explained by considering the basic theory of bandwidth selection : optimal bandwidth should help PDF estimate best fitted to the theory curve.

Tab. 3.5 shows the corresponding initial and optimal values of the smoothing parameter $h_{N_+}$ for SNR from 1 $dB$ to 6 $dB$. It is shown that the difference between the initial and optimal values changes with SNR. To ensure the reliability of Newton's method, this difference must be much smaller than a given threshold, as presented in section 3.2.2.2.

| SNR | 1 $dB$ | 2 $dB$ | 3 $dB$ | 4 $dB$ | 5 $dB$ | 6 $dB$ |
|---|---|---|---|---|---|---|
| Mean $h_{N,\text{initial}}$ | 0.1545 | 0.1322 | 0.1313 | 0.1274 | 0.1165 | 0.1103 |
| Mean $h_{N,\text{optimal}}$ | 0.1867 | 0.1506 | 0.1354 | 0.1306 | 0.1298 | 0.1140 |

**Table 3.5** — Initial and optimal smoothing parameter $h_{N_+}$ for different SNR in the CDMA system

Furthermore, we also have computed the variance of BER estimation with the three methods (Monte-Carlo simulation, Kernel method without bandwidth optimization and Kernel method using optimal bandwidth) when $N = 2000$ for different SNR. The variance of the BER estimation is computed by performing 1000 trials.

| | | SNR = 0 $dB$ | SNR = 4 $dB$ | SNR = 6 $dB$ |
|---|---|---|---|---|
| Monte-Carlo | Variance | $5.0 \cdot 10^{-5}$ | $2.4 \cdot 10^{-5}$ | $1.3 \cdot 10^{-5}$ |
| Kernel using intial $h_N$ | Variance | $4.5 \cdot 10^{-5}$ | $2.1 \cdot 10^{-5}$ | $1.1 \cdot 10^{-6}$ |
| Kernel using optimal $h_N$ | Variance | $4.3 \cdot 10^{-5}$ | $2.0 \cdot 10^{-5}$ | $1.0 \cdot 10^{-6}$ |

**Table 3.6** — Variance of BER estimation for Monte-Carlo and Kernel methods for different SNR using 2000 data samples and 1000 trials

In the sense of the minimum variance (or standard deviation) of BER estimation, Kernel method using optimal bandwidth provides best performance. Without optimizing smooth parameters, the Kernel method leads to greater bias. The mean of the BER estimation of Monte-Carlo method could be very close to the theoretical value while taking a large number of trials, but the bias is the maximum compared with Kernel method.

In conclusion, the Kernel method is more reliable than the Monte-Carlo method. Specially, single trial of Kernel-based simulation can provide good performance, this presents a major strength of the proposed Kernel-based BER estimator that the BER could be estimated in a real-time fashion based on very limited number of data samples.

### 3.3.3 Turbo coding system

The $3^{rd}$ framework is a digital system using turbo codes with 1/3 coding rate over an AWGN channel. Principle and performance assessment of the turbo codes can be found in [BGT93, BG96]. The Turbo coder/decoder codes are given by M. Mohamed ET-TOLBA.

Fig. 3.20 shows the simulation results of BER estimation in the case of turbo code. 8 iterations for Max-Log-MAP (Max-Log-Maximum A Posteriori) decoding algorithm have been used to compute the soft observed LLRs. For MC simulation, we have taken 600 frames and 2000 frames, each frame contains 500 data bits.

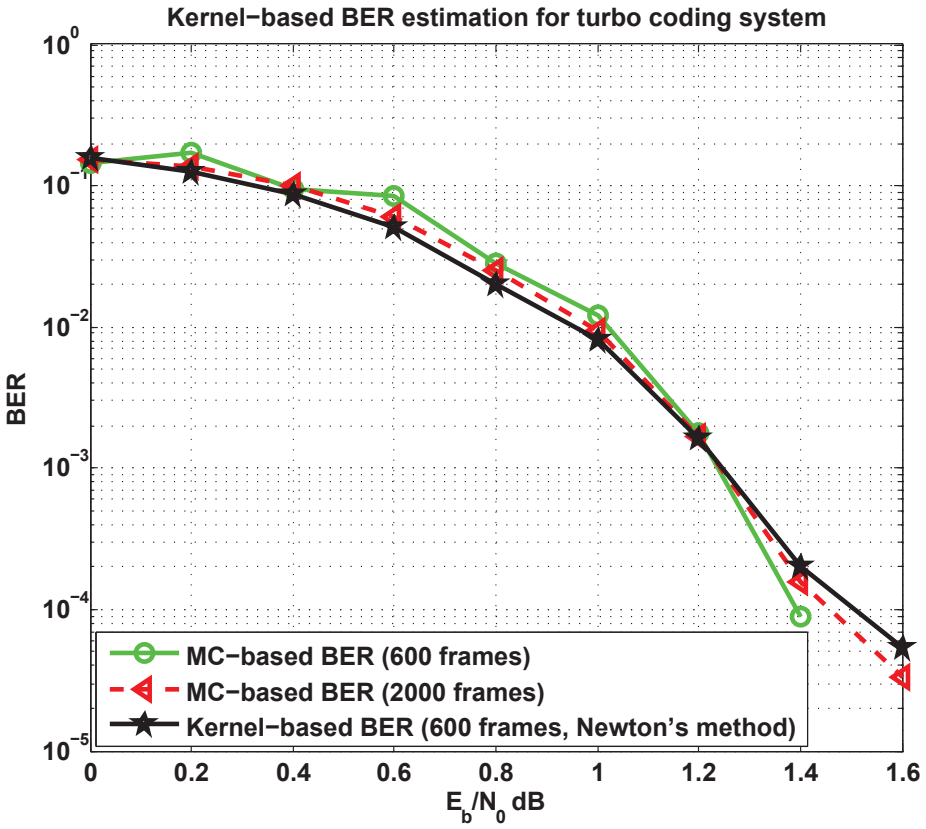


***Figure 3.20*** — Kernel-based BER estimation (300,000 samples) and MC-based BER (300,000 and 1,000,000 samples) for turbo coding system

The Kernel method provides better performance than Monte-Carlo method in single trial. For our simulation, when $SNR = 1.4\ dB$, only 26 errors are counted for the Monte-Carlo simulation. When $SNR = 1.6\ dB$, the Monte-Carlo method fails to obtain BER estimation since there is no errors. Without using the proposed smoothing parameters optimization methods, the Kernel-based BER estimation also provides good performance which is quite similar to the one using optimal $h_N$.

In general, the Kernel-based BER estimation is much more reliable since the BER curve seems quite "smooth", this accords with the regular rule in the sense of the minimum of bias, as presented in section 3.2.2.2.

Fig. 3.21 shows the statistical results of length of database for turbo code-based system at different values of BER estimate. At least 100 errors are counted for Monte-Carlo simulation. It can be shown that the Kernel method needs much less number of data points than the Monte-Carlo simulation. To obtain a BER equals to $10^{-5}$, the Monte-Carlo simulation needs at least $10^7$ samples, whereas the Kernel method requires $5.0 \cdot 10^5$ samples. If each frame contains 500 data bits, only 1000 frames can be used to have a similar precision.

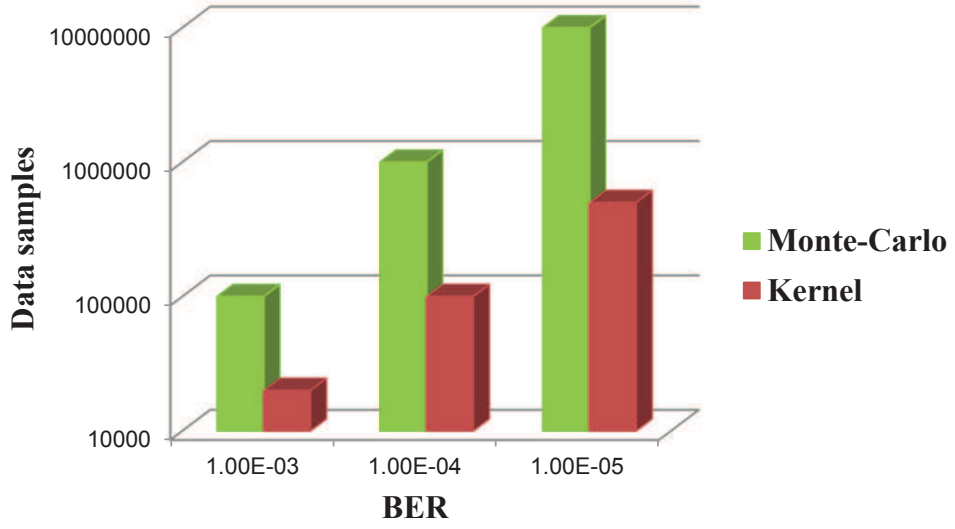


***Figure 3.21*** — Length of database at different values of BER for turbo coding system

Therefore, we demonstrate that the proposed Kernel-based technique provides similar BER results as Monte-Carlo and only requires a small number of soft outputs compared with the classical Monte-Carlo simulation.

This advantage in reducing the number of data samples can be explained by considering the previous PDF estimation results (cf. Fig. 3.14) : the Kernel-based PDF estimation can still be sufficiently "smooth" even if few data samples are used.

### 3.3.4 LDPC coding system

We have also tested the Kernel method for a 1/2 rate Quasi-Cyclic-Low Density Parity Check (QC-LDPC) system. Principle and implementation of the QC-LDPC code are presented in Appendix A.

Instead of computing the LLR, we use the pseudo a posteriori probabilities $Q_j$ (cf. Appendix A (3)) as the soft observations :

$$\text{soft observation} = Q_j^1(T) - Q_j^0(T) \tag{3.55}$$

where $Q_j^i(T)$ denotes the pseudo a posteriori probability that the $j^{th}$ transmitted bit $b_j = i$ at $T^{th}$ iteration of decoding algorithm.

Obviously, we have : $\text{soft observation}_{LDPC} \in [-1, +1]$. The decision is given by :

$$I(b_i) = \begin{cases} 1 & \text{if} \quad Q_j^1 - Q_j^0 > 0, \\ 0 & \text{otherwise.} \end{cases} \tag{3.56}$$

Fig. 3.22 shows the simulation results of BER estimation in the case of QC-LDPC code. The parity check matrix $G$ is generated with a dimension of $635 \times 1270$. We have used 500 frames and each one contains 635 random bits. For MC simulation, 500 frames and 15 000 frames are used respectively.

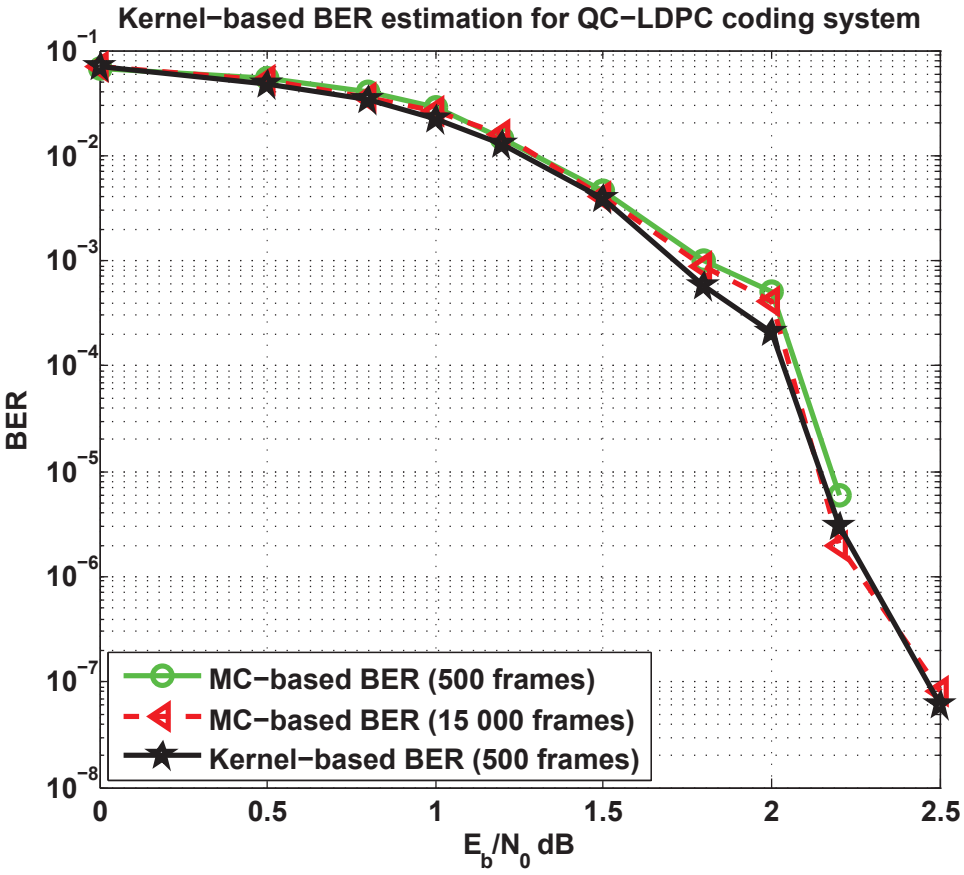


***Figure 3.22*** — Kernel-based BER estimation for QC-LDPC system with 500 frames

As the previous results for other frameworks, we can always find the dependence of the Kernel-based BER estimation on the Monte-Carlo simulation results (e.g., the BER estimates from $SNR = 1.5\ dB$ to $SNR = 2\ dB$). However, the Monte-Carlo-based BER curve stops at $SNR = 2\ dB$, whereas the Kernel method still works for high values of SNR. If we want to achieve low BER (e.g., $< 10^{-5}$) by using Monte-Carlo method, a very big number of data samples must be used. In our simulation, $500 \times 635 = 317{,}500$ samples are not sufficient since only single-digit errors could be found for $BER = 10^{-5}$. While using 15 000 frames, we can obtain the BER estimates down to $10^{-7}$ for MC simulation, whereas 500 frames are sufficient for the Kernel method to have a good precision.

However, this may not be very promising for many other practical systems where it is required to obtain a real-time estimation of the BER, since the bandwidth optimization algorithm will take excessively long computing time. In order to reduce the simulation complexity, we propose to use a modified dataset which only contains a part of samples. Let us consider the obtained histogram for $C_+$ when $SNR = 2.5\ dB$.

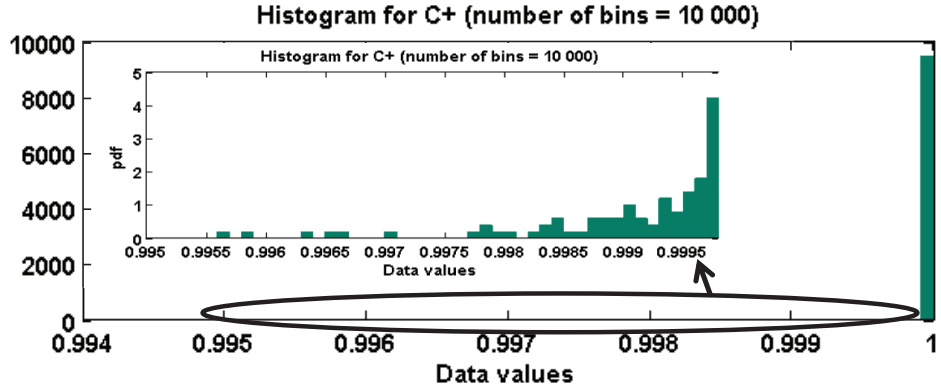


***Figure 3.23*** — Histogram of the soft output ($C_+$) for QC-LDPC system when $SNR = 2.2\ dB$

It can be shown that the soft output is extremely centered at +1 with very small bias. The zoomed-in figure shows the histogram from 0.995 to 0.9995, in this area, only few samples are found.

Tab. 3.7 reports the statistic of the values of soft outputs, $X^{b+}$ and $X^{b-}$, when $SNR = 2.5\ dB$. Recall that there are totally 317,500 outputs.

| | min value | max value | total number | number : $\lvert x^{b\pm}\rvert = 1$ | number : $\lvert x^{b\pm}\rvert < 1$ |
|---|---|---|---|---|---|
| $X^{b+}$ | 0.4061 | +1 | 158,512 | 121,049 | 37,463 |
| $X^{b-}$ | -1 | -0.6888 | 158,988 | 121,387 | 37,601 |

**Table *3.7*** — Statistic of soft data values when $SNR = 2.5\ dB$

Since the BER is determined by the "error area" which represents the tails of the conditional PDFs, we can get rid of the soft observations that equal to ±1. As shown in Tab. 3.7, for $C_+$, only 37,463 soft outputs will be taken into account, which is about 4 times less than the original size of dataset.

Furthermore, we can even reduce the size of dataset by setting a threshold $\epsilon$. As an example, when $SNR = 2.5\ dB$, $\epsilon = 0.9995$ means that only the soft outputs that belong to [-0.9995, 0.9995] need to be taken into account. For the above simulation result shown in Fig. 3.23, only 1368 soft outputs shall be used. However, the threshold $\epsilon$ must be selected very carefully. For high SNR, $\epsilon$ cannot be too small in order to avoid losing too much information. Empirically, we propose to use a threshold greater than or equal to 0.9995.

Last, we shall test the performance of the proposed modified Kernel-based method in the sense of mean and variance. Fig. 3.24 shows the maximum, minimum and mean values of Kernel-based BER estimates for QC-LDPC system using 500 frames and 20 trials.

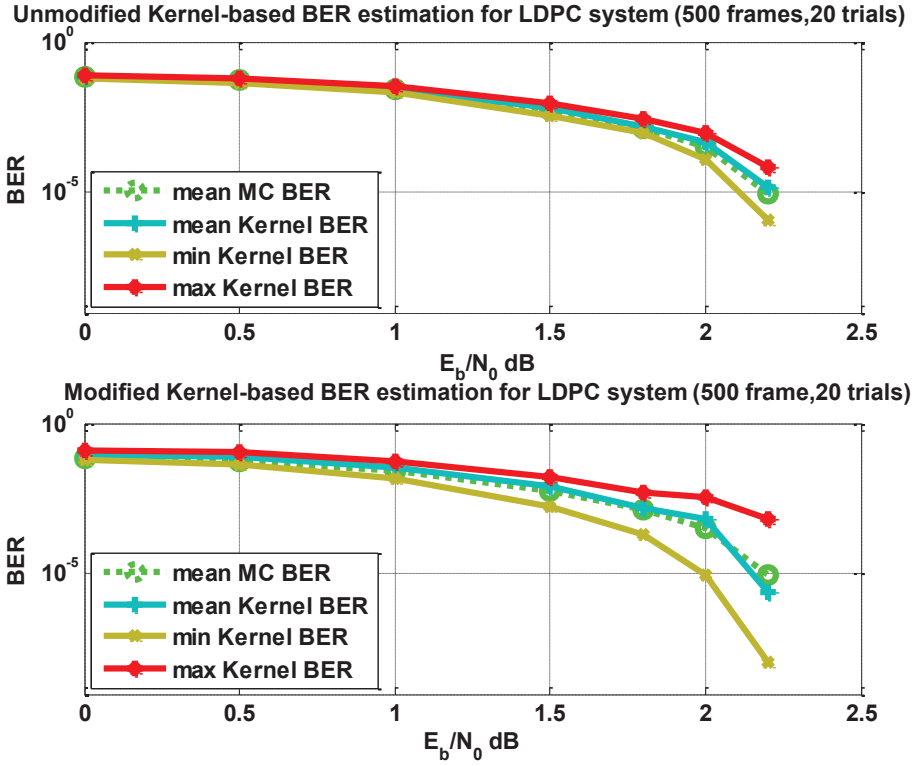


***Figure 3.24*** — Kernel-based BER estimates comparison for QC-LDPC system (500 frames, 20 trials) : top : using unmodified dataset ; bottom : using modified dataset (threshold = 0.9996)

It can be shown that the mean values of BER estimates are very close to the true values. However, compared with the unmodified version, the proposed modified method has worse precision in the sense of the minimum variance : the BER estimates obtained by using modified dataset could be randomly variant in the region of high SNR, since we only take in account a few number of observations corresponding to the tails of PDFs.

## 3.4 Conclusion

In this chapter, we have firstly presented the properties of Kernel estimator. Then, for Gaussian Kernel and AWGN channel, we have given the optimal smoothing parameters by minimizing the IMSE. Next, for practical situation, we have proposed two algorithms to iteratively estimate the optimal bandwidths. Once we obtain the Kernel estimators for both classes $C_+$ and $C_-$, the BER can be computed analytically. Last, we have reported the behavior of the Kernel-based estimator for different digital communication systems.

In the context of BER estimation, the Kernel method consists in estimating the conditional PDFs and smooth the histogram. It was shown that, compared with the conventional Monte-Carlo technique, the Kernel-based estimator provides better precision even with very limited number of data samples.

Moreover, the soft observations can be any kind of soft information, e.g., for Turbo coding system, the LLRs are used, whereas for LDPC coding system, we have applied the pseudo a posteriori probabilities. This makes the Kernel estimator flexible to fit the receiver's schema and decoding algorithm.

For non-Gaussian distribution, the smoothing parameters given by Eq. 3.27 is no longer applicable. The two proposed methods, curve fitting method and Newton's method, should be used.

For LDPC coding system, we have proposed to modify the size of dataset to reduce the computing time. This method consists in getting rid of the soft outputs equal (or very close) to $\pm 1$ since these samples are useless for BER estimation. This method will also be used for BER estimation based on Gaussian Mixture Model.

CHAPTER 4

# Bit Error Rate estimation based on Gaussian Mixture Model

In Chapter 3, we have proposed a new technique based on non-parametric Kernel method to estimate Bit Error Rate. It was shown that the proposed method provides similar BER results as Monte-Carlo while it only requires few soft outputs.

As presented in Chapter 2, the mixture model is a major class of semi-parametric model. Compared with the proposed Kernel method based on non-parametric model, we can easily control the number of mixture components in order to reduce the number of parameters. In this chapter, instead of using the Kernel method, a semi-parametric Gaussian Mixture Model will be utilized.

## 4.1 Missing data of component assignment

In section 2.2.4.2, we have shown that the unknown parameters of Mixture Model cannot be obtained without finding the missing data $Z_i$ which determines the assignment of every data point to the different components.

As Mixture Models are widely used in the domain of data clustering, we will show how to perform the cluster assignment (the centroids of the $K$ clusters, $(\mu_k)_{1\leq k\leq K}$) by using the mixture model. First of all, we consider a simple case : the K-mean clustering.

### 4.1.1 K-means clustering

#### 4.1.1.1 Principle of K-means clustering

K-means clustering is the classical method of clustering analysis [M$^+$67,Har75]. Let us consider $N$ observations $X_i \in \{X_1, \ldots, X_N\}$, K-means clustering method aims to partition the $N$ observations into $K$ sets ($K \leq N$) so as to minimize the Within-Cluster Sum of Squares (WCSS).

$$\arg\min_{z_{ik}}(WCSS) = \arg\min_{z_{ik}}\Big(\sum_{i=1}^{N}\sum_{k=1}^{K} z_{ik}||X_i - \mu_k||^2\Big) \tag{4.1}$$

where $\mu_k$ is the mean of data points which represents the centroid of cluster ; $z_{ik}$ is the "beacon" which is equal to 1 if $X_i$ belongs to the $k^{th}$ cluster.

The centroid is also called the model parameter, often defined as $\theta = (\mu_k)_{1\leq k\leq K}$. The data samples $X_i$ are incomplete data because of missing the cluster assignment information. Thus, in clustering problem we need to find estimates for both $z_{ik}$ and $\mu_k$ that minimize the objective function (WCSS).

#### 4.1.1.2 K-means clustering algorithm : KMA

The most common method for K-means clustering is an iterative algorithm, called the Forgy's batch K-Means Algorithm (Forgy's batch KMA) [For65, ÄKM07]. The algorithm proceeds by alternating between two steps :

- **Assignment step** : *minimizing the WCSS with regard to* $z_{ik}$.
  Since all data samples are independent, we can choose $z_{ik}$ to be 1 for whichever value $k$ gives the minimum value of the squared distance. Then we can assign the current observation to the nearest cluster center.

$$z_{ik}^{(t)} = \big\{X_i : ||X_i - \mu_k^{(t)}|| \leq ||X_i - \mu_j^{(t)}||\big\} \qquad \forall j,\ 1 \leq k, j \leq K \tag{4.2}$$

- **Update step** : *minimizing the WCSS with regard to* $\mu_k$.
  We take the derivative of WCSS with regard to $\mu_k$ and equate to zero. This step is similar to the Maximization step of Expectation-Maximization algorithm for GMM. In this subsection, we just give the expression of updated $\mu_k$, the details of calculations and proofs will be shown in section 4.2.2.

$$\mu_k = \frac{\sum_{i=1}^{N} z_{ik}X_i}{\sum_{i=1}^{N} z_{ik}} \tag{4.3}$$

Another similar method for on-line clustering is called the sequential K-means algorithm. This method updates the centers whenever a data point $x_i$ is available, as follows :

- first, we initialize the centroids for $K$ clusters.
- second, find the cluster centroid that is nearest to the incoming data point. Add the data point to the cluster and update the cluster center as the mean vector of all the data points in this cluster.
- then check if the nearest centroid of a data point is the center this data point belongs to. Repeat to check all the data points.

The K-means procedure can be seen as an algorithm for partitioning the $N$ samples into $K$ clusters so as to minimize the sum of the squared distances to the cluster centers. However, K-means algorithm does have some weaknesses :

- we cannot specify the way to initialize the means. In general, we start with a random number $K$ as the number of Gaussians ;

– the final results produced depend on the initial values of the means ;
– sometimes it makes hard guess for cluster assignment. In other words, for some cases our model may not be sure about exact cluster assignment ;
– the final results produced also depend on the value of $K$.

Particularly, the last problem is troublesome, since we often have no idea how many clusters "should" exist. Furthermore, there is no theoretical solution to find the optimal number of clusters for any data samples. In general, we should compare the results of different trials with different values of $K$ and choose the best one.

There are some other popular algorithms of clustering, such as :
– Fuzzy C-means algorithm [BEF84, JSM97], which is the most used method based on overlapping clustering algorithm ;
– Hierarchical clustering algorithm [TJBB06].

In the following subsection, we will discuss an improved approach based on K-means algorithm : the model-based clustering method.

### 4.1.2 Probabilistic clustering as a mixture of models

The model-based clustering method consists in using certain models for clusters and attempting to optimize the fit between the data samples and the chosen model. Comparing with the K-means algorithm, the missing data of model-based method is not constant (1 or 0) but defined as the probability that the studied observation belongs to a cluster. For this reason, this method is also called the probabilistic clustering approach.

In practice, each cluster can be mathematically represented by a parametric distribution, like a Gaussian or a Poisson. Therefore, the entire dataset is modeled by a mixture of the chosen distributions. The most widely used model-based clustering method is the one based on learning a mixture of Gaussian distributions. Therefore, the probabilistic clustering based on Gaussian distributions is equivalent to the GMM-based PDF estimation.

Let $Z_i \in \{Z_1, \ldots, Z_N\}$ be the missing data that determines the component (cluster) from which the data samples originate. $Z_i = k$ means that the data sample $X_i$ belongs to the $k^{th}$ component of the Gaussian mixture.

Let us consider the conditional PDF given by Eq. 2.49, the a priori probability $\alpha_k$ represents the probability that $Z_i = k$ [JAN13], i.e.,

$$P(Z_i = k) = \alpha_k \qquad \alpha_k \geq 0 \; and \; \sum_{k=1}^{K} \alpha_k = 1 \tag{4.4}$$

Therefore, $Z_i$ can be considered as missing data that follows a multinomial distribution, i.e., $Z_i \sim Multinomial(\alpha)$. For a chosen value of $Z_i$, we have :

$$(X|Z_i = k) \sim N(\mu_k, \sigma_k^2) \qquad 1 \leq k \leq K$$

We can obtain the joint probability distribution :

$$P(X_i, Z_i) = P(X|Z_i = k)P(Z_i) \tag{4.5}$$

Let $\theta = (\alpha_k, \mu_k, \sigma_k^2)_{1 \leq k \leq K}$ be the unknown parameters that we need to estimate. The criterion we will use is the maximization of the joint likelihood of both observed data samples, $X$, and missing data, $Z$. The log-likelihood function is given by [DLR77] :

$$\begin{aligned} \log[L(X, Z; \theta)] &= \sum_{i=1}^{N} \log[P(X_i, Z_i; \theta)] \\ &= \sum_{i=1}^{N} \log \sum_{k} P(X_i, Z_i = k; \theta) = \sum_{i=1}^{N} \log \sum_{k=1}^{K} P(X_i|Z_i = k; \theta)P(Z_i = k; \theta) \end{aligned} \tag{4.6}$$

Unfortunately, the maximum value of Eq. 4.6 cannot be computed by using the analytical derivative of the log-likelihood function, since we cannot obtain the close form solution.

Let us consider the following simple case : assume that we know the values of $Z_i$ for every data samples, we get :

$$\log[L(X, Z; \theta)] = \sum_{i=1}^{N} [\log P(X_i|Z_i = k; \theta) + \log P(Z_i = k; \theta)] \tag{4.7}$$

Take the derivatives with respect to $\{\alpha_k, \mu_k, \sigma_k^2\}$ :

$$\begin{aligned} \mu_k &= \frac{\sum_{i=1}^{N} P(Z_i = k)X_i}{\sum_{i=1}^{N} P(Z_i = k)} \\ \sigma_k^2 &= \frac{\sum_{i=1}^{N} P(Z_i = k)(X_i - \mu_k)^2}{\sum_{i=1}^{N} P(Z_i = k)} \\ \alpha_k &= \frac{1}{N} \sum_{i=1}^{N} P(Z_i = k) \end{aligned} \tag{4.8}$$

This means that :
– $\alpha_k$ is the mean of a posteriori probabilities that $Z_i = k$ ;
– $\mu_k$ is the mean of $X_i$ with respect to $Z_i = k$ ;
– $\sigma_k^2$ is the variance of $X_i$ with respect to $Z_i = k$.

We have obtained the mathematical expression of the unknown parameters $\theta = (\alpha_k, \mu_k, \sigma_k^2)_{1 \leq k \leq K}$ by assuming that we know the values of $Z_i$. However, $Z_i \in \{Z_1, \ldots, Z_N\}$ is the missing data that we need to find. In the next paragraph, we will discuss the way in which the missing data can be estimated and the algorithm of PDF/BER estimation based on GMM.

## 4.2 BER estimation based on Gaussian Mixture Model

The most widely used approach to compute the missing data and estimate the PDF is the Expectation-Maximization (EM) algorithm [Moo96, McL80], which is quite similar to the K-means algorithm. The EM algorithm also estimates the unknown parameters in an iterative way :

- **Estimation step** : we compute the missing data. This step corresponds to the Assignment step of the K-means algorithm (cf. section 4.1.1.2) ;
- **Maximization step** : we compute the new parameters by maximizing the log-likelihood function of data samples and missing data. This step corresponds to the Update step of the K-means algorithm.

First, we will give a brief presentation of the EM algorithm. Then we will discuss how to use the EM algorithm to estimate the unknown parameters of GMM. Finally, we will present how to compute the BER by using the PDF estimates.

### 4.2.1 Introduction to Expectation-Maximization algorithm

In section 4.1.2, we have presented that it is hard to directly estimate the unknown parameter $\theta$ from the log-likelihood function given by Eq. 4.6, since there is a missing data $Z_i$. That's why we need to maximize the log-likelihood function in an iterative way :

- in the **Estimation step**, we will try to find the "lower bound" of the log-likelihood function ;
- in the **Maximization step**, we will optimize this "lower bound".

Before discussing the details of EM algorithm, it is necessary to introduce the Jensen's inequality.

#### 4.2.1.1 Jensen's inequality

Let f be a function in real number field, for a given real variable $X$, if the $2^{nd}$ derivative $f^{''}(X) \geq 0$, $f(\cdot)$ is called convex function, we get :

$$E[f(X)] \geq f(E[X]) \tag{4.9}$$

Eq. 4.9 is called the Jensen's inequality.

If $f^{''}(X) \leq 0$, $f(\cdot)$ is called concave function. Then we get :

$$E[f(X)] \leq f(E[X]) \tag{4.10}$$

#### 4.2.1.2 Principle of Expectation-Maximization algorithm

We consider the same example discussed in section 4.1.2. Let $\{X_i, Z_i\}_{i=1,\dots,N}$ be the complete data which represents the observations along with the missing data. The log-likelihood function is given by Eq. 4.6. Let $\rho_{k,i}$ be a certain kind of distribution :

$$\sum_k \rho_{k,i}(Z_i) = 1 \qquad \rho_{k,i}(Z_i) \geq 0,\ k = 1, 2, 3 \tag{4.11}$$

Then Eq. 4.6 can be rewritten as :

$$\begin{aligned} \log[L(X, Z; \theta)] &= \sum_{i=1}^{N} \log \sum_k P(X_i, Z_i = k; \theta) \\ &= \sum_{i=1}^{N} \log \sum_k \rho_{k,i}(Z_i) \frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}(Z_i)} \end{aligned} \tag{4.12}$$

Assume that the distribution of random variable $X$ is a discrete distribution and we know the probability mass function $f_X$, then the expected value of a function $Y = g(X)$ of the variable $X$ is :

$$E[g(X)] = \sum_x g(x) f_X(x) \tag{4.13}$$

The above equation is referred to as the rule of the lazy statistician, which can be applied in Eq. 4.12 :

- $Z_i$ corresponds to the discrete random variable $X$ ;
- $\rho_{k,i}$ corresponds to the probability $f_X$, i.e., $f_X = \rho_{k,i}(Z_i)$ ;
- $P(X_i, Z_i = k; \theta)/\rho_{k,i}(Z_i)$ corresponds to $Y$ ;
- $g(X)$ is the mapping from $Z_i$ to $P(X_i, Z_i = k; \theta)/\rho_{k,i}(Z_i)$.

By using Eq. 4.13, we can write :

$$E\Big[\frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}(Z_i)}\Big] = \sum_k \rho_{k,i}(Z_i) \frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}(Z_i)} \tag{4.14}$$

According to the Jensen's inequality given by Eq. 4.10, consider that $\log(\cdot)$ is a concave function, we get :

$$\log\left(E\Big[\frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}(Z_i)}\Big]\right) \geq E\left[\log\Big(\frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}(Z_i)}\Big)\right] \tag{4.15}$$

By using Eq. 4.14 and Eq. 4.15, Eq. 4.12 can be rewritten as :

$$\begin{aligned} \log[L(X, Z; \theta)] &= \sum_{i=1}^{N} \log \sum_k \rho_{k,i}(Z_i) \frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}(Z_i)} \\ &\geq \sum_{i=1}^{N} \sum_k \rho_{k,i}(Z_i) \log\Big(\frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}(Z_i)}\Big) \end{aligned} \tag{4.16}$$

The "lower bound" of the log-likelihood function is then given by Eq. 4.16. Assume that we know the value of parameter $\theta$, then the probabilities $P(X_i, Z_i = k; \theta)$ and $\rho_{k,i}(Z_i)$ determinate the value of the log-likelihood function. We shall adjust these two probabilities to optimize the value of "lower bound" till the equality in Eq. 4.16 holds. The condition for equality is given by :

$$\frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}(Z_i)} = \text{constant} \tag{4.17}$$

Consider that $\sum_Z \rho_{k,i}(Z_i) = 1$ (cf. Eq. 4.11), we get :

$$\begin{aligned} \rho_{k,i}(Z_i) &= \frac{P(X_i, Z_i = k; \theta)}{\text{constant}} = \frac{P(X_i, Z_i = k; \theta)}{\sum_k P(X_i, Z_i = k; \theta)} \\ &= \frac{P(X_i, Z_i = k; \theta)}{P(X_i; \theta)} = P(Z_i = k | X_i; \theta) \end{aligned} \tag{4.18}$$

Eq. 4.18 shows that $\rho_{k,i}$ is seen as the a posteriori probability.

Therefore, the two steps of the Expectation-Maximization algorithm can be interpreted as :

– **Estimation step** :
at iteration $t$, we estimate the missing/hidden data $\rho_{k,i}$ for each data sample $X_i$ using the parameter value $\theta^{(t-1)}$ computed at the last Maximization step at the previous iteration $t-1$ (if $t = 1$, we use the initial value of the parameter, i.e. $\theta^{(0)}$) ;

$$\rho_{k,i}^{(t)}(Z_i) = P(Z_i = k | X_i = x_i; \theta^{(t-1)}) \qquad \text{for } i = 1, \dots N \text{ and } k = 1, \dots K \tag{4.19}$$

– **Maximization step** :
at the current iteration, we compute the unknown parameter $\theta^{(t)}$ by maximizing the log-likelihood function, assuming independent observation $X_i$.

$$\theta^{(t)} = \arg\max_{\theta} \left[ \sum_{i=1}^{N} \sum_{k} \rho_{k,i}^{(t)}(Z_i) \log\Big( \frac{P(X_i, Z_i = k; \theta^{(t-1)})}{\rho_{k,i}^{(t)}(Z_i)} \Big) \right] \tag{4.20}$$

The obtained parameter $\theta^{(t)}$ will be used at the next Estimation step at iteration $t+1$.

### 4.2.2 Expectation-Maximization algorithm for Gaussian Mixture Model

Now we shall back to the Gaussian Mixture Model. We will compute the unknown parameter $\theta = (\alpha_k, \mu_k, \sigma_k^2)_{1 \le k \le K}$ by using the Expectation-Maximization algorithm.

#### 4.2.2.1 Estimation step

We have given the expression of the a posteriori probabilities $\rho_{k,i}$ in section 4.2.1.2. (cf. Eq. 4.19). Then, by using simple Bayes' rule, at iteration $t$, we have :

$$\rho_{k,i}^{(t)} = P(Z_i = k|X_i = x_i; \theta^{(t-1)}) = \frac{P(X_i = x_i|Z_i = k; \theta^{(t-1)})P(Z_i = k; \theta^{(t-1)})}{P(X_i = x_i; \theta^{(t-1)})}$$

For $i = 1, \ldots, N$ and for $k = 1, \ldots, K$, we get :

$$\rho_{k,i}^{(t)} = \frac{\alpha_k^{(t-1)} f_k(X_i; \mu_k^{(t-1)}, \sigma_k^{(t-1)2})}{\sum_{k=1}^{K} \alpha_k^{(t-1)} f_k(X_i; \mu_k^{(t-1)}, \sigma_k^{(t-1)2})} \tag{4.21}$$

#### 4.2.2.2 Maximization step

Since the values of $\rho_{k,i}$ at iteration $t$ have been computed, we will maximize the joint log-likelihood function which can be rewritten as :

$$\begin{aligned}
L(\theta) &= \sum_{i=1}^{N} \sum_{k} \rho_{k,i}^{(t)}(Z_i) \log \frac{P(X_i, Z_i = k; \theta)}{\rho_{k,i}^{(t)}(Z_i)} \\
&= \sum_{i=1}^{N} \sum_{k=1}^{K} \rho_{k,i}^{(t)}(Z_i) \log \frac{P(X_i|Z_i = k; \theta)P(Z_i = k; \theta)}{\rho_{k,i}^{(t)}(Z_i)} \\
&= \sum_{i=1}^{N} \sum_{k=1}^{K} \rho_{k,i}^{(t)} \log \frac{\frac{\alpha_k^{(t)}}{\sqrt{2\pi}\sigma_k^{(t)}} \exp\left(-\frac{(X_i - \mu_k^{(t)})^2}{2\sigma_k^{(t)2}}\right)}{\rho_{k,i}^{(t)}} \\
&= \sum_{i=1}^{N} \sum_{k=1}^{K} \rho_{k,i} \left[\log \alpha_k^{(t)} - \frac{1}{2} \log(2\pi\sigma_k^{(t)2}) - \frac{(X_i - \mu_k^{(t)})^2}{2\sigma_k^{(t)2}}\right]
\end{aligned} \tag{4.22}$$

##### 4.2.2.2.1 Calculation of $\mu_k$

Setting the derivative with regard to $\mu_k$ of the joint log-likelihood function to zero, we find :

$$\frac{\partial(L(\theta))}{\partial(\mu_k)} = \sum_{i=1}^{N} \rho_{k,i}^{(t)} \frac{X_i - \mu_k^{(t)}}{\sigma_k^{(t)2}} = 0$$

Thus, for $k = 1, \ldots, K$, we get :

$$\mu_k^{(t)} = \frac{\sum_{i=1}^{N} \rho_{k,i}^{(t)} X_i}{\sum_{i=1}^{N} \rho_{k,i}^{(t)}} \tag{4.23}$$

It can be shown that Eq. 4.23 corresponds to the $1^{st}$ equation of Eq. 4.8.

#### 4.2.2.2.2 Calculation of $\sigma_k^2$

Setting the derivative of the joint log-likelihood function with regard to $\sigma_k^2$ to zero, we find :

$$\frac{\partial(L(\theta))}{\partial(\sigma_k^2)} = \frac{\partial\left(\sum_{i=1}^{N}\sum_{k=1}^{K}\rho_{k,i}^{(t)}\left(\frac{(X_i-\mu_k^{(t)})^2}{2\sigma_k^{(t)2}} - \frac{1}{2}\log(\sigma_k^{(t)2})\right)\right)}{\partial(\sigma_k^{(t)2})}$$
$$= \sum_{i=1}^{N}\rho_{k,i}^{(t)}\frac{\frac{(X_i-\mu_k^{(t)})^2}{\sigma_k^{(t)2}} - 1}{2\sigma_k^{(t)2}} = 0$$

Thus, for $k = 1, \ldots, K$, we get :

$$\sigma_k^{(t)2} = \frac{\sum_{i=1}^{N}\rho_{k,i}^{(t)}(X_i - \mu_k^{(2)})^2}{\sum_{i=1}^{N}\rho_{k,i}^{(t)}} \tag{4.24}$$

It can be shown that Eq. 4.24 corresponds to the $2^{nd}$ equation of Eq. 4.8.

#### 4.2.2.2.3 Calculation of $\alpha_k$

Taking into account the constraint $\sum_{k=1}^{K}\alpha_k = 1$, we shall firstly add a Lagrange Multiplier into Eq. 4.22, we have :

$$L_{lagrange}(\theta) = \sum_{i=1}^{N}\sum_{k=1}^{K}\rho_{k,i}^{(t)}\Big[\log\alpha_k^{(t)} - \frac{1}{2}\log(2\pi\sigma_k^{(t)2}) - \frac{(X_i-\mu_k^{(t)})^2}{2\sigma_k^{(t)2}}\Big] + \beta(\sum_{k=1}^{K}\alpha_k^{(t)} - 1) \tag{4.25}$$

Since $\alpha_k \geq 0$, setting the derivative of Eq. 4.25 with regard to $\alpha_k$ to zero, we find :

$$\frac{\partial(L(\theta))}{\partial(\alpha_k)} = \sum_{i=1}^{N}\frac{\rho_{k,i}^{(t)}}{\alpha_k^{(t)}} + \beta = 0$$

For $k = 1, \ldots, K$, we get :

$$\alpha_k^{(t)} = \frac{-\sum_{i=1}^{N}\rho_{k,i}^{(t)}}{\beta}$$

Invoking the constraint $\sum_{k=1}^{K}\alpha_k = 1$ and the fact that $\sum_{k=1}^{K}\rho_{k,i}^{(t)} = 1$, we have :

$$-\beta = \sum_{i=1}^{N}\sum_{k=1}^{K}\rho_{k,i}^{(t)} = \sum_{i=1}^{N}1 = N$$

Then, we have :

$$\alpha_k^{(t)} = \frac{\sum_{i=1}^{N}\rho_{k,i}^{(t)}}{N} \tag{4.26}$$

It can be shown that Eq. 4.26 corresponds to the $3^{rd}$ equation of Eq. 4.8.

Eq. 4.23, Eq. 4.24 and Eq. 4.26 give the values of unknown parameter $\theta^{(t)} = (\alpha_k^{(t)}, \mu_k^{(t)}, \sigma_k^{(t)2})_{1 \leq k \leq K}$ at iteration $t$. The obtained parameter $\theta^{(t)}$ will be used at the next Estimation step at iteration $t+1$.

### 4.2.3 Example of GMM-based PDF estimation using Expectation-Maximization algorithm

Consider three Gaussian distributions, $N_1(0,1)$, $N_2(-2,3)$ and $N_3(2,2)$. Assume that we have 6000 samples :

- Component 1 ($k=1$) : 2000 samples are the distribution $N_1(0,1)$ ;
- Component 2 ($k=2$) : 1000 samples are the distribution $N_2(-2,1/9)$ ;
- Component 3 ($k=3$) : 3000 samples are the distribution $N_3(2,1/4)$.

Fig. 4.1 shows the histogram of the total 6000 data samples.

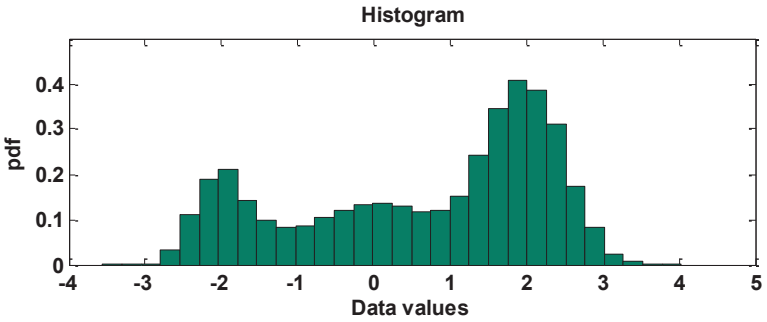


***Figure 4.1*** — Example of GMM : Histogram of 6000 data samples

With the 6000 samples, we will estimate the PDF based on Gaussian Mixture method by using the EM algorithm.

From Fig. 4.1, it is easy to guess that the initial number of Gaussians is 3 ($K=3$). The initial values of the unknown parameters, $\alpha_k^{(0)}$, $\mu_k^{(0)}$ and $\sigma_k^{(0)2}$, with $k=1,2,3$, are defined as :

- $\alpha_k^{(0)}$ are defined as $1/K$ ;
- $\mu_k^{(0)}$ are defined as the randomly selected data points from the 6000 samples ;
- $\sigma_k^{(0)2}$ are defined as the range of the input data.

$$\sigma_k^{(0)2} = \text{max value of data} - \text{min value of data}$$

Tab. 4.1 shows the estimated values of the unknown parameters, $\alpha_k$, $\mu_k$ and $\sigma_k$, $k=1,2,3$.

It can be shown that the obtained values of parameters are very close to the true values of parameters of the Gaussian PDFs.

| Parameter | $\alpha_k$ | $\mu_k$ | $\sigma_k^2$ |
|---|---|---|---|
| True value | $k=1:\alpha_1=1/3$<br>$k=2:\alpha_2=1/6$<br>$k=3:\alpha_3=1/2$ | $k=1:\mu_1=0$<br>$k=2:\mu_2=-2$<br>$k=3:\mu_3=2$ | $k=1:\sigma_1^2=1$<br>$k=2:\sigma_2^2=1/9$<br>$k=3:\sigma_3^2=1/4$ |
| Estimated value | $k=1:\alpha_1=0.3312$<br>$k=2:\alpha_2=0.1724$<br>$k=3:\alpha_3=0.4963$ | $k=1:\mu_1=0.0434$<br>$k=2:\mu_2=-1.9875$<br>$k=3:\mu_3=2.0052$ | $k=1:\sigma_1^2=0.9648$<br>$k=2:\sigma_2^2=0.1082$<br>$k=3:\sigma_3^2=0.2531$ |

**TABLE 4.1** — Estimated values of unknown parameters $\{\alpha_k, \mu_k, \sigma_k^2\}$

Fig. 4.2 shows the Gaussian components probabilities and the obtained PDF.

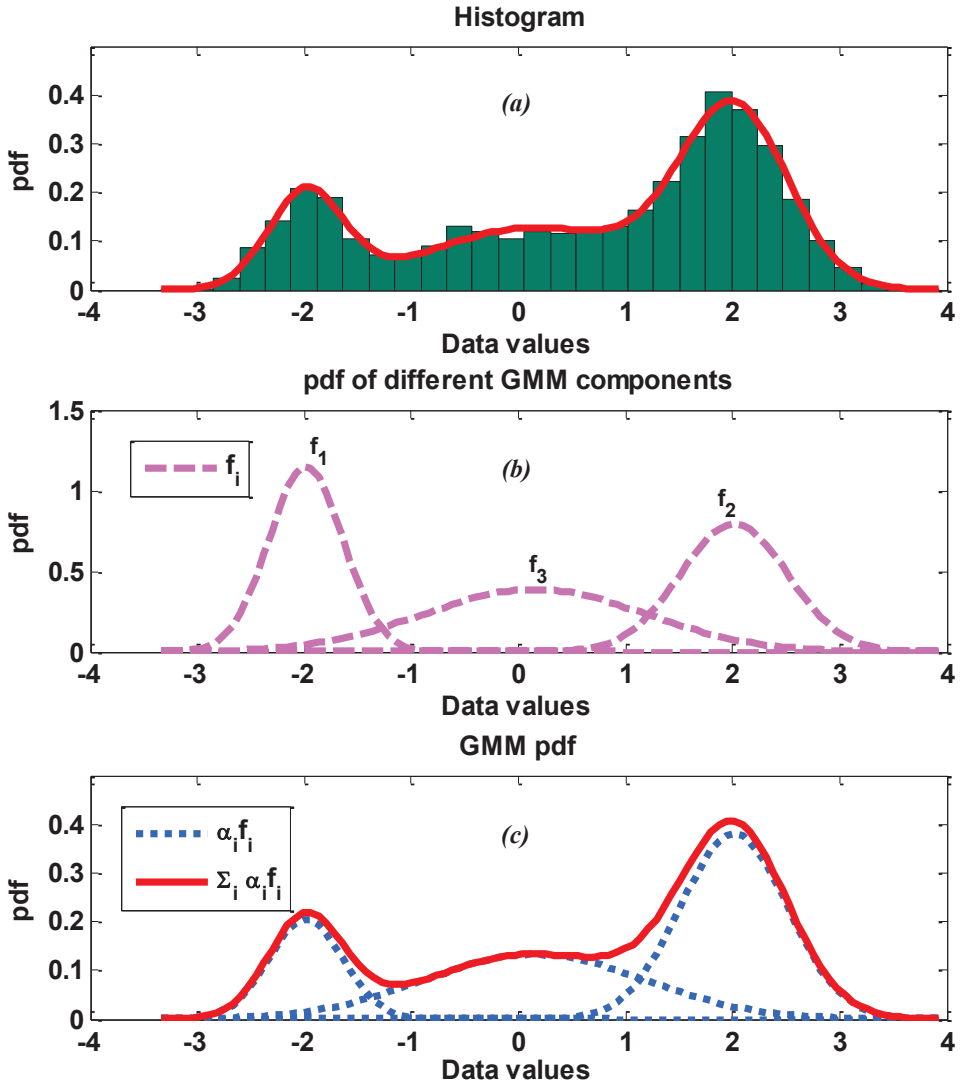


***Figure 4.2*** — PDF estimates based on GMM with 6000 samples : (a). Histogram compared with the final result of estimated PDF; (b). PDF of the three Gaussian components; (c). Final result of estimated PDF

The red curve represents the final result of the estimated PDF. From Fig. 4.2 (a), it can be shown that it fits very well to the histogram of data points.

Fig. 4.3 shows the curve of joint log-likelihood probability computed by (Eq. 4.22) with respect to the number of iterations of EM algorithm.

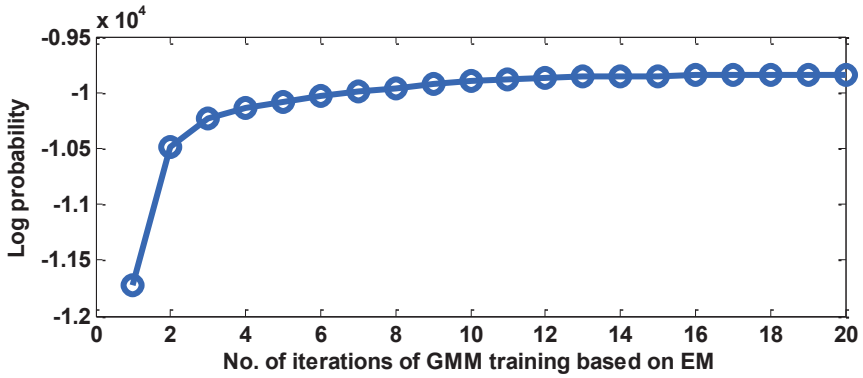


***Figure 4.3*** — GMM example : Log-likelihood probability versus iteration number of EM algorithm

It can be shown that the log-likelihood probability, given by (Eq. 4.22), is monotonically non-decreasing throughout the training iteration number.

From the above example, the identified Gaussian Mixture Model PDF can match the data histogram closely. However, this is based on the fact that we are able to guess the number of Gaussians correctly. In practice, this condition does not always hold. The remedy will be presented in section 4.2.5.

### 4.2.4 BER calculation with GMM-based PDF estimates

In this section, we shall derive the mathematical expression of BER estimate when using the Gaussian Mixture Model-based PDF estimator.

Let us recall the PDF-based BER expression given by (Eq. 2.55). Assume that we know the partitions of the received observations $(X_i)_{1\leq i\leq N}$ into two classes, $C_+$ and $C_-$, which contains the observed output such as the corresponding transmitted bit ($b_i = +1$ and $b_i = -1$, respectively). Thus, we need to perform the EM algorithm two times and in independent way :

- for the data base $C_+$, we perform the EM algorithm to estimate the unknown parameters of the $K_+$ Gaussians. Let $\theta_+^{(T)} = (\alpha_k^{+(T)}, \mu_k^{+(T)}, \sigma_k^{+(T)2})_{1\leq k\leq K_+}$ be the reached values at the last iteration $T$ of the EM algorithm ;
- for the data base $C_-$, we perform the EM algorithm to estimate the unknown parameters of the $K_-$ Gaussians. Let $\theta_-^{(T)} = (\alpha_k^{-(T)}, \mu_k^{-(T)}, \sigma_k^{-(T)2})_{1\leq k\leq K}$ be the reached values at the last iteration $T$ of the EM algorithm.

The reliable estimate of the parameter $\theta_+^{(T)}$ (resp. $\theta_-^{(T)}$) allows the estimation of the conditional PDF $f_X^{b_+}(\cdot)$ (resp. $f_X^{b_-}(\cdot)$) and the computation of BER.

Let $\hat{f}^{b_+}_{X,N_+}(x)$ and $\hat{f}^{b_-}_{X,N_-}(x)$ be the two conditional PDFs estimates, from Eq. 2.55 we can express the BER estimates as :

$$\hat{p}_{e,N} = \pi_+ \underbrace{\int_{-\infty}^{0} \hat{f}^{b_+}_{X,N_+}(x)dx}_{D_+} + \pi_- \underbrace{\int_{0}^{+\infty} \hat{f}^{b_-}_{X,N_-}(x)dx}_{D_-} \tag{4.27}$$

where $P[b_i = +1] = \pi_+$ and $P[b_i = -1] = \pi_-$, with $\pi_+ + \pi_- = 1$.

We shall compute the values of $D_+$ and $D_-$ by using the reached values of parameters $\theta_+^{(T)}$ and $\theta_-^{(T)}$. Given the fact that the two conditional PDFs are estimated using the Gaussian Mixture Model, according to Eq. 2.49, for $D_+$, we get [STTP10] :

$$\begin{aligned} D_+ &= \int_{-\infty}^{0} \sum_{k=1}^{K_+} \alpha_k^+ f_k^+(x; \mu_k^+, \sigma_k^{+2})dx \\ &= \sum_{k=1}^{K_+} \alpha_k^+ \int_{-\infty}^{0} \frac{1}{\sqrt{2\pi}\sigma_k^+} \exp\Big(-\frac{(x-\mu_k^+)^2}{2\sigma_k^{+2}}\Big)dx \end{aligned}$$

Let $z = (x - \mu_k^+)/\sigma_k^+$, we get :

$$D_+ = \sum_{k=1}^{K_+} \alpha_k^+ \int_{\frac{\mu_k^+}{\sigma_k^+}}^{+\infty} \frac{1}{\sqrt{2\pi}} \exp\Big(-\frac{z^2}{2}\Big)dz$$

Invoking the classical complementary unit cumulative Gaussian distribution $Q(\cdot)$, we have :

$$D_+ = \sum_{k=1}^{K_+} \alpha_k^+ Q\Big(\frac{\mu_k^+}{\sigma_k^+}\Big) \tag{4.28}$$

In the same way, for $D_-$, we get :

$$D_- = \sum_{k=1}^{K_-} \alpha_k^- Q\Big(-\frac{\mu_k^-}{\sigma_k^-}\Big) \tag{4.29}$$

Combining Eq. 4.27, Eq. 4.28 and Eq. 4.29, we can derive the expression of the BER estimate :

$$\hat{p}_{e,N} = \pi_+ \sum_{k=1}^{K_+} \alpha_k^+ Q\Big(\frac{\mu_k^+}{\sigma_k^+}\Big) + \pi_- \sum_{k=1}^{K_-} \alpha_k^- Q\Big(-\frac{\mu_k^-}{\sigma_k^-}\Big) \tag{4.30}$$

**Remark** : For simplicity, in our works, $K_+$ is equal to $K_-$.

### 4.2.5 Optimal choice of the number of Gaussian components

In section 4.2.3., we have seen an example of PDF estimation based on Gaussian Mixture method. It is easy to guess the components number ($K = 3$) from the histogram shown in Fig. 4.1. In practice, this could be very difficult so we need to use some heuristic search to find the optimum number of Gaussian PDFs.

Let us remark that :
- if $K$ is too small :
  this corresponds to the case where the smoothing parameter $h_N$ is too large, as shown in Fig. 2.4 (b)-2. The PDF estimate will be too smooth ;
- if $K$ is too big :
  the same class of observations will come from different Gaussian components and the different Gaussians will be correlated. This is not useful for simulation since all the received samples are assumed to be independent.

Consequently, the optimal choice of the number of Gaussians consists in finding the largest one such that all the components are independent. We suggest initializing the EM algorithm with a high enough value of $K$, and testing the independence of components at the end of EM iteration. If it is not the case, we have to decrease iteratively the number of components until the independence is reached.

An adaptive way to test the independence of two components, $k_1$ and $k_2$, is the mutual information theory [Sha01, YZ01]. For $C_+$, we consider the mutual relationship defined in [STTP10, LLL06] :

$$MI_+(k_1, k_2) = p_+(k_1, k_2) \log_2 \frac{p_+(k_1, k_2)}{p_+(k_1)p_+(k_2)} \tag{4.31}$$

where :
- $p_+(k_1)$ and $p_+(k_2)$ are the probabilities of the mixtures $k_1$ and $k_2$, respectively, i.e.,

$$p_+(k_1) = \alpha_{k_1}^+ = \frac{1}{N_+} \sum_{i=1}^{N_+} \rho_{k_1,i}^+$$

$$p_+(k_2) = \alpha_{k_2}^+ = \frac{1}{N_+} \sum_{i=1}^{N_+} \rho_{k_2,i}^+$$

- $p_+(k_1, k_2)$ is the joint probability of the two components, i.e.,

$$p_+(k_1, k_2) = \frac{1}{N_+} \sum_{i=1}^{N_+} \rho_{k_1,i}^+ \rho_{k_2,i}^+$$

The sign of Eq. 4.31 allows us to know the independence of the two components :
- if $sign(MI_+(k_1, k_2)) = 0$, $p_+(k_1, k_2)$ is equal to the product of the probabilities of the two components, i.e., $p_+(k_1, k_2) = p_+(k_1)p_+(k_2)$. Thus, the two components are independent ;
- if $sign(MI_+(k_1, k_2)) < 0$, the two components are much less correlated ;
- if $sign(MI_+(k_1, k_2)) > 0$, the two components are statistically dependent and one of them can be removed.

For $k_1, k_2 = 1, \ldots, K_+$, we define the indicator $I_+$ which denotes the maximum value of the mutual information $MI_+(k_1, k_2)$.

$$I_+ = max[MI_+(k_1, k_2)]_{1 \leq k_1, k_2 \leq K_+}$$

Thus, we can use the indicator $I_+$ and $I_-$ to compute the optimal value of $K_+$ and $K_-$ in a parallel way. At the end of the last iteration of EM algorithm, if $sign(I_+) \leq 0$ ($sign(I_-) \leq 0$), we stop the algorithm, otherwise we have to decrease the number of components by one.

The following equation can be used to choose the component which should be removed.

$$k = \arg\max_{1 \leq k_1 \leq K_+} \sum_{k_2=1}^{K_+} MI_+(k_1, k_2)$$

The quantity $\sum_{k_2=1}^{K_+} MI_+(k_1, k_2)$ represents the mutual information for the component $k_1$ and denotes whether this component has a significant and independent contribution to the PDF estimation. If the largest positive value is found, the $k^{th}$ component has a dependent contribution to the Gaussian Mixture and should therefore be removed.

For a new decreased value, $K_+ - 1$ ($K_- - 1$), initial parameters of the new EM algorithm can be given by the output parameters at the last iteration of the previous EM algorithm. However, the initial probabilities of the new mixtures, $\alpha_k^{+(0)}$ ($\alpha_k^{-(0)}$) should be redefined to match the constraint $\sum_{k=1}^{K} \alpha_k = 1$.

### 4.2.6 Conclusion of Gaussian Mixture Model-based BER estimation using Expectation-Maximization algorithm

The flow chart of the proposed GMM-based algorithm is shown in Fig. 4.4. The EM algorithm can be considered as an extension method of the K-means algorithm. We started with a sufficient big value of $K_+$ (resp., $K_-$) then utilize the previous method to choose the optimal number of Gaussians. Normally, $K_+$ and $K_-$ have the same initial value since the soft output of $C_+$ and $C_-$ should have similar distribution.

Furthermore, the maximum number of EM iterations $T$ should be chosen carefully :
- if $T$ is too small, the log-likelihood probability cannot be well maximized ;
- if $T$ is too big, the simulation time will be enormous although the log-likelihood probability has already reached the maximum value..

It is helpful to define a criterion of the maximum iteration number $T$. We suggest using the "Floating-Point Relative Accuracy"(FPRA), $c_{FPRA}$, which is constant for a certain simulation platform. The rule for stopping the EM iteration is :

$$\text{EM iteration} \begin{cases} \text{break} & \text{if } L^{(t)}(\theta) - L^{(t-1)}(\theta) \leq c_{FPRA}; \\ \text{continue} & \text{otherwise.} \end{cases} \tag{4.32}$$

where $L^{(t)}(\theta)$ presents the log-likelihood probability at iteration $t$, $t \leq T$.

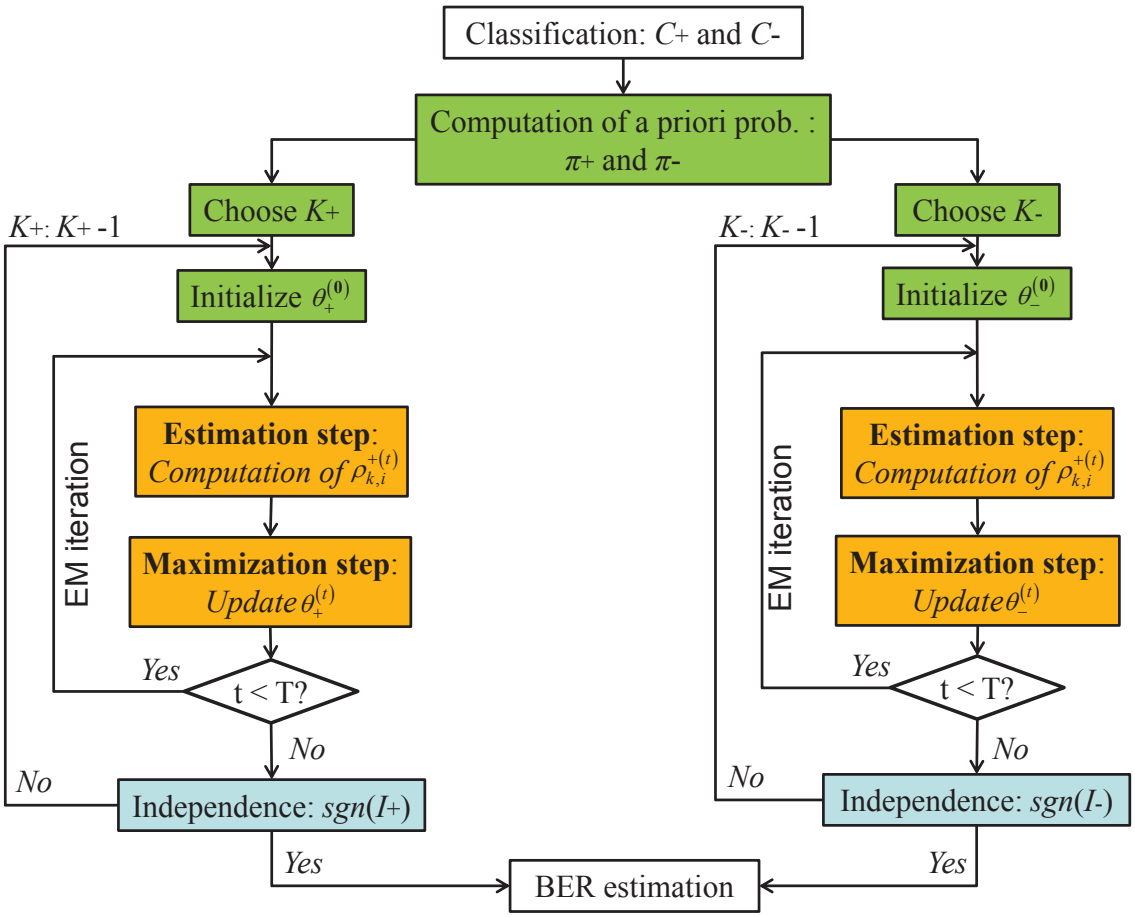


***Figure 4.4*** — Flow chart for the suggested BER estimation based on EM-GMM and Mutual Information theory [STTP10]

## 4.3 Simulation results of BER estimation based on Gaussian Mixture Model

To evaluate the performance of the proposed non parametric BER estimation based on Gaussian Mixture method. We have considered the same frameworks as given in Chapter 3 :

- Sequence of BPSK signal over AWGN and Rayleigh channels ;
- CDMA system ; for simplicity, we only consider the performance of the BER estimation for the system employing decorrelator-based receiver ;
- Turbo coding system ;
- LDPC coding system.

### 4.3.1 Sequence of BPSK symbol over AWGN and Rayleigh channels

Fig 4.5 shows the GMM-based BER estimation for BPSK signal over AWGN channel. We have taken 2000 samples, single Gaussian component has been used.

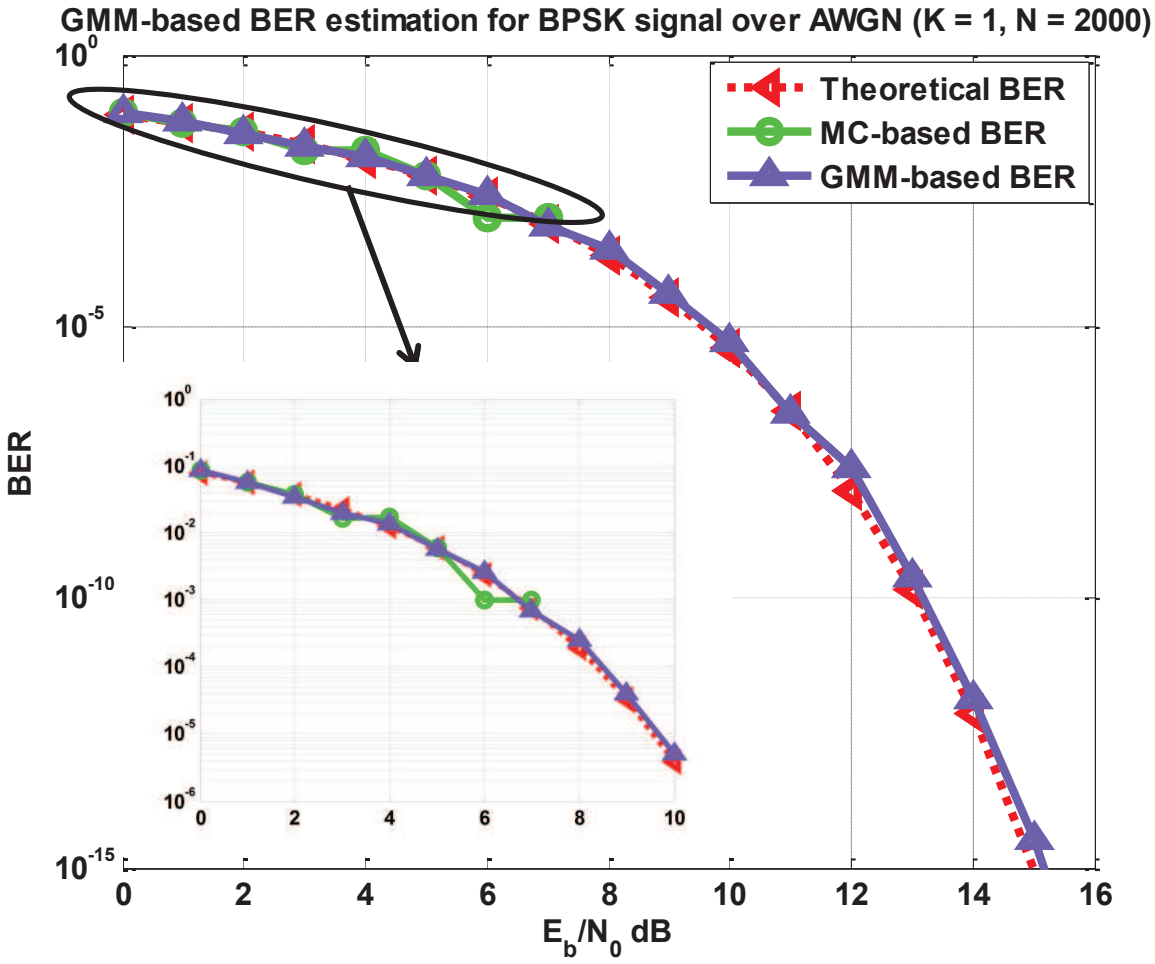


***Figure 4.5*** — GMM-based BER estimation for BPSK signal over AWGN channel ($K = 1$)

It can be shown that the Monte-Carlo simulation fails to return reliable estimate when $SNR > 7\ dB$, whereas the proposed estimation method based on Gaussian Mixture Model provides extremely perfect performance for almost all studied SNR values. The proposed GMM method results in a better estimate even for very high SNR.

Moreover, compared with the Kernel-based simulation results (cf. Fig 3.11) which depend on (in the sense of curve form) the BER estimates using Monte-Carlo simulation (i.e., Kernel-based BER estimates for $SNR \geq 6\ dB$), the GMM method does not exhibit such dependence. In fact, this dependence on MC-based BER estimate originates in the fact that, for high SNR, the PDF is badly estimated and becomes extreme matte.

Then we take a GM model which contains 3 Gaussian components.

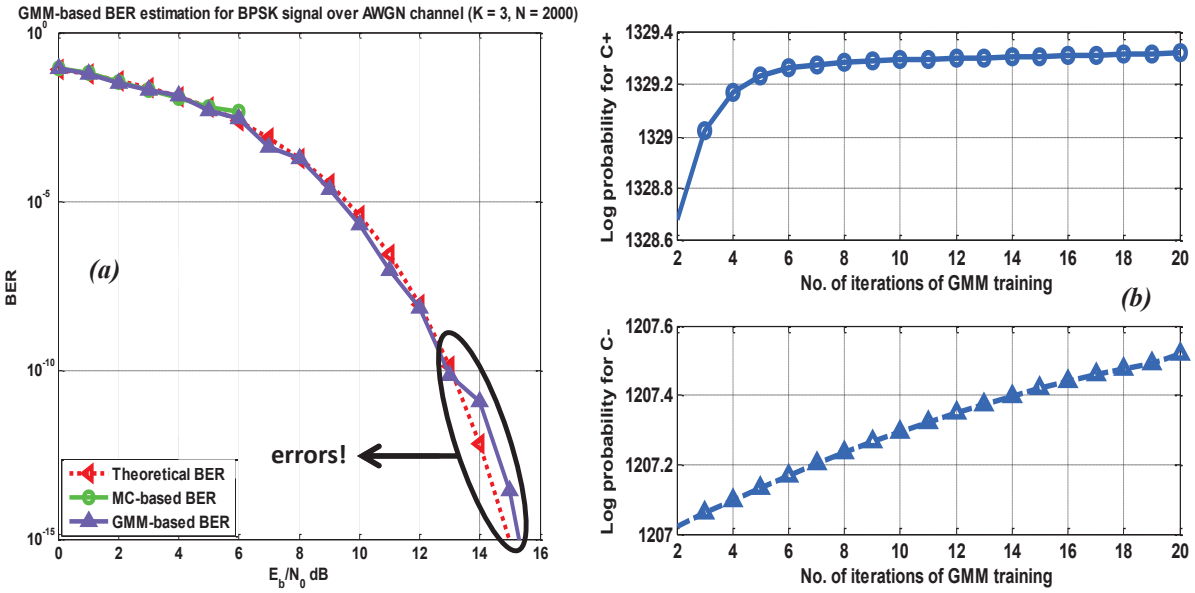


***Figure 4.6*** — GMM-based BER estimation and log-likelihood probabilities ($SNR = 18\ dB$) for BPSK signal over AWGN channel ($K = 3$)

Fig 4.6 (a) shows the BER estimation results for $K = 3$ and $T = 20$. The estimation performance is similar to the previous one shown in Fig 4.5. However, we found the significant mismatch errors when $SNR > 13\ dB$.

Fig 4.6 (b) shows the log-likelihood probabilities for $SNR = 18\ dB$. For the class $C_+$, the maximization step is well performed, whereas for $C_-$ it fails to be done (we can imagine that the value of log-likelihood probability is still increasing when $t > 20$). This is the reason why we observed that mismatch error.

Therefore, it might be helpful to theoretically or "intuitively" estimate the channel information and the soft output (e.g., by analyzing the histogram) before employing the proposed GM method. If a quasi-Gaussian distribution is found, only few Gaussian components can be taken to simplify the simulation complexity and reduce the computing time.

We also have analyzed the BER estimation performance when BPSK signal is transmitted over Rayleigh channel, as shown in Fig 4.7. We have taken 2000 data samples. The maximum iteration number is 10 and we shall study the BER estimation results at different number of Gaussian components.

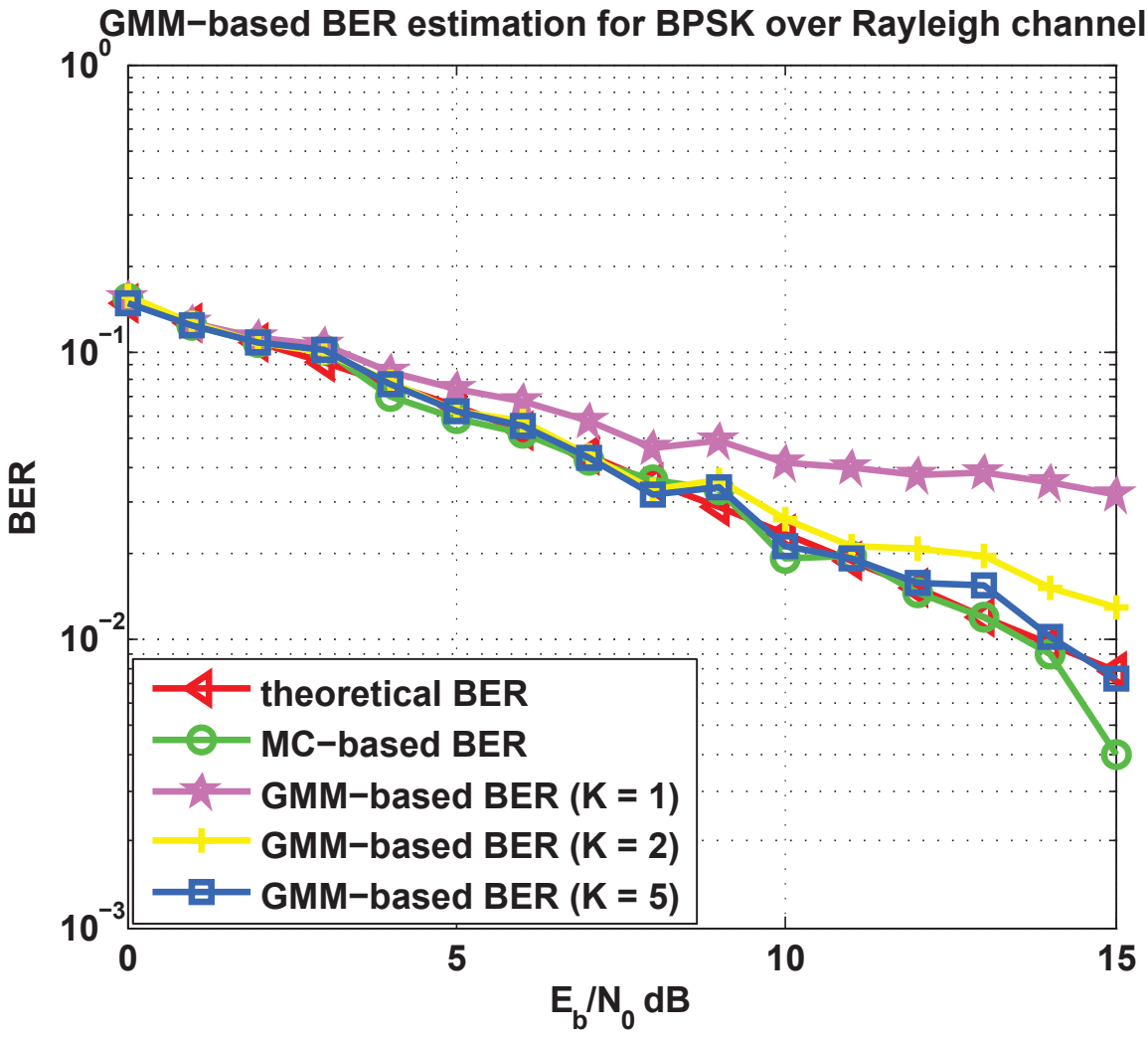


***Figure 4.7*** — GMM-based BER estimation for Rayleigh channel at different $K$ ($K = 1,\ 2,\ 5$), $N = 2000$, $T = 10$

It can be shown that :

- when $K = 1$, the BER estimate is good for low SNR. However, for high SNR, the GMM method results in an inaccurate estimate since the distribution is non-Gaussian ;
- when $K = 2$, the performance of BER estimation becomes better but still to be improved ;
- when $K = 5$, the performance at high SNR is much better. Also, we can observe that the GMM method also exhibits a dependence on Monte-Carlo simulation, as the Kernel method does. It seems that this dependence goes somewhat against the previous conclusion, since we did not note any dependence and similarity between the obtained GMM-based BER curve and the Monte-Carlo one. To explain that, we consider the histogram and the PDF estimates (Fig 4.8) at different $K$ for $SNR = 10\ dB$. The database is completely same as the one used in the previous BER estimation.

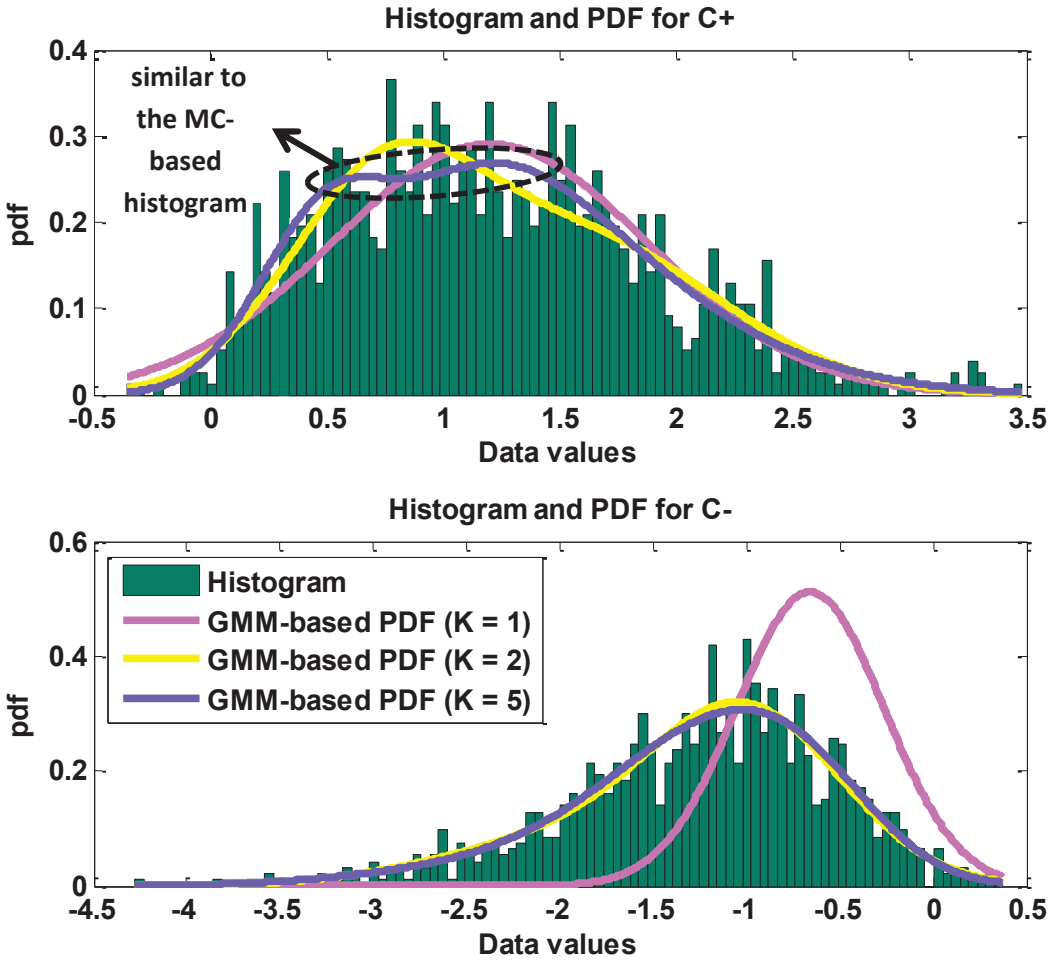


***Figure 4.8*** — Histogram and GMM-based PDF estimation for Rayleigh channel at different $K$ ($K = 1, 2, 5$), $N = 2000$, $T = 10$, $SNR = 10\ dB$

As increasing the number of Gaussians, the PDF estimation performance is improved, which means the PDF curve matches closely to the histogram. An obvious relationship between the PDF estimate for $C_+$ and the histogram can be found at the top of the PDF curve.

Basically, the GM method keeps step with the Kernel method in the final aim — the PDF estimate need to be sufficiently smooth and also should be a reflex of the histogram. In this sense, the number of Gaussians, $K$, and the maximum iteration number, $T$, also can be considered as a couple of "smoothing parameter". For non-Gaussian distribution :

- a bigger number of Gaussian components provide better fitting to the Monte-Carlo histogram ;
- more iterations also allow improving the estimation performance in the sense of the histogram fitting. For the simulation shown in Fig 4.7, we have taken 10 iterations as the maximum value. We found that the log-likelihood maximization did not reach the limit of the Floating-Point Relative Accuracy. Therefore, more iterations can be used.

**Remark** : it is important to note that, increasing the number of EM-iteration does not always provides the most accurate BER estimate. In fact, in single trial, if the performance of Monte-Carlo simulation is very poor, the proposed estimation methods could not work well even if we perform the "best" fitting to histogram.

Increasing iteration number shall increase the computing time. Thus, it is useful to find a solution that keep the estimation performance with lower computational cost.

Note that the number of EM-iteration depends on the algorithm that determines the initial values, $\theta^{(0)} = (\alpha_k^{(0)}, \mu_k^{(0)}, \sigma_k^{(0)2})_{1 \leq k \leq K}$, of unknown parameters . Especially, the value of $\mu_k^{(0)}$ should be carefully selected (in general, $\alpha_k^{(0)}$ is set to $1/K$, $\sigma_k^{(0)2}$ is set to the min squared distance between centers $\mu_k^{(0)}$). if suitable initial values are used, the number of EM-iteration can be reduced.

The simplest solution is to initialize the unknown parameters in a random fashion, i.e., we randomly select a data point as the mean $\mu_k^{(0)}$. Obviously, this is not a good solution since the selected $\mu_k^{(0)}$ could be far from the true center of the set of observations.

To initialize $\theta^{(0)}$, we use the Forgy's batch-version K-means algorithm described in section 4.1.1.2. In Fig 4.9, we report the behavior of GMM-based BER estimation for BPSK system over Rayleigh channel with the two initialization methods. The number of Gaussians is set to 3 and the maximum number of EM iterations is set to 10.

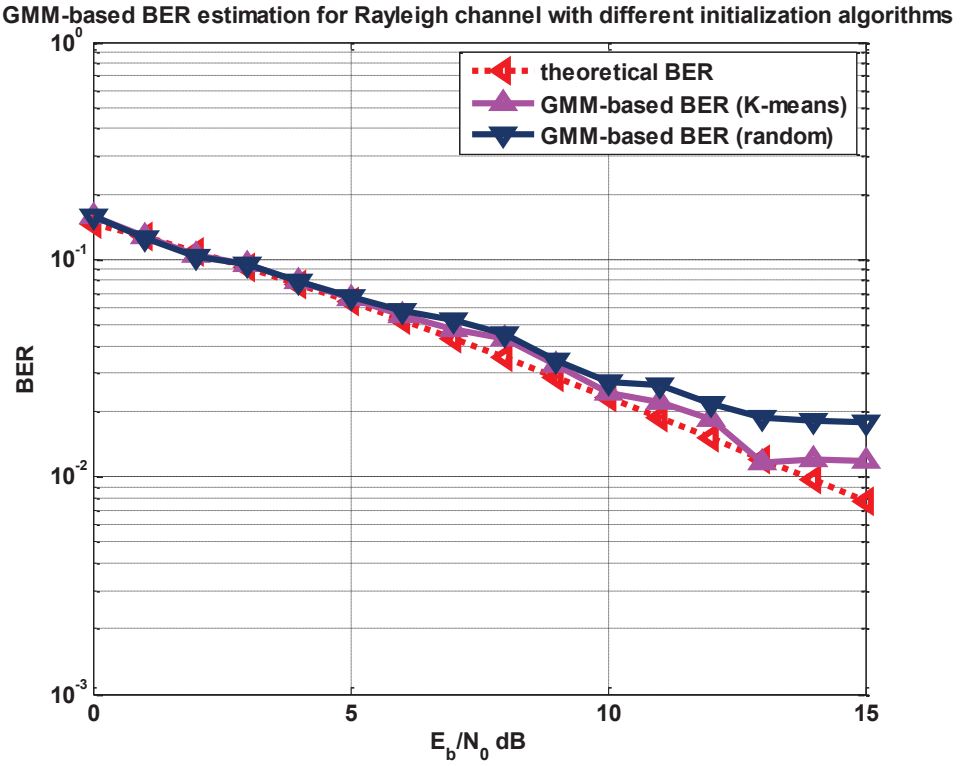


***Figure 4.9*** — GMM-based BER estimates for BPSK system over Rayleigh channel with different initialization methods ($K = 3$, $T = 10$)

It can be shown that the K-means algorithm provides more reliable BER estimates for 10 iterations. Whereas initializing the parameters in a random way may cause unreliable estimates, this can be improved by increasing the number of iterations or using more data samples but obviously the computing time must be increased.

Many other methods can also be used to initialize the unknown parameter, such as MacQueen's mode [Gna11] and Y. LU's version [LLF+04].

**Remark** :
- we do not have to use the sequential K-means algorithm since the characteristics of the dataset is not varying with time ;
- A better set of initial centers will still have positive influence for K-means algorithm. Some commonly used methods for initial center selection include :
  1. randomly select $K$ data points from the dataset ;
  2. select the farthest/nearest $K$ data points from the mean of the dataset ;
  3. select $K$ data points that have the largest sum of pairwise square distance ;
  4. choose the centers one by one based on K-means++ [AV07].

  In our works, the initial centers are randomly selected.

### 4.3.2 CDMA system with decorrelator-based receiver

We consider the same CDMA system model with 2 users as presented in section 3.2.2.2. Fig 4.10 shows the BER estimation for decorrelator-based CDMA system over AWGN channel at different numbers of iteration. We have taken 2000 data samples. 4 Gaussian components have been used.

It can be shown that the performance of the GMM-based BER estimation is distinctly improved as increasing the number of iterations. This is always true for any configuration of parameters since the log-likelihood probability increases monotonically with increasing of iteration number.

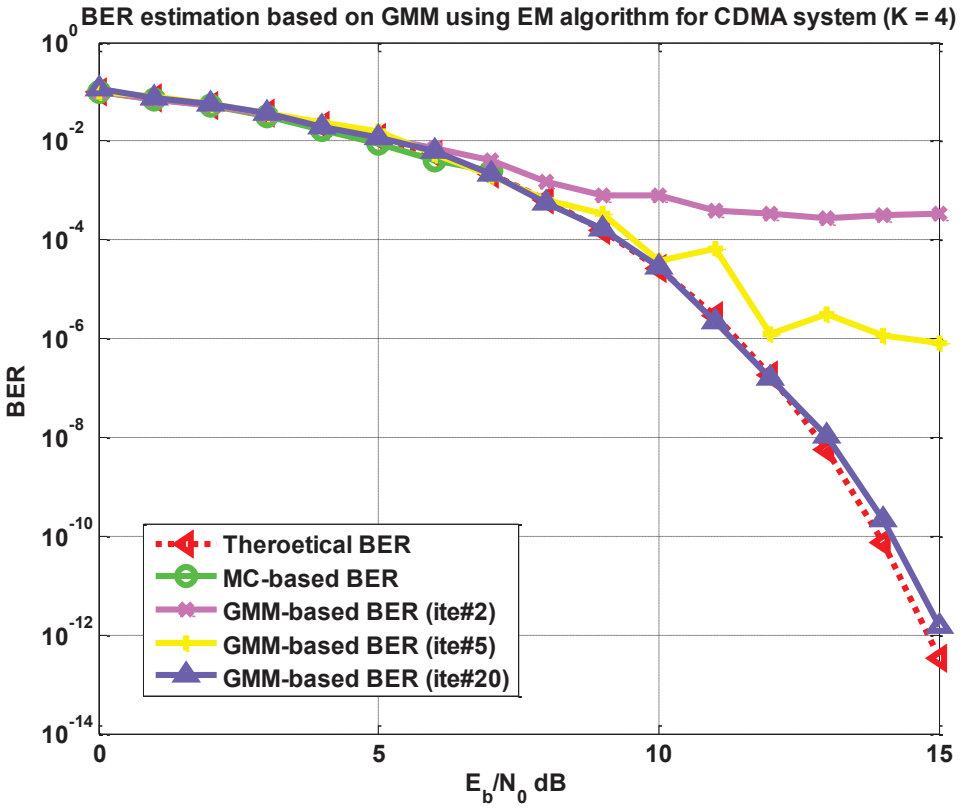


***Figure 4.10*** — GMM-based BER estimation for CDMA system using decorrelator at different iteration numbers ($K = 4$, $N = 2000$)

For $SNR = 20\ dB$ and maximum iteration number $= 5$, we plot the values of the log-likelihood probabilities, as shown in Fig 4.11. To clearly visualize the increasing trends, we started from the $2^{nd}$ iteration.

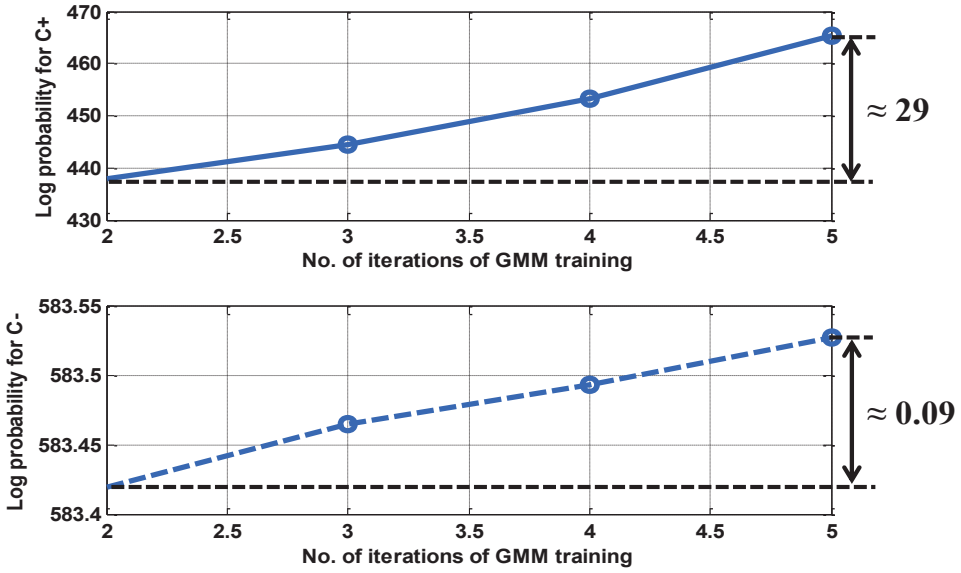


***Figure 4.11*** — Log-likelihood probabilities for decorrelator-based CDMA system ($N = 2000$, $K = 5$, $SNR = 20\ dB$, $T = 5$)

We observed the significant increasing either for both two classes, especially for $C_+$ where the increase is by a factor of 29 from the $2^{nd}$ to the $5^{th}$ iteration.

This characteristic of the GM method provides flexibility for the estimation, but can also be partly considered as a drawback — the estimation always requires long simulation time, since we have to increase the iteration number to have a good precision while taking several Gaussian components. That is why we suggest optimizing the number of Gaussians, as presented in section 4.2.5.

Moreover, we found that extreme small BER values can be obtained. For this reason, we guess the PDF of the soft observations of deccorelator-based CDMA system over AWGN channel is similar to Gaussian. Therefore, it is possible to take fewer components.

Fig 4.12 shows the BER estimation using GM method with different number of Gaussian components for decorrelator-based CDMA system over AWGN channel.

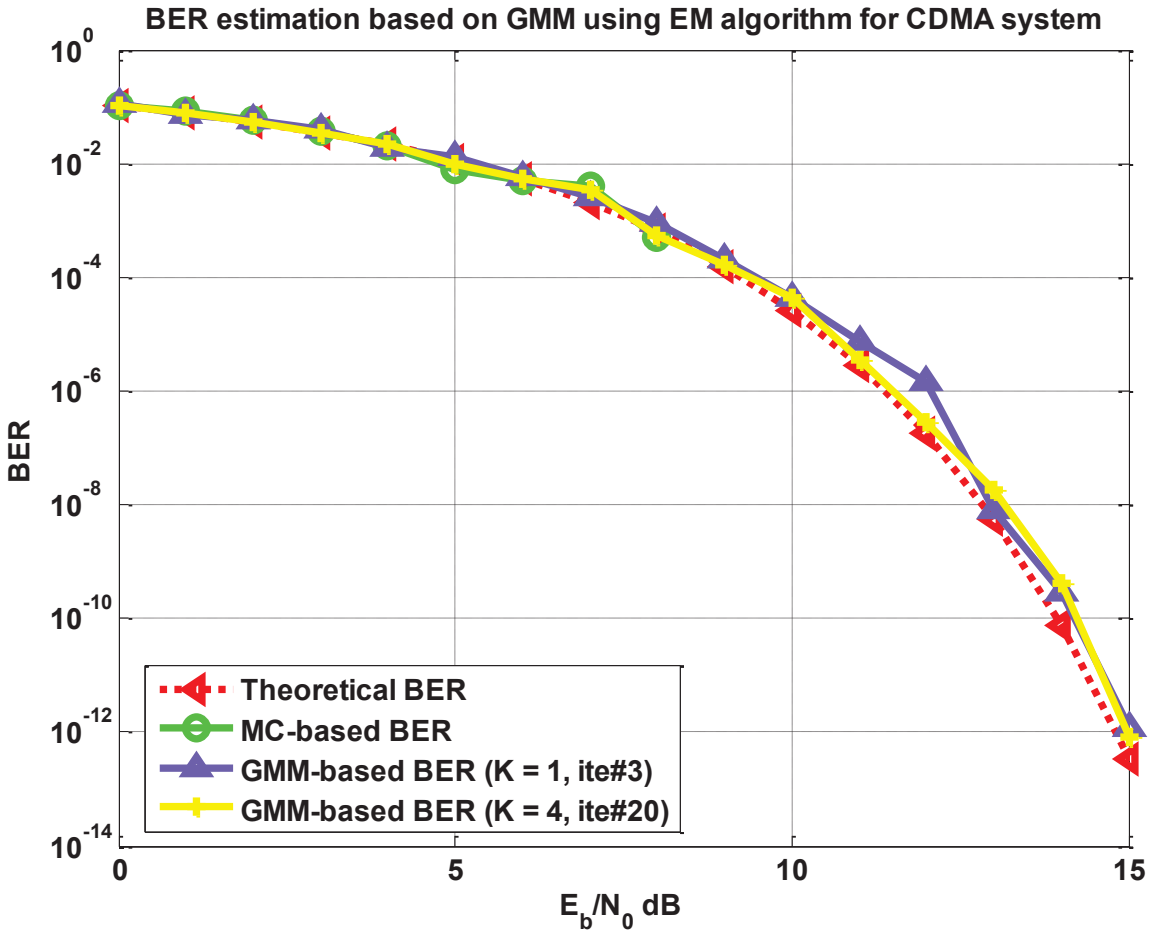


***Figure 4.12*** — GMM-based BER estimation for CDMA system with decorrelator ($K = 1$ and 4, $N = 2000$)

It can be shown that the performance of BER estimation based on GM method with single Gaussian is not bad even for high SNR values (only with some fluctuations). The maximum number of iterations could be greatly reduced (only 3 iterations lead to reach the criterion of the Floating-Point Relative Accuracy) and the simulation time for such system is only 2 seconds (from $SNR = 0\ dB$ to $15\ dB$).

Fig 4.13 shows the histograms and the GMM-based PDF estimates for the CDMA system over AWGN channel when $SNR = 6\ dB$. It can be shown that the true PDF is quite similar to Gaussian function.

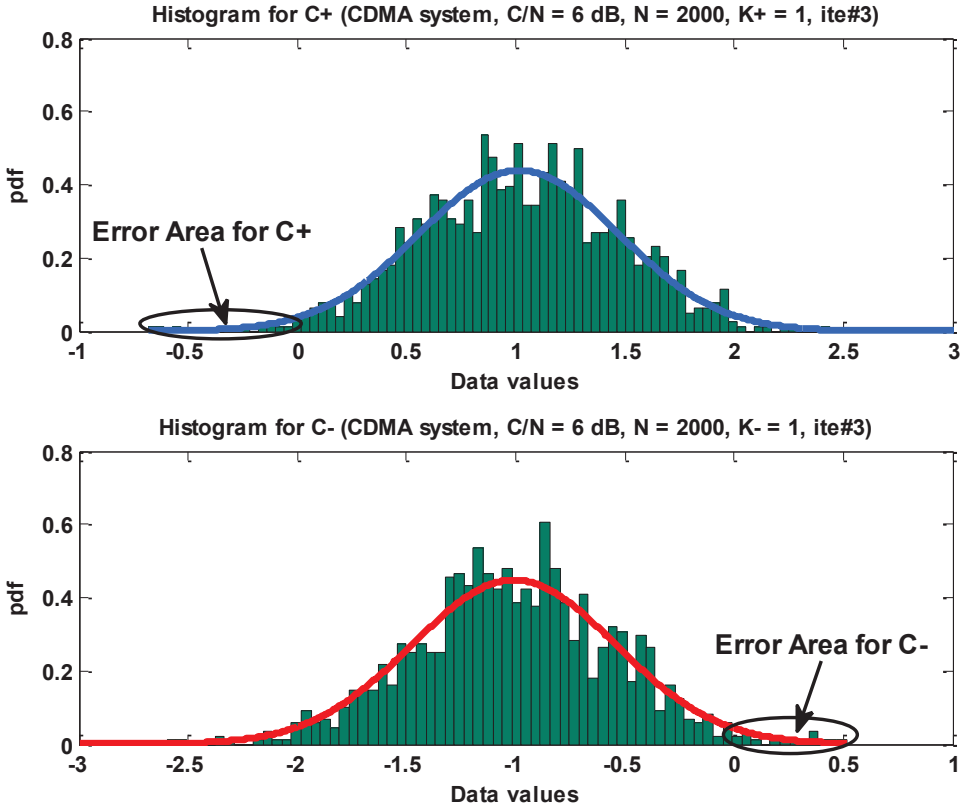


***Figure 4.13*** — Histograms and GMM-based PDF estimates for CDMA system over AWGN ($SNR = 6\ dB$, $N = 2000$, $K = 1$, $T = 3$)

Furthermore, we have listed the estimated parameters, $\mu_k^{+(3)}$ and $\sigma_k^{+(3)2}$, $1 \leq k \leq K, K = 1$ or 4, for $SNR = 15\ dB$, as shown in Tab. 4.2. We can find that, either for $K = 1$ or for $K = 4$, the values of the Gaussian means ($C_+$ or $C_-$) are close to +1 or -1, respectively.

| | $\mu_k^+$ | $\mu_k^-$ | $\sigma_k^{+2}$ | $\sigma_k^{-2}$ |
|---|---|---|---|---|
| $K = 1,$ max ite. number $= 3$ | 0.9973 | -1.0005 | 0.0160 | 0.0159 |
| $K = 4,$ max ite. number $= 20$ | 1.0345 | -1.1092 | 0.0162 | 0.0160 |
| | 0.9215 | -0.8996 | 0.0165 | 0.0166 |
| | 0.9694 | -1.0302 | 0.0156 | 0.0154 |
| | 1.0301 | -0.9875 | 0.0156 | 0.0153 |

**Table 4.2** — Estimated Gaussian Parameters for CDMA system ($SNR = 20\ dB$, $N = 2000$, $K$ =1 and 4)

Next, we have compared the variance of GMM-based BER estimates with the Kernel-based results and the one of Monte-Carlo simulation, as shown in Tab. 4.3. The variance of the BER estimation is computed by performing 1000 trials.

| | | SNR = 0 $dB$ | SNR = 4 $dB$ | SNR = 6 $dB$ |
|---|---|---|---|---|
| Monte-Carlo | Variance | $5.0 \cdot 10^{-5}$ | $2.4 \cdot 10^{-5}$ | $10.3 \cdot 10^{-6}$ |
| Kernel using optimal $h_N$ | Variance | $4.3 \cdot 10^{-5}$ | $2.0 \cdot 10^{-5}$ | $1.0 \cdot 10^{-6}$ |
| GMM | Variance | $3.7 \cdot 10^{-5}$ | $1.6 \cdot 10^{-5}$ | $0.85 \cdot 10^{-6}$ |

**Table 4.3** — Variance of BER estimation for Monte-Carlo and Kernel methods for different SNR using 2000 data samples and 1000 trials

It can be shown that the GMM method provides the best performance in the sense of the minimum variance (or standard deviation) of BER estimation. This is a logical conclusion — in fact, we have found that the PDF of the soft output for the studied CDMA system is similar to Gaussian function, and the GM model has natural advantage on training the quasi-Gaussian distribution.

### 4.3.3 Turbo coding system

The $3^{rd}$ framework is the same turbo code-based system as given in section 3.2.7. Fig 4.14 shows the GMM-based BER estimation and the log-likelihood probabilities. We have taken 300,000 samples in 600 frames. 5 Gaussian components are used and the maximum iteration number is set to 30.

It can be shown that :

- the obtained BER estimates are quite similar to the Kernel-based results shown in Fig 3.20 ;
- although the predefined maximum number of iterations is 30, we found that only 6 iterations are performed (Fig 4.14 (b)), since the criterion of the Floating-Point Relative Accuracy is reached.

As we have taken a big number of data samples, it makes sense to compare the simulation time of the GMM-based estimation with the one of the Kernel-based estimation, as shown in Tab. 4.4. 600 frames are used for each simulation. The simulation time is counted by using MATLAB.

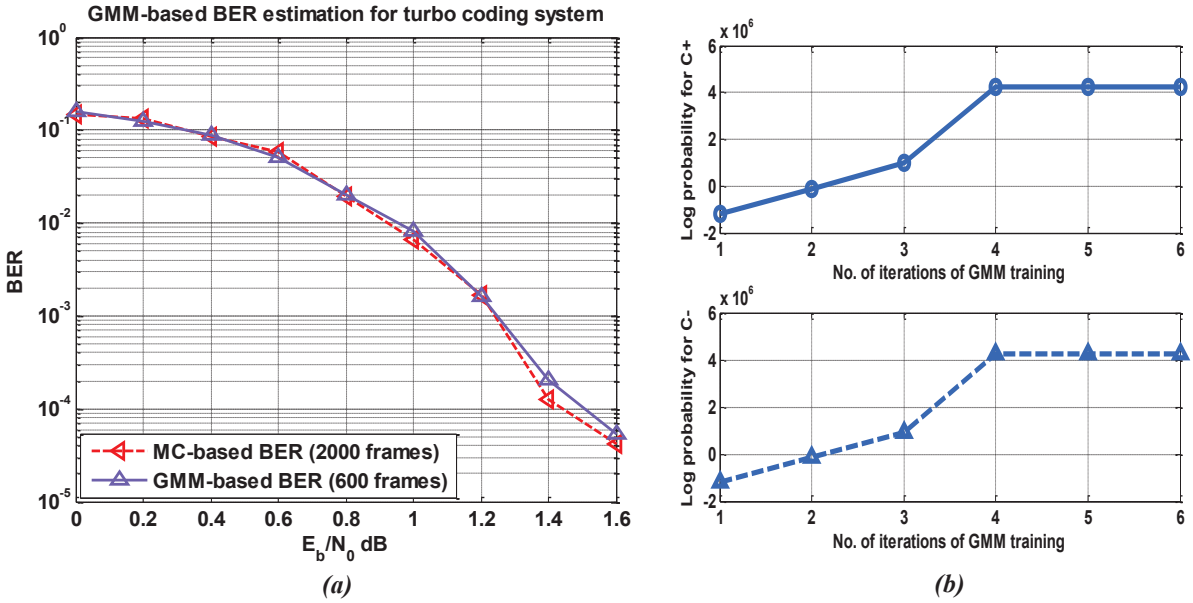


***Figure 4.14*** — (a). GMM-based BER estimation for turbo code system (600 frames, $K = 5$, $T = 30$) ; (b). Log-likelihood probabilities ($SNR = 1.6\ dB$)

| BER estimation method | BER | simulation run time |
|---|---|---|
| Kernel method using optimal $h_N$ | $10^{-4}$ | 5 min 15 s |
| | $10^{-5}$ | 41 min 5 s |
| GMM | $10^{-4}$ | 1 min 5 s |
| | $10^{-5}$ | 8 min 45 s |

**Table 4.4** — Simulation run time at different BER for GMM-based and Kernel-based BER estimation

It can be shown that the GMM method requires less time to finish the simulation. The Kernel method is less efficient due to the complexity of the algorithm for bandwidth optimization.

### 4.3.4 LDPC coding system

A 1/2 rate Quasi-Cyclic-LDPC (QC-LDPC) system has also been taken to analyze the performance of GMM-based BER estimation. The soft observations are given by Eq. 3.55

As presented in section 3.3.4, it is suggested to modify the original dataset to reduce simulation run time. Fig. 4.15 shows the GMM-based BER estimation results while using either the modified dataset and the original unmodified dataset. 5 Gaussian components are used (the value of $K$ could be reduced as employing the algorithm for optimizing the number of Gaussian components, cf. section 4.2.5). The maximum number of iteration is set to 30. We get rid of the soft outputs equal to $\pm 1$ for $SNR \geq 2\ dB$, for small values of SNR, we use the entire data samples to avoid losing information.

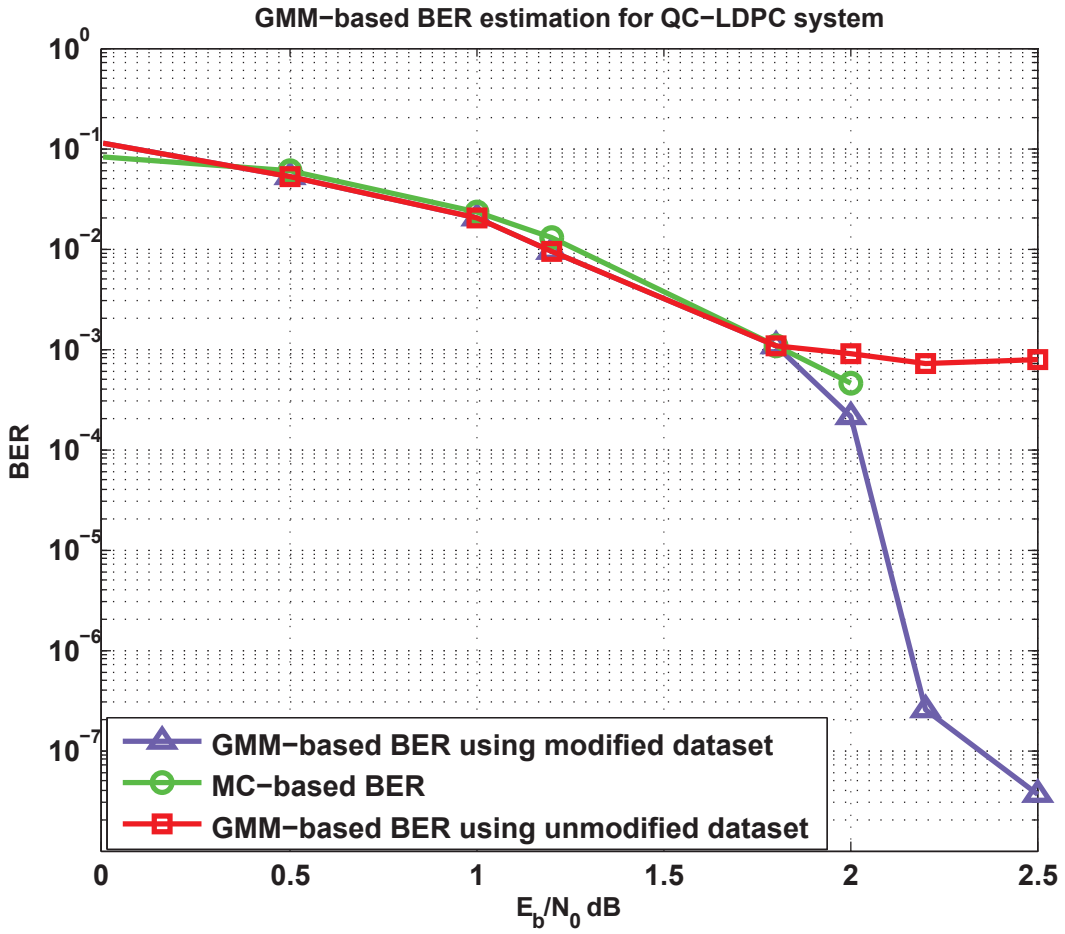


***Figure 4.15*** — BER estimation based on the modified GM method for QC-LDPC system (150 frames, K = 5, T = 30)

When using the modified dataset, the GMM-based BER estimator can provide accurate BER estimate even for high SNR, whereas the MC simulation fails to do so due to the very limited number of data samples (for MC method, we have to count 10 errors at different SNR). Compared with the Kernel-based simulation result (cf. Fig. 3.22) obtained by using 500 frames, only 150 frames ($150 \times 635 = 95{,}250$ samples) are used while using modified dataset. This means that the size of outputs (outputs $\in (-1, +1)$) is sufficient large to provide good precision.

Moreover, for high SNR, the BER estimates obtained by using the unmodified dataset are completely wrong. This seems unusual and cannot disappear even if we increase the size of dataset.

As shown in 3.23 and Tab. 3.7, the majority of soft outputs of the studied LDPC decoder are equal to +1 and -1. The PDF of the soft observations can be seem as two delta functions at +1 and -1 with small single-side tails and bias. Thus, the PDF estimation will be primarily "dedicated" to fit the "peaks" (+1 and -1) of the histogram and will ignore the tails. However, these "peaks" does not provide any information of the bit errors.

We can sum up the importance of using the modified dataset for the studied QC-LDPC coding system :

– using the modified dataset can reduce the computational cost ;
– for GMM-based estimator, it is necessary to use the modified dataset in order to avoid the erroneous estimates for high SNR.

## 4.4 Conclusion

In this chapter, we have firstly presented the principle of probabilistic clustering by using mixture models. The entire dataset can be modeled by a mixture of distributions, such as Gaussian distributions, along with the known data samples and the missing data. Then we have introduced the Expectation-Maximization algorithm, which can be used to iteratively estimate the unknown parameters of the mixture model. In section 4.2.2, we have shown how to use the EM algorithm to estimate the unknown parameter $\theta = (\alpha_k, \mu_k, \sigma_k^2)_{1 \leq k \leq K}$ of Gaussian Mixture Model. Once we obtain the parameter and the missing data for both classes $C_+$ and $C_-$, the BER can be computed analytically. Then we have discussed the choice of the number of Gaussian components and reported the behavior of GMM-based estimator for different frameworks of communication system.

It was shown that, compared with the Kernel estimator, the GMM-based estimator provides better performance for Gaussian distribution, e.g., for the system with BPSK symbols over AWGN channel, the BER estimate obtained by using GMM method can be extremely close to the theoretical value even for very high SNR, whereas the Kernel method provides wrong results.

In the sense of the minimum variance, the GMM method provides the best performance for Gaussian distribution. Moreover, the GMM method is also efficient in the sense of the minimum computational cost : for Turbo coding system, we have compared the simulation run time between the Kernel method and GMM, we found that the GMM-based estimator takes few computing time and the Kernel method is less efficient due to the high complexity of the algorithm for bandwidth optimization.

The performance of GMM-based BER estimator strongly depends on the number of Gaussian components $K$ and the number of EM-iteration $T$. For Gaussian distribution, single Gaussian would be sufficient, whereas for non-Gaussian distributions, $K$ must be set to an enough large number. The choice of EM-iteration number depends on the initial values of the unknown parameters. In our works, the K-means algorithm was used to initialize these parameters of Gaussians. It was shown that, compared with the random selection, K-means algorithm allows reducing maximum number of iterations.

For the studied QC-LDPC coding system which use the pseudo APPs as the soft observations, it is necessary to modify the dataset (get rid of $\pm 1$ or set a threshold), whereas this is not required for Kernel-based estimator.

CHAPTER 5

# Unsupervised Bit Error Rate Estimation

In Chapter 2, we have presented the famous Monte-Carlo simulation and some modified methods (cf. section 2.1.1 - 2.1.5) for BER estimation. In Chapter 3 and 4, two BER estimation techniques based on Kernel method and Gaussian Mixture Model have been proposed. These techniques are called soft Bit Error Rate estimation since the BER is computed by estimating the PDF of receiver's soft output and no knowledge about hard decision information is required. Moreover, the proposed methods could apply to any digital communication system because the estimator does not depend on the communication model and the transmitter scheme.

Unfortunately, all the above mentioned methods assume that the whole transmitted data bits were known at the receiver side, i.e., for a given data set, we assumed that the exact partitions of the observations into two classes $C_+$ and $C_-$ were known. In practical situation, the BER estimation should be performed in an unsupervised way since we do not know this information.

In this chapter, we shall present an unsupervised Bit Error Rate estimation technique based on the well-known Stochastic Expectation Maximization (SEM) algorithm combined with the use of the Kernel method or the Gaussian Mixture Model, in an iterative way.

## 5.1 Unsupervised BER estimation based on Stochastic Expectation Maximization algorithm using Kernel method

The same notation as shown in Chapter 3 shall be used. Let $(X_i)_{1\leq i\leq N}$ be the independent soft receiver output which have the same PDF, $f_X(x)$. The transmitted information bits, $b_i = +1$ and $b_i = -1$ become the missing data which gives the classes $C_+$ and $C_-$ to which $X_i$ should be linked. Let $N_+$ (resp., $N_-$) be the number of elements of $C_+$ (resp., $C_-$), with $N = N_+ + N_-$. When using the Kernel method, the conditional PDF, $f_X^{b_+}(x)$ and $f_X^{b_-}(x)$, are given by Eq. 3.26. The task of the SEM algorithm is to find the two classes $C_+$ and $C_-$, and then estimate the set of unknown parameters $\theta = (N_+, \pi_+, h_{N_+}, N_-, \pi_-, h_{N_-})$.

In Chapter 4, we have presented the principle of Expectation Maximization algorithm for the BER estimation based on Gaussian Mixture Model. The Stochastic EM algorithm, proposed by Broniatowski, Celeux and Diebolt [CD$^+$84,CD85,CD$^+$86,CD87], is an expanding method by introducing a random rule in the classical EM technique.

The unsupervised iterative BER estimation is still performed through the **Initialization**, **Estimation** and **Maximization** steps, as the proposed GMM-based method using classical EM algorithm [SAIM11]. We denote $T$ as the maximum number of SEM iterations, at each iteration $t$, we update the values of the unknown parameters, $\theta^{(t)} = (N_+^{(t)}, \pi_+^{(t)}, h_{N_+}^{(t)}, N_-^{(t)}, \pi_-^{(t)}, h_{N_-}^{(t)})$.

### 5.1.1 Initialization

For the SEM-based unsupervised BER estimation, the most important parameters to be decided are the classes $C_+$ and $C_-$ which determines the values of the cardinalities ($N_+$ and $N_-$) and the smoothing parameters ($h_{N_+}$ and $h_{N_-}$). Moreover, the a priori probabilities, $\pi_+$ and $\pi_-$, are also required to compute the conditional PDF. Thus, we initialize the following parameters :

- $C_+^{(0)}$ and $C_-^{(0)}$ : the initial condition of the two subsets is very important, a bad choice may lead to increasing of iteration number. Since the soft output is the only data set we know, the $C_+^{(0)}$ and $C_-^{(0)}$ can be given by :

$$C_+^{(0)} = \{X_i | X_i \geq 0\}, \qquad C_-^{(0)} = \{X_i | X_i < 0\} \tag{5.1}$$

  for a given number of data samples, we can foresee that :

  a) for small values of SNR, the initial condition may lead to more erroneous partitions (i.e., those positive soft outputs corresponding to transmitted bit value -1 and vise versa), since the theoretical bit error probability is high ;
  b) for high SNR, the initial partitions may be quite exact. In this situation, small number of EM iterations is sufficient to have a good precision of the BER estimate.

- $N_+^{(0)}$ and $N_-^{(0)}$ :

$$N_+^{(0)} = \text{Card } C_+^{(0)}, \qquad N_-^{(0)} = \text{Card } C_-^{(0)} \tag{5.2}$$

- $h_{N_+}^{(0)}$ and $h_{N_-}^{(0)}$ : we can use Eq. 3.27 to compute the initial values of the smoothing parameters (optimal values for Gaussian distribution). The standard deviations of $(X_i)_{1 \leq i \leq N}$ in their corresponding classes $C_+^{(0)}$ and $C_-^{(0)}$, $\sigma_+^{(0)}$ and $\sigma_-^{(0)}$, need to be firstly computed ;
- $\pi_+^{(0)}$ and $\pi_-^{(0)}$ : these two parameters represent the a priori probabilities of the information bits and can be approximated by :

$$\pi_+^{(0)} = \frac{N_+^{(0)}}{N}, \qquad \pi_-^{(0)} = \frac{N_-^{(0)}}{N} = 1 - \pi_+^{(0)} \tag{5.3}$$

- finally, we compute the conditional PDF $f_X^{b_+(0)}$ and $f_X^{b_-(0)}$, given by Eq. 3.26, using the initial parameters $\theta^{(0)} = (N_+^{(0)}, \pi_+^{(0)}, h_{N_+}^{(0)}, N_-^{(0)}, \pi_-^{(0)}, h_{N_-}^{(0)})$.

### 5.1.2 Estimation step

The Estimation step of iteration $t$ consists in estimating the a posteriori probabilities of the unknown information bits, $(b_i)_{1\leq i\leq N}$, conditioned on observations $(X_i)_{1\leq i\leq N}$ with knowledge of the estimate $\theta^{(t-1)}$ obtained at the Maximization step of the previous iteration $t-1$. We denote $\rho_{i+}^{(t)}$ and $\rho_{i-}^{(t)}$ the APPs of $b_i$ at iteration $t$, we get :

$$\rho_{i+}^{(t)} = P(b_i = +1|X_i;\theta^{(t-1)}) = \frac{P(b_i=+1,X_i;\theta^{(t-1)})}{P(X_i;\theta^{(t-1)})} = \frac{P(X_i|b_i=+1;\theta^{(t-1)})P(b_i=+1)}{P(X_i;\theta^{(t-1)})}$$
$$\rho_{i-}^{(t)} = P(b_i = -1|X_i;\theta^{(t-1)}) = \frac{P(b_i=-1,X_i;\theta^{(t-1)})}{P(X_i;\theta^{(t-1)})} = \frac{P(X_i|b_i=-1;\theta^{(t-1)})P(b_i=-1)}{P(X_i;\theta^{(t-1)})}$$

Using Eq. 3.26, for $1 \leq i \leq N$, we have :

$$\begin{cases} \rho_{i+}^{(t)} = \dfrac{\pi_+^{(t-1)}\hat{f}^{b_+}_{X,N_+^{(t-1)}}(X_i)}{\pi_+^{(t-1)}\hat{f}^{b_+}_{X,N_+^{(t-1)}}(X_i)+\pi_-^{(t-1)}\hat{f}^{b_-}_{X,N_-^{(t-1)}}(X_i)} \\ \rho_{i-}^{(t)} = \dfrac{\pi_-^{(t-1)}\hat{f}^{b_-}_{X,N_-^{(t-1)}}(X_i)}{\pi_+^{(t-1)}\hat{f}^{b_+}_{X,N_+^{(t-1)}}(X_i)+\pi_-^{(t-1)}\hat{f}^{b_-}_{X,N_-^{(t-1)}}(X_i)} \end{cases} \tag{5.4}$$

### 5.1.3 Maximization step

In this step, at iteration $t$, we compute the estimate $\theta^{(t)}$ by maximizing the joint log-likelihood function based on conditional probabilities obtained at the Estimation step of the same iteration. The log-likelihood function, $L(\theta^{(t)})$, is given by [SAIM11] :

$$\begin{aligned} L(\theta^{(t)}) &= E\left[\log\Big(\prod_{i=1}^{N} f_{X_i,b_i}(X_i,b_i)\Big)|(X_i)_{i\leq i\leq N}\right] \\ &= \sum_{i=1}^{N} E\Big[\log\Big(f_{X_i,b_i}(X_i,b_i)\Big)|X_i\Big] \\ &= \sum_{i=1}^{N}\left[\rho_{i+}^{(t)}\log\Big(\pi_+^{(t)}\hat{f}^{b_+}_{X,N_+^{(t)}}(X_i)\Big)+\rho_{i-}^{(t)}\log\Big(\pi_-^{(t)}\hat{f}^{b_-}_{X,N_-^{(t)}}(X_i)\Big)\right] \end{aligned} \tag{5.5}$$

Invoking a Lagrange Multiplier and taking into account the constraint $\pi_+^{(t)}+\pi_-^{(t)}=1$, we have :

$$\begin{aligned} L_{\text{Lagrange}}(\theta^{(t)}) = &\sum_{i=1}^{N}\left[\rho_{i+}^{(t)}\log\Big(\pi_+^{(t)}\hat{f}^{b_+}_{X,N_+^{(t)}}(X_i)\Big)+\rho_{i-}^{(t)}\log\Big(\pi_-^{(t)}\hat{f}^{b_-}_{X,N_-^{(t)}}(X_i)\Big)\right] \\ &- \lambda(\pi_+^{(t)}+\pi_-^{(t)}-1) \end{aligned}$$

To find a maximum of $L(\theta^{(t)})$, we have :

$$\frac{\partial L_{\text{Lagrange}}(\theta^{(t)})}{\partial \pi_+^{(t)}} = 0 \quad \Rightarrow \quad \pi_+^{(t+1)} = \frac{1}{\lambda}\sum_{i=1}^{N} \rho_{i+}^{(t)}$$
$$\frac{\partial L_{\text{Lagrange}}(\theta^{(t)})}{\partial \pi_-^{(t)}} = 0 \quad \Rightarrow \quad \pi_-^{(t+1)} = \frac{1}{\lambda}\sum_{i=1}^{N} \rho_{i-}^{(t)}$$

Invoking the constraint $\pi_+^{(t)} + \pi_-^{(t)} = 1$, we can obtain that $\lambda = N$, then :

$$\begin{cases} \pi_+^{(t)} = \frac{\sum_{i=1}^{N} \rho_{i+}^{(t)}}{N} \\ \pi_-^{(t)} = \frac{\sum_{i=1}^{N} \rho_{i-}^{(t)}}{N} = 1 - \pi_+^{(t)} \end{cases} \tag{5.6}$$

It can be shown that Eq. 5.6 has the same form as Eq. 4.26. For the GMM-based BER estimation proposed in Chapter 4, we can start the next iteration while obtaining the new values of $\pi_+^{(t)}$ and $\pi_-^{(t)}$. However, for unsupervised BER estimation, there remain two pairs of parameters to be decided : $(N_+^{(t)}, h_{N_+}^{(t)}, N_-^{(t)}, h_{N_-}^{(t)})$, which depend on the outcome of the classification of subsets $C_+^{(t)}$ and $C_-^{(t)}$ and determine the conditional PDF estimates and the APPs in subsequent iterations. Therefore, the classification procedure must be carefully performed.

A simple method to update the two subsets consists in using Bayes' rule as follows :

$$\begin{cases} C_+^{(t)} = \{X_i : \rho_{i+}^{(t)} \geq \rho_{i-}^{(t)}\} = \{X_i : \pi_+^{(t)} \hat{f}^{b+}_{X,N_+^{(t)}}(X_i) \geq \pi_-^{(t)} \hat{f}^{b-}_{X,N_-^{(t)}}(X_i)\} \\ C_-^{(t)} = \bar{C}_+^{(t)} = \{X_i : \rho_{i+}^{(t)} < \rho_{i-}^{(t)}\} = \{X_i : \pi_+^{(t)} \hat{f}^{b+}_{X,N_+^{(t)}}(X_i) < \pi_-^{(t)} \hat{f}^{b-}_{X,N_-^{(t)}}(X_i)\} \end{cases} \tag{5.7}$$

However, this procedure could be quite "strict" that the received information may not be correctly reassigned to the adapted subset, i.e., a positive output $X_i$ corresponding to transmitted bit value -1 may not be exchanged from $C_+$ to $C_-$.

Thus, we introduce a Stochastic EM algorithm that randomly performs the classification of the soft outputs [SAIM11].

### 5.1.4 Stochastic step

In the Stochastic step of iteration $t$, for $i = 1, \ldots, N$, a uniform random variable, $U_i^{(t)} \in [0, 1]$, is generated. A random Bayes' rule is applied as follows :

$$\begin{cases} C_+^{(t)} = \{X_i : \rho_{i+}^{(t)} \geq U_i^{(t)}\} \\ C_-^{(t)} = \bar{C}_+^{(t)} = \{X_i : \rho_{i+}^{(t)} < U_i^{(t)}\} \end{cases} \tag{5.8}$$

The parameters $N_+^{(t)}$ and $N_-^{(t)}$ can be obtained by using Eq. 5.2. To compute the optimal smoothing parameters, we can :

– directly use the optimal values for Gaussian distribution followed by Eq. 3.27 if we know that the PDF is Gaussian ;
– iteratively use the exact expressions of Eq. 3.13 followed by Eq. 3.19 with an initial values given by Eq. 3.27 ;
– use the proposed methods in section 3.2.2.

### 5.1.5 Conclusion for SEM-based unsupervised BER estimation using Kernel method

After $T$ SEM iteration, the soft BER can be given by Eq. 3.41 taking into account the estimated parameters $\theta^{(T)} = (N_+^{(T)}, \pi_+^{(T)}, h_{N_+}^{(T)}, N_-^{(T)}, \pi_-^{(T)}, h_{N_-}^{(T)})$.

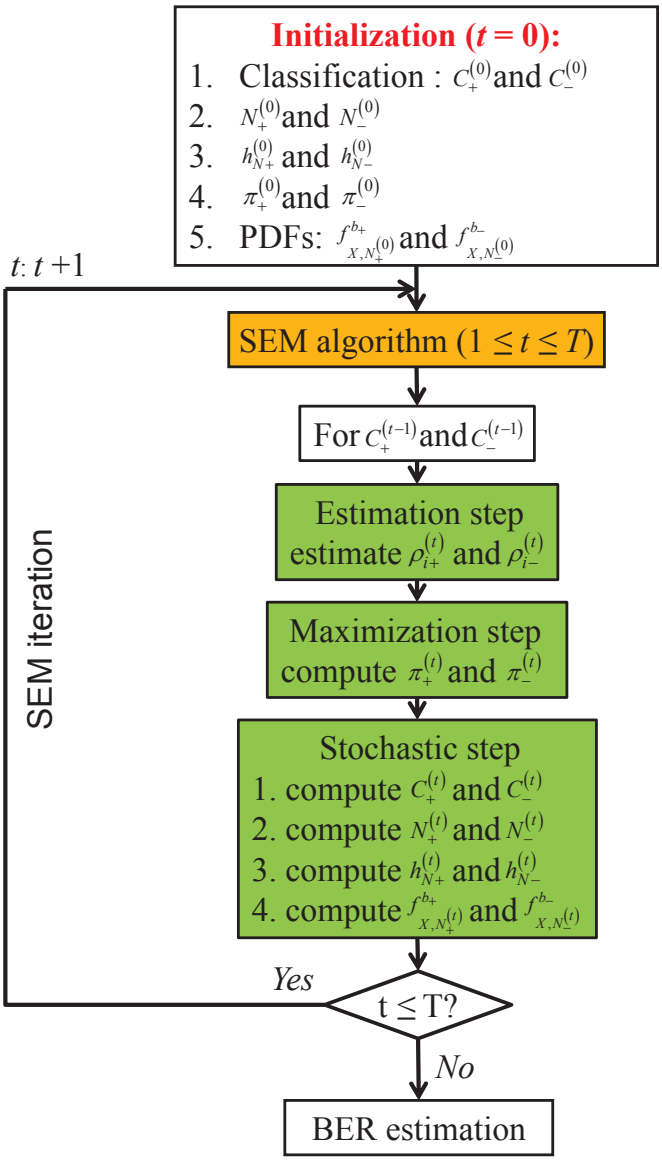


***Figure 5.1*** — Summary of the SEM-based unsupervised BER estimation using Kernel method

Fig. 5.1 shows the flow chart of unsupervised BER estimation based on Stochastic Expectation-Maximization algorithm combined with Kernel method.

The value of maximum iteration number, $T$, should be carefully chosen :

- for low SNR, erroneous classifications (according to the signs of the soft outputs) occur frequently. Thus, the maximum number of SEM iteration must be big enough ;
- for high SNR, the initial condition of classification could be very good due to improvement of SNR. In this situation, a small number of iteration is sufficient to have an exact classification.

Therefore, to reduce the simulation run time, the values of $T$ can be in different scale : for low SNR, we use a big value, for high SNR, we use a small value.

## 5.2 Unsupervised BER estimation based on Stochastic Expectation Maximization algorithm using Gaussian Mixture Model

The unsupervised BER estimation can also be performed by using the Gaussian Mixture Model.

We start with the initialization step. The initial classification is always the first one and the cardinalities are then computed. Instead of finding the smoothing parameters, for a given number of Gaussians $K$, we should initialize the $\theta^{(0)}_{GMM} = (\alpha^{(0)}_k, \mu^{(0)}_k, \sigma^{2(0)}_k)_{1\leq k\leq K}$ , which is computed by using EM algorithm through $T^{'}$ iterations. We denote $\theta^{(0)} = (\pi^{(0)}_+, \theta^{(0)}_{+,GMM}, \pi^{(0)}_-, \theta^{(0)}_{-,GMM})$ the unknown parameters.

In the Estimation step, we estimate the APPs for every data samples $(X_i)_{1\leq i\leq N}$ using Eq. 5.4. The conditional PDF estimates for $C^{(t)}_+$ and $C^{(t)}_-$ are given by :

$$\begin{cases} \hat{f}^{b_+}_{X,N_+} = \sum_{k=1}^{K} \alpha^+_k f^+_k(x; \mu^+_k, \sigma^{+2}_k) \\ f^{b_-}_{X,N_-} = \sum_{k=1}^{K} \alpha^-_k f^-_k(x; \mu^-_k, \sigma^{-2}_k) \end{cases} \tag{5.9}$$

The Maximization step is the same as the one for SEM algorithm combined with Kernel method. In the Stochastic step, at each iteration $t, t = 1, \ldots, T$, we introduce an inner-iteration $t^{'}, t^{'} = 1, \ldots, T^{'}$ to compute the GMM parameters $\theta^{(t)}_{GMM} = (\alpha^{(t)}_k, \mu^{(t)}_k, \sigma^{2(t)}_k)_{1\leq k\leq K}$. After $T$ iteration, the soft BER estimate can be obtained by Eq.4.30 taking into account the estimated parameters $\theta^{(T)} = (\pi^{(T)}_+, \theta^{(T)}_{+,GMM}, \pi^{(T)}_-, \theta^{(T)}_{-,GMM})$.

Fig 5.2 shows the flow chart of the SEM-based BER estimation using Gaussian Mixture Model.

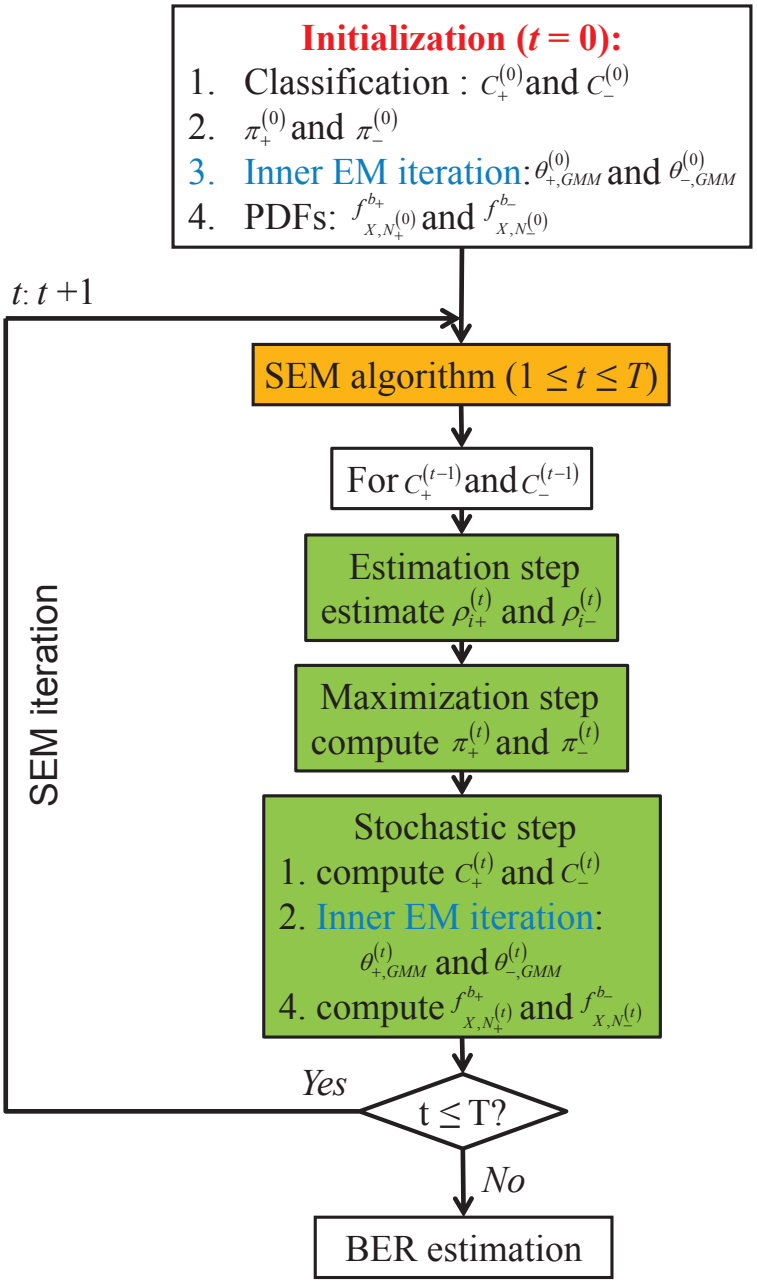


***Figure 5.2*** — Summary of the SEM-based unsupervised BER estimation using Gaussian Mixture Model

Compared with the SEM-based BER estimation algorithm using Kernel method, inner EM iterations must be added to the initialization step and the SEM iteration. Thus, the SEM-based BER estimation technique combined with GMM is more complex than the one with Kernel method.

## 5.3 Simulation results

To evaluate the performance of the proposed unsupervised SEM-based soft BER estimation using Kernel method and Gaussian Mixture Model. We have considered the same frameworks given in Chapter 3 and 4 :

- BPSK sequence over AWGN channel ;
- CDMA system with standard receiver ;
- Turbo coding system ;
- LDPC coding system.

### 5.3.1 Sequence of BPSK symbol over AWGN channel

First, we shall report the performance of SEM-based PDF estimation. In all simulations, we consider $T = 6$ iterations for the SEM-based parameter estimation while at each iteration, $t = 1, \ldots, T$, we compute the optimal smoothing parameters $h_{N_+}^{(t)}$ and $h_{N_-}^{(t)}$ for the current $C_+$ and $C_-$ by using the proposed Newton's method.

We consider the simplest case of uniform distributed probability BPSK bits that the information bits are equiprobably generated. The number of data samples (soft observations) is $N = 1000$ outputs. In Fig 5.3, we report the histogram and the estimated conditional PDFs using the obtained parameters at the last iteration $T = 6$ for $SNR = 10\ dB$.

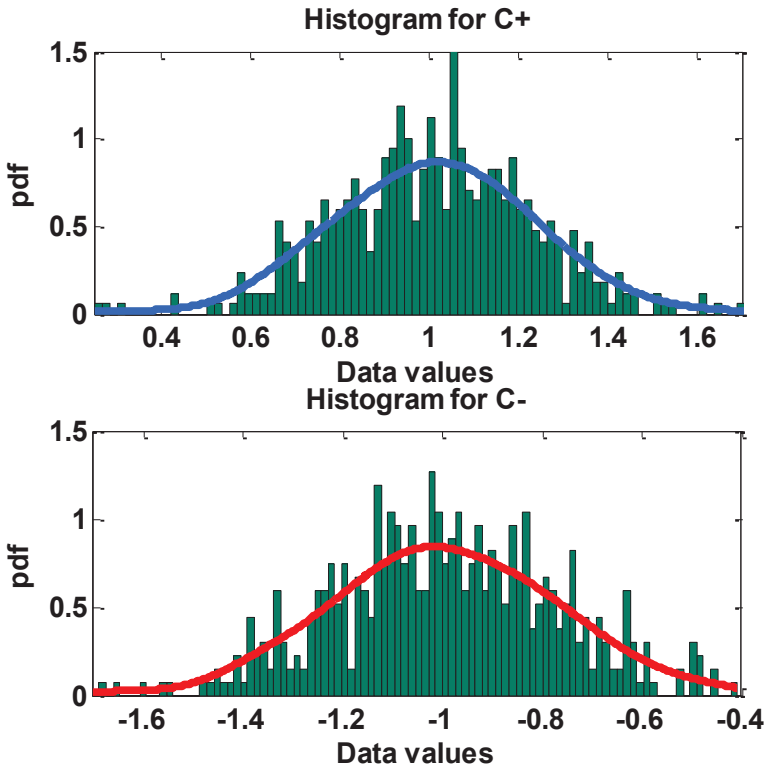


***Figure 5.3*** — Histograms and estimated conditional PDFs (BPSK symbols, $N = 1000$) for uniform sources and $SNR = 10\ dB$

We can observe that the estimated PDFs are very close to the true Gaussian PDFs.

Secondly, we turn to the case when the transmitted data bits are not equiprobable. Let $\pi_+$ and $\pi_-$ be the a priori probabilities, we consider the scenario where $\pi_+ = 0.7$ and $\pi_- = 0.3$. As in the previous simulation of equiprobable transmitted bits, $N = 1000$ information bits are used. Fig 5.4 shows the histograms and the obtained conditional PDFs for $SNR = 10\ dB$.

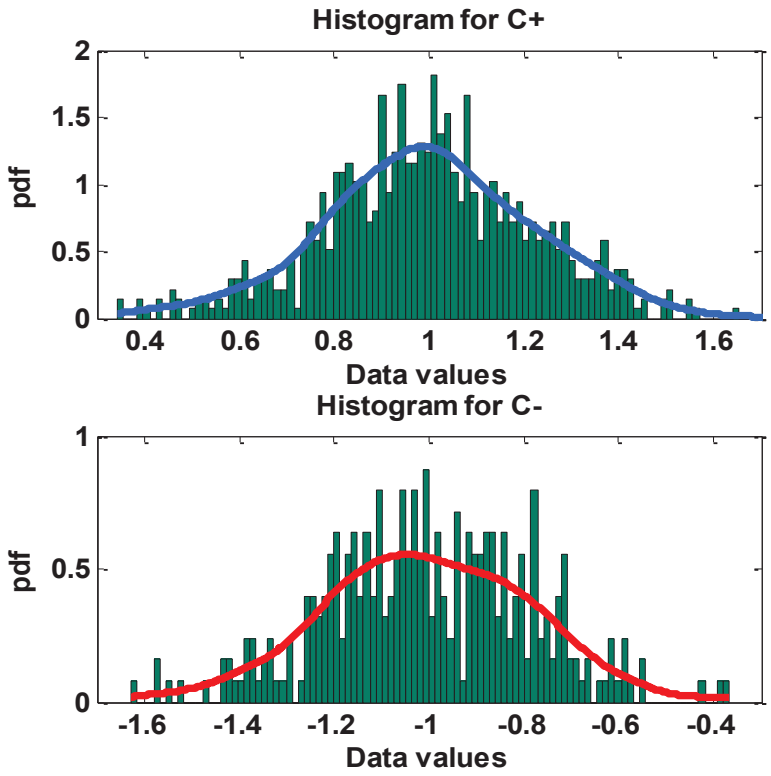


***Figure 5.4*** — Histograms and estimated conditional PDFs (BPSK symbols, $N = 1000$) for non-uniform sources ($\pi_+ = 0.7$ and $\pi_- = 0.3$) and $SNR = 10\ dB$

We notice that the estimated conditional PDFs for $C_+^{(6)}$ and $C_-^{(6)}$ are still close to the theoretical values even if the a priori probabilities are not equal.

Also, in the case of non-uniform sources, we report in Tab. 5.1 the estimated $N_+^{(t)}$, $N_-^{(t)}$, $\pi_+^{(t)}$ and $\pi_-^{(t)}$ at different iteration $t$ ($t = 1, \ldots, 6$) for $SNR = 0\ dB$ and $10\ dB$. In this simulation, according to the values of a priori probabilities ($\pi_+ = 0.7$ and $\pi_- = 0.3$), 679 bits of +1 and 321 bits of -1 are generated. This means that when the Kernel estimator knows the transmitted bits, $\pi_+ = 0.679$ and $\pi_- = 0.321$ are used as the estimates of the a priori probabilities for the non-parametric estimation.

| Iteration $t$ | | $t=1$ | $t=2$ | $t=3$ | $t=4$ | $t=5$ | $t=6$ |
|---|---|---|---|---|---|---|---|
| $SNR = 0\ dB$ | $N_+^{(t)}$ | 664 | 666 | 681 | 680 | 677 | 680 |
| | $N_-^{(t)}$ | 336 | 334 | 319 | 320 | 323 | 320 |
| | $\pi_+^{(t)}$ | 0.670 | 0.666 | 0.667 | 0.679 | 0.680 | 0.677 |
| | $\pi_+^{(t)}$ | 0.330 | 0.334 | 0.333 | 0.321 | 0.320 | 0.322 |
| $SNR = 10\ dB$ | $N_+^{(t)}$ | 679 | 679 | 679 | 679 | 679 | 679 |
| | $N_-^{(t)}$ | 321 | 321 | 321 | 321 | 321 | 321 |
| | $\pi_+^{(t)}$ | 0.679 | 0.679 | 0.679 | 0.679 | 0.679 | 0.679 |
| | $\pi_+^{(t)}$ | 0.321 | 0.321 | 0.321 | 0.321 | 0.321 | 0.321 |

**TABLE *5.1*** — Estimated $N_+^{(t)}$, $N_-^{(t)}$, $\pi_+^{(t)}$ and $\pi_-^{(t)}$ for $SNR = 0\ dB$ and $10\ dB$ (BPSK smbols, $T = 6$, $N = 1000$, $\pi_+ = 0.7$ and $\pi_- = 0.3$)

It can be shown that :

- for low SNR (e.g., $SNR = 0\ dB$), the estimated a priori probabilities, $\pi_+^{(t)}$ and $\pi_-^{(t)}$, and the obtained cardinalities, $N_+^{(t)}$ and $N_-^{(t)}$, are changed with different SEM iteration $t$, since many errors are generated while classifying the soft outputs ;
- for high SNR (e.g., $SNR = 10\ dB$), the values of $N_+^{(t)}$, $N_-^{(t)}$, $\pi_+^{(t)}$ and $\pi_-^{(t)}$ are constant and equal to their true values ($N_+ = 679$, $N_- = 321$, $\pi_+ = \frac{N_+}{N} = 0.679$ and $\pi_- = \frac{N_-}{N} = 0.321$), since very limited SNR can provide reliable classification.

This means that, for big values of SNR, the whole transmitted information bits were "known" at the receiver, then the unsupervised PDF estimation corresponds to the supervised non-parametric (Kernel) or semi-parametric (GMM) estimation technique. In this situation, we only need to compute the smoothing parameters or the GMM parameters and the SEM algorithm is not required.

Moreover, for small values of SNR, the number of SEM iterations is not the better the bigger it gets due to the use of random classification Bayesian rule in stochastic step. Note that for $SNR = 0\ dB$, $\pi_+^{(t)}$ and $\pi_-^{(t)}$ are equal to the true values ($\pi_+ = 0.679$ and $\pi_- = 0.321$) when $t = 4$.

In Fig. 5.5, we report the behavior of the proposed SEM-based BER estimation using Kernel method for non-uniform sources ($\pi_+ = 0.7$ and $\pi_- = 0.3$) and $T = 6$. The optimal smoothing parameters of Kernel estimator are computed by using Newton's method.

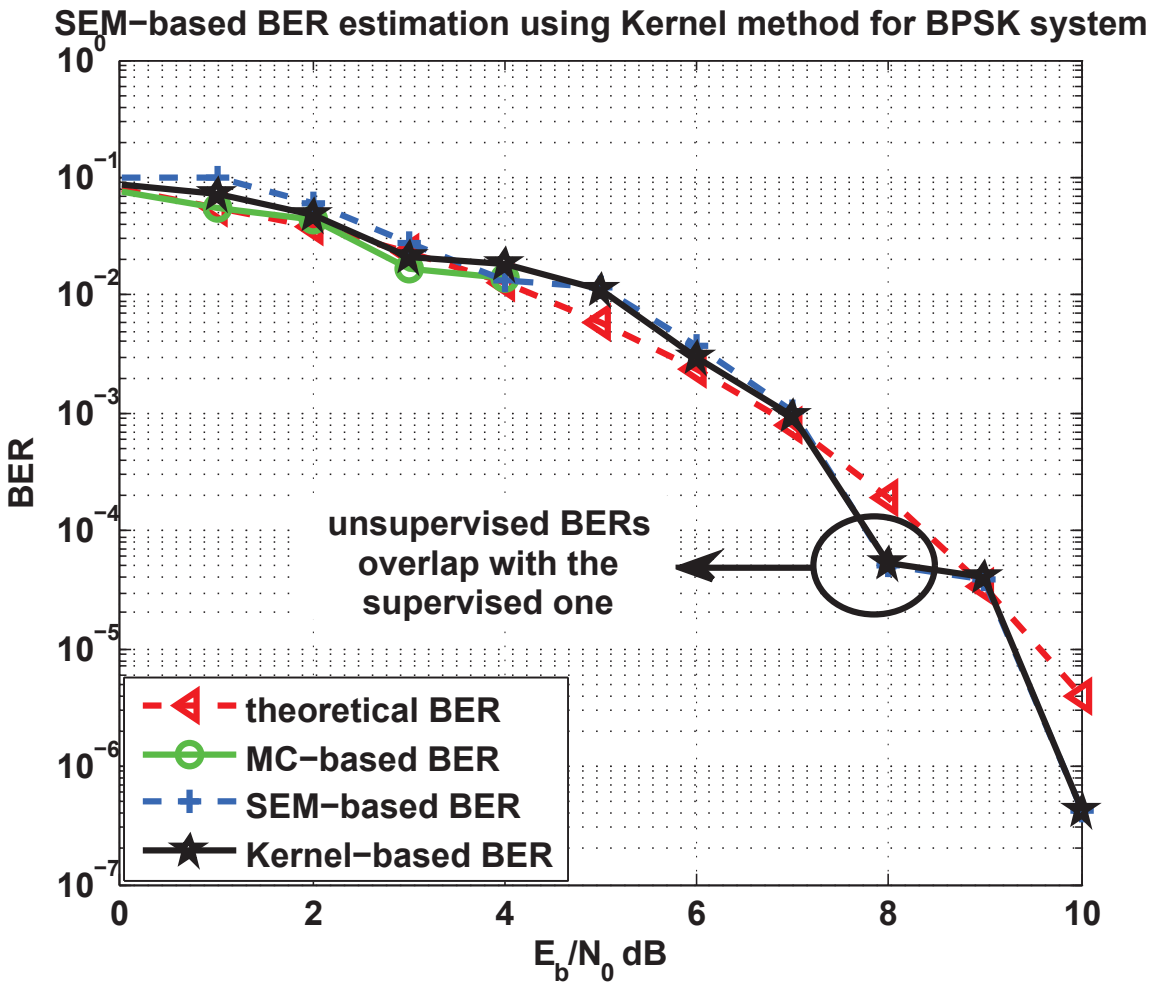


***Figure 5.5*** — Behavior of the SEM-based BER estimation using Kernel method for non-uniform sources (BPSK symbols, $N = 1000$, $\pi_+ = 0.7$ and $\pi_- = 0.3$

We observe that the proposed SEM algorithm provides quasi-reliable BER estimates for SNR values up to 10 $dB$. For simplicity, we only count 10 errors for the classical Monte-Carlo simulation at each value of SNR. We observe that the MC technique fails to obtain BER estimates and stops at $SNR = 4\ dB$.

Furthermore, we notice that the values of SEM-based BER estimates are very close to the supervised Kernel-based one, especially for high SNR, i.e., for $SNR = 10\ dB$, the two techniques have the same value of BER estimates. This is in line with the results shown in Tab. 5.1. In fact, for high SNR, the SEM classification would be the same as the one of supervised Kernel method, thus, the unsupervised SEM algorithm is equivalent to the supervised technique.

However, we can find that the obtained BER estimates are not quite smooth. In Fig. 5.6, we report the behavior of the SEM-based BER estimation for the same non-uniform scenario but with 15,000 observations.

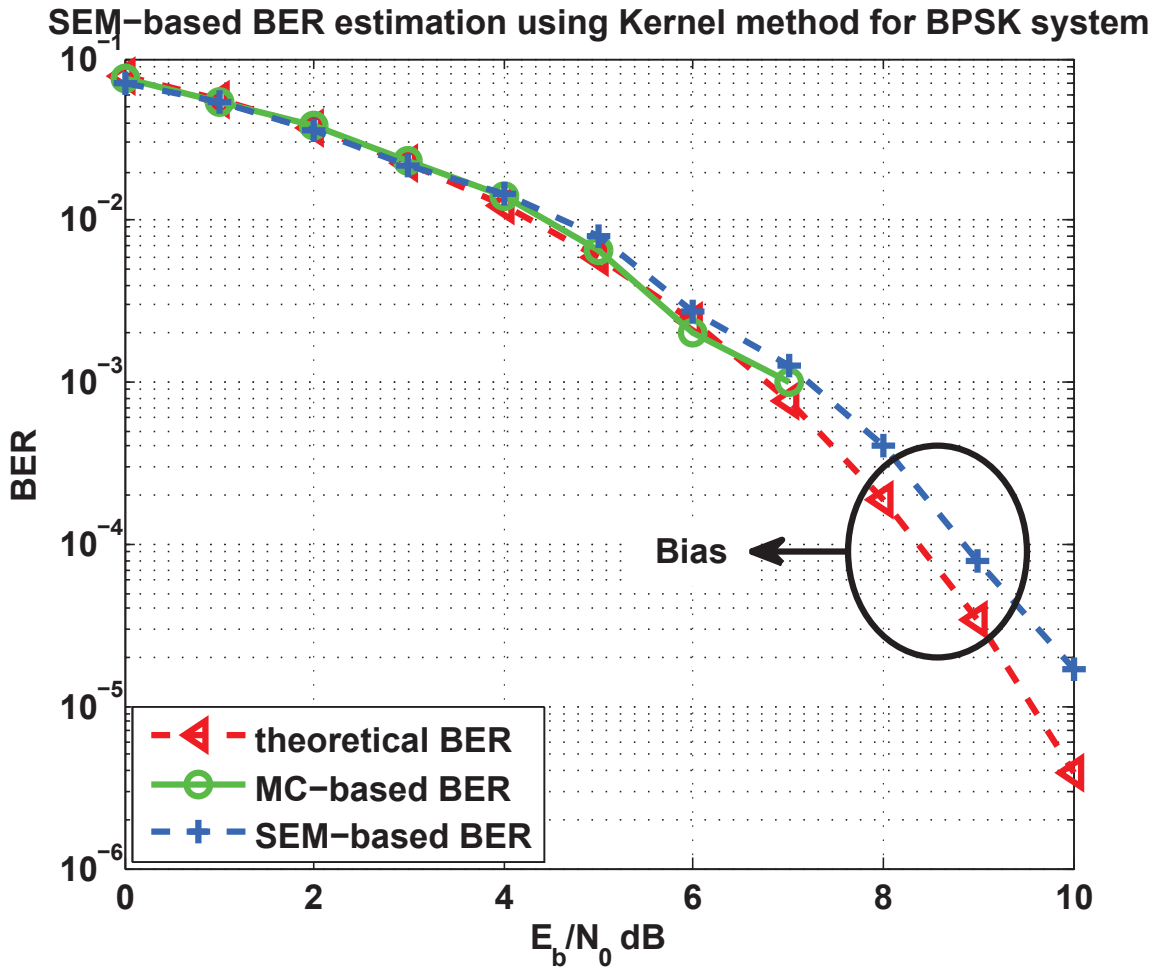


***Figure 5.6*** — Behavior of the SEM-based BER estimation using Kernel method for non-uniform sources (BPSK symbols, $N = 15,000$, $\pi_+ = 0.7$ and $\pi_- = 0.3$

It can be shown that the smoothness of the BER estimates is much improved. However, significant biases can be found for high SNR. In fact, as shown in 3.11, these biases are caused by the "nature" of Kernel-based estimator — compared with the GMM-based estimator, the Kernel-based one cannot provide accurate PDF estimate in the region of high SNR.

### 5.3.2 CDMA system with standard receiver

We consider the same CDMA system with two users and standard receiver as in section 3.2.2.1. We focus on the scenario where $\pi_+ = 0.7$ and $\pi_- = 0.3$. The two users have equal powers $A_1 = A_2 = 1$. The maximum iteration number $T$ is set to 6 and 1000 data samples are used.

First, we report in Fig 5.7 and Fig 5.8 the conditional PDFs obtained by using unsupervised Kernel method for $SNR = 0\ dB$ and $10\ dB$.

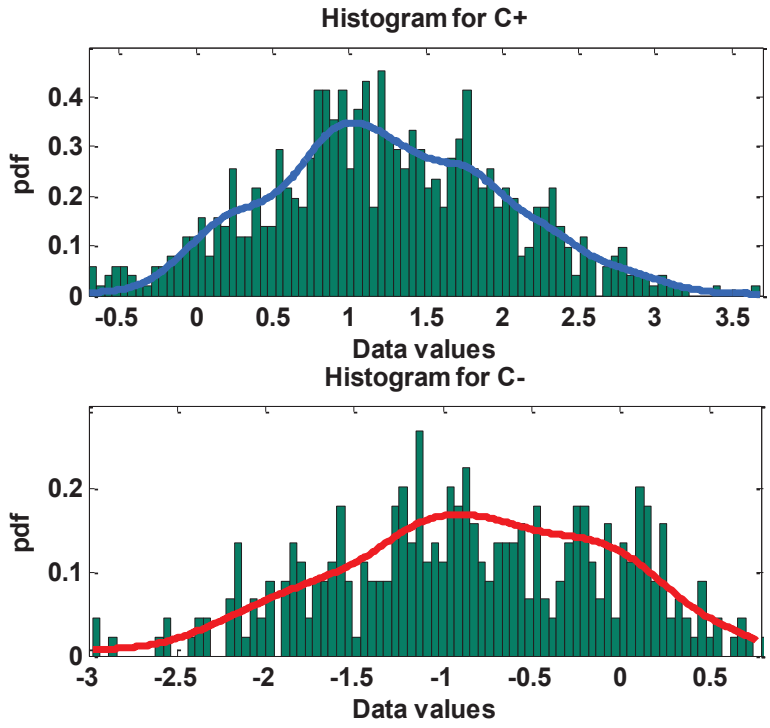


***Figure 5.7*** — Histogram and estimated PDFs for CDMA system ($SNR = 0\ dB$, $N = 1000$, $\pi_+ = 0.7$ and $\pi_- = 0.3$)

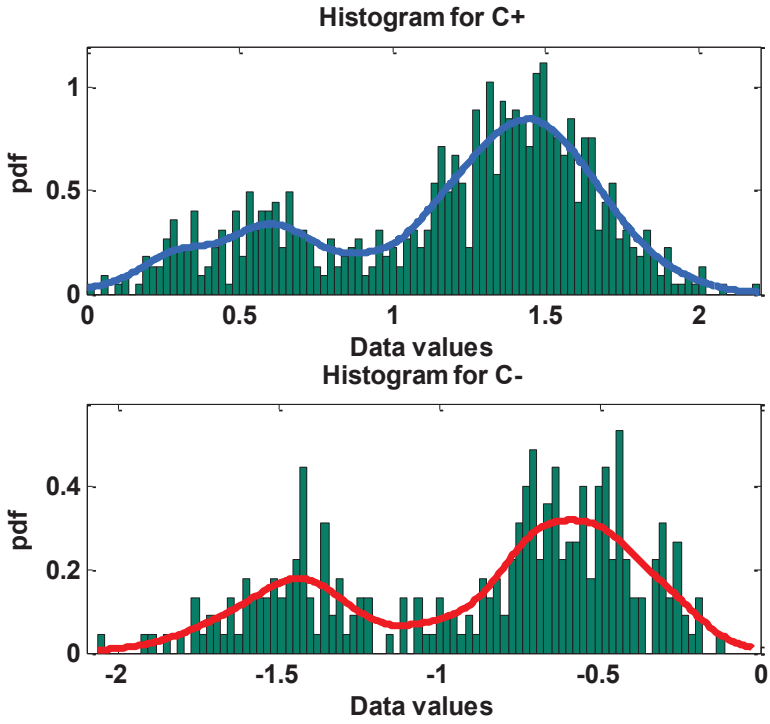


***Figure 5.8*** — Histogram and estimated PDFs for CDMA system ($SNR = 10\ dB$, $N = 1000$, $\pi_+ = 0.7$ and $\pi_- = 0.3$)

It can be shown that :

- for low SNR ($SNR = 0\ dB$), we observe that the conditional PDFs are correctly estimated according to the Monte-Carlo-aided histogram but with some oscillations. In fact, in consideration of the cross-correlation $\rho = 0.4286$ (cf. section 3.2.2.1), the Multiple Access Interferences (MAI) will be terrible and the soft outputs of the CDMA receiver will be corrupted by severe MAI and thermal noise. Therefore, a large number of data samples are required to improve the smoothness ;
- for high SNR ($SNR = 10\ dB$), no severe oscillations are found since the variance of the MAI plus noise is reduced.

Secondly, we report in Fig 5.9 the behavior of SEM-based BER estimation using either Kernel method or Gaussian Mixture Model. For GMM-based estimator, the maximum number of EM iterations is set to 5.

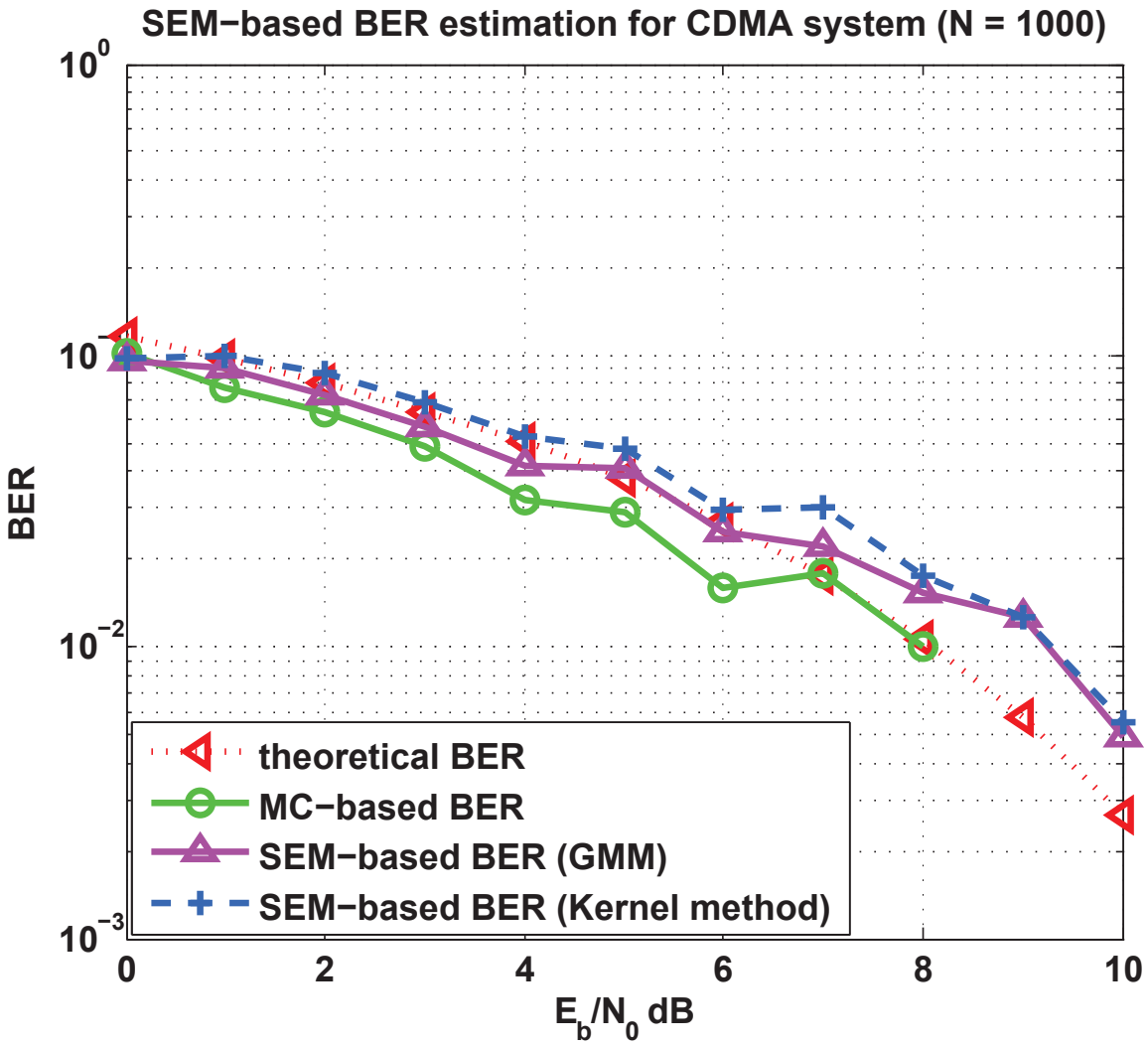


***Figure 5.9*** — Behavior of the SEM-based BER estimation using GMM and Kernel method for non-uniform sources (CDMA system, $N = 1000$, $\pi_+ = 0.7$ and $\pi_- = 0.3$)

We observe that the unsupervised SEM algorithm combined with either Kernel method or Gaussian Mixture Model provides reliable BER estimates. For high SNR, the Monte-Carlo technique fails to do so due to the very limited number of observations. Moreover, the simulation run time is quite fast : for $SNR = 10\ dB$, SEM algorithm combined with GMM only takes 20 $s$ to obtain the BER estimate ; the one with Kernel method takes 7 $min$ because the Newton's method need long computing time to optimize the smoothing parameters.

### 5.3.3 Turbo coding and LDPC coding systems

We shall consider the Turbo codes-based system. As in Chapter 3 and 4, we have taken 300,000 samples in 600 frames. For all the simulations of this subsection, we consider the case of uniform sources (APPs $\pi_+ = \pi_- = 0.5$) and $T = 6$ SEM iterations. The SEM algorithm is combined with the non-parametric Kernel method. For simplicity, we use the initial values of the smoothing parameters and did not perform the bandwidth optimization. In Fig 5.10, we report the behavior of the unsupervised BER estimates.

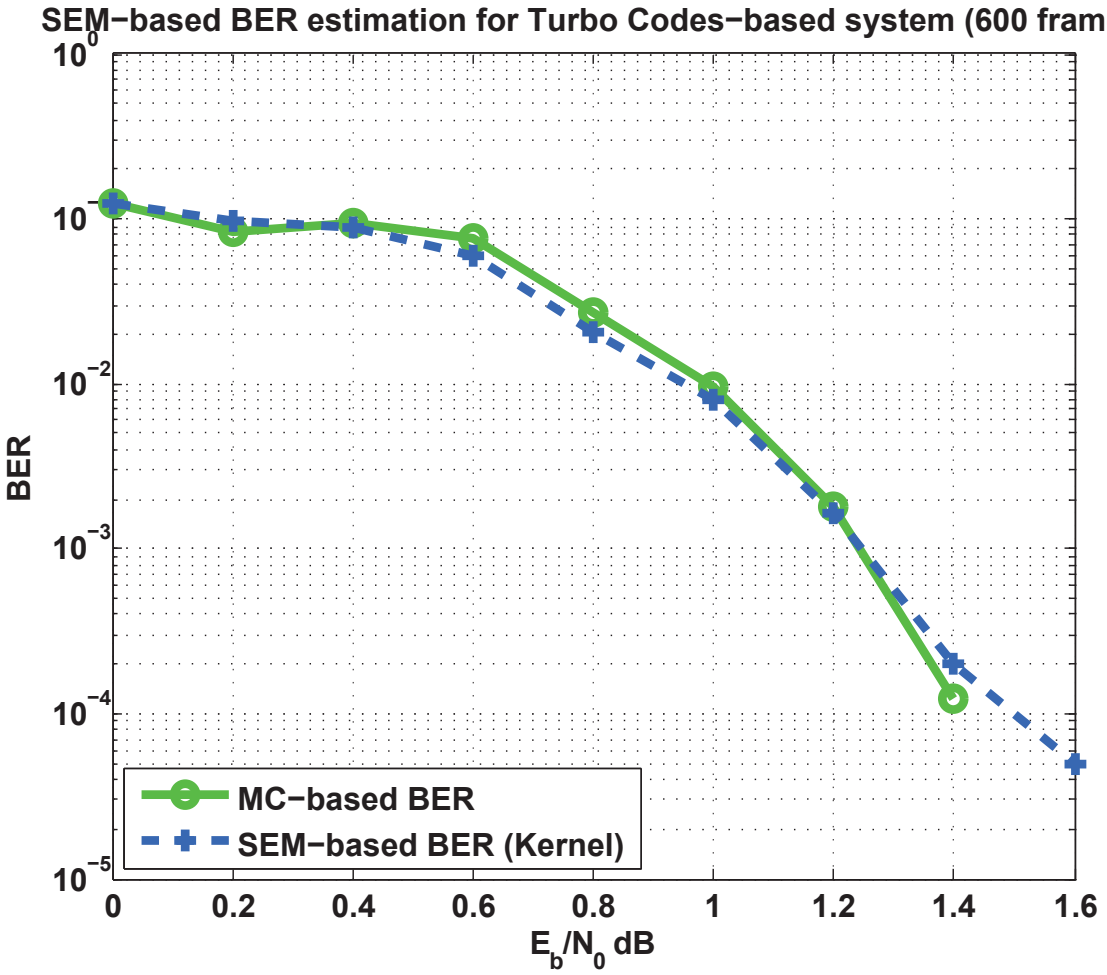


***Figure 5.10*** — Behavior of the SEM-based BER estimation using Kernel method for non-uniform sources (Turbo coding system, 600 frames, $\pi_+ = \pi_- = 0.5$)

For Monte-Carlo simulation, we count 100 errors for each value of SNR. The MC-based BER estimation stops at $SNR = 1.2\ dB$, whereas the unsupervised SEM technique provides reliable BER estimates for SNR values up to 1.6 $dB$.

Then, we focus on the 1/2 rate QC-LDPC system as considered in Chapter 3 and 4. The parity check matrix G is generated as a dimension of 635 $\times$ 1270. The SEM algorithm is combined with the Kernel method.

First, we have used 500 frames for BER estimation where SNR values up to $SNR = 2.5\ dB$, each frame contains 635 random bits as transmitted information bits, which means $500 \times 635 = 317,500$ samples are used. The proposed unsupervised technique can provide reliable BER estimates but the simulation run time is terrible. In Tab. 5.2, we report the simulation run time for $SNR = 2.5\ dB$. The program takes about 24 hours to finish the SEM algorithm because of the very large number of data samples.

| | coding/decoding | SEM algorithm (without bandwidth optimization) |
|---|---|---|
| Elapsed time | 5 $min$ 30 $s$ | 23 $h$ 38 $min$ 10 $s$ |

**Table 5.2** — Simulation run time of SEM-based BER estimation for QC-LDPC system (500 frames, $SNR = 2.5\ dB$)

In order to reduce the program complexity and to decrease simulation run time, we have used the proposed modified method (cf. section 3.3.4) — we ignore the soft observations equal (or near) to the true bit values ($-1$ and $+1$) and only focus on the minority since the BER is determined by the "error area" which is far from $+1$ and $-1$. The predefined threshold $\epsilon$ is set to 0.9995, this means that we only consider the observations which are less than $+0.9995$ for the positive soft outputs and bigger than $-0.9995$ for the negative outputs. This modified dataset shall only be used for high SNR (e.g., $SNR \geq 2\ dB$), for small values of SNR, we should use the entire data samples to avoid losing information.

We have tested the unsupervised SEM-based BER estimation by using Kernel method along with the modified dataset in single trial. For $SNR = 2.5\ dB$, with 317,500 data samples, 158,230 soft observations which belong to $C_+$ are generated, and only 1374 observations are less than $+0.9995$. This means that, by using the modified unsupervised Kernel method, we only need to take into account 1374 samples to estimate the $f_X^{b+}$. Thus, the simulation run time could be much reduced.

In Tab. 5.3 and Fig. 5.11, we report the simulation run time and the BER estimates for the modified SEM-based technique.

| | coding/decoding | SEM algorithm (without bandwidth optimization) |
|---|---|---|
| Elapsed time | 5 *min* 30 *s* | 9 *s* |

**Table 5.3** — Simulation run time of SEM-based BER estimation for QC-LDPC system (500 frames, $SNR = 2.5\ dB$)

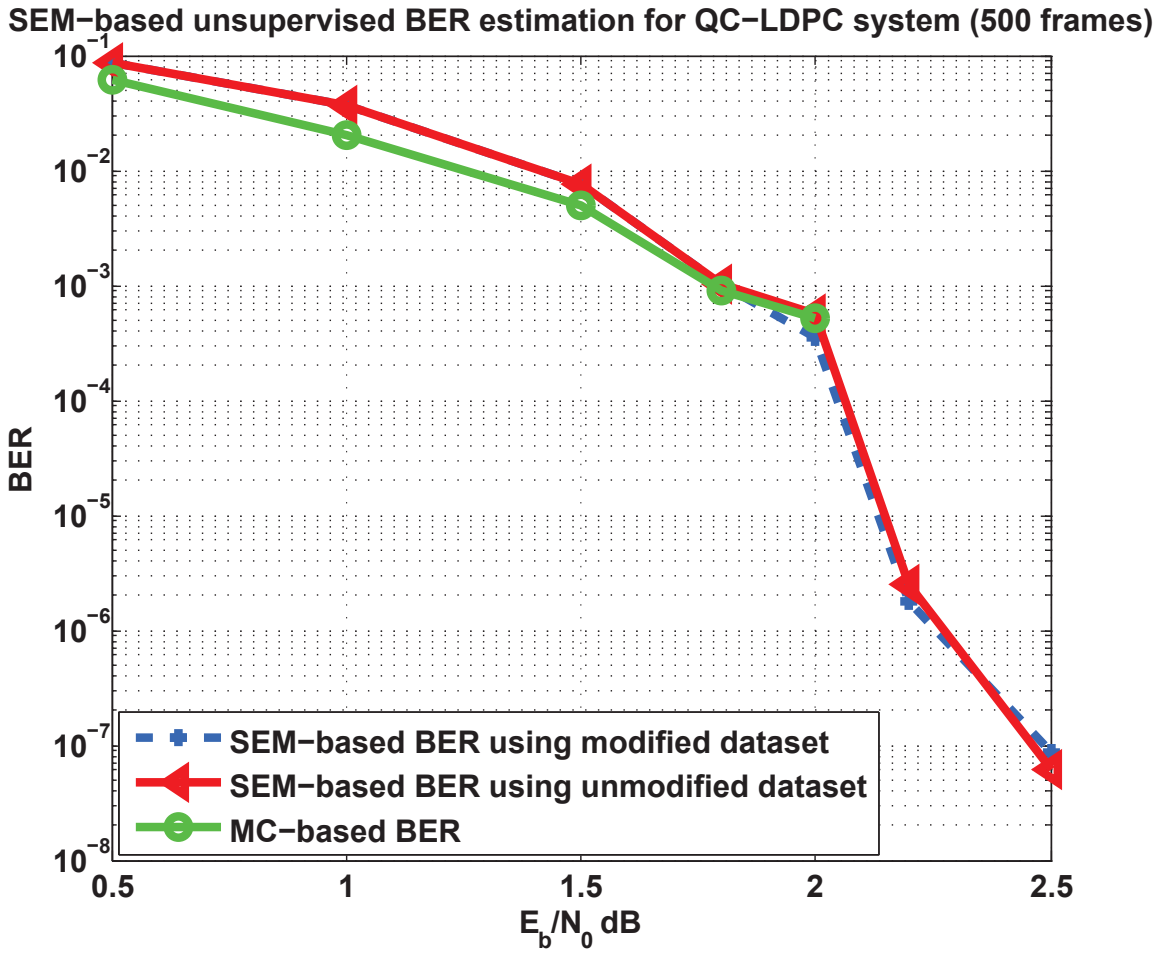


***Figure 5.11*** — Behavior of the modified SEM-based BER estimation using Kernel method for uniform sources (QC-LDPC system, 500 frames, $\pi_+ = \pi_- = 0.5$)

It can be shown that, in the region of high SNR ($SNR \geq 2\ dB$), the SEM-based BER estimates combined with Kernel method and modified dataset are close to the one with original dataset. Compared to the simulation run time presented in Tab. 5.2, only 9 seconds are sufficient to finish the modified SEM algorithm for $SNR = 2.5\ dB$. For low SNR, the simulation time would be longer since more samples which are far from the true bit values $+1$ and $-1$ should be taken into account.

## 5.4 Conclusion

In this chapter, we have presented an unsupervised BER estimation technique based on the Stochastic Expectation-Maximization algorithm combined with the proposed non-parametric (Kernel method) or semi-parametric (Gaussian Mixture Model) PDF estimation method. We have tested and reported the behavior of PDF and BER estimates for different digital communication system. The proposed unsupervised BER estimation technique provides reliable BER estimates even for high SNR independently of the distribution of data bits.

The key of the proposed unsupervised technique consists in estimating the a priori probabilities in an iterative way. Once we obtain the estimates of the classification information, we use the Kernel method or the GM model to estimate the PDFs and therefore the BER. The initial classification is done by taking into account the signs of the soft observations. For high SNR, the initial a priori probabilities could be the same to the true values, then the iterative SEM algorithm is not required.

Compared to the classical Monte-Carlo simulation and other presented BER estimation techniques, the unsupervised BER estimation does not require the knowledge of the transmitted information bit values. Moreover, we do not need to know the transmitter scheme, channel condition, and reception technique. These characteristics help the unsupervised estimator to adapt to practical situations.

The SEM algorithm may take very long simulation run time when the size of dataset is large. For this reason, we have proposed to use modified dataset that only the soft outputs around the tails of PDFs ("error area") are taken into account. By doing so, the simulation run time could be much more reduced by using the same data frames. However, using modified dataset might cause unreliable estimation since the variance is not minimum, as presented in section 3.3.4.

# Conclusion and perspectives

In many practical communication systems, a real-time and on-line Bit Error Rate estimation would be of great interest to perform system-level functions, e.g., power control, resource allocation, link adaptation, etc. Under this framework, several issues must be taken into account :

- the estimator should be unsupervised, or blind, which means that no information about transmitted data bits is available ;
- the estimation should be performed with a very limited number of data samples ;
- the performance and reliability of BER estimates should be immune to either the transceiver scheme/techniques or to the channel model.

Classically, the Monte-Carlo technique is used to estimate the BER of a digital communication system. This BER estimate approaches the true BER but the number of transmitted bits becomes very high as long as the BER becomes very low. To solve this issue, Monte-Carlo techniques have been modified, like for example : importance sampling, tail extrapolation and quasi-analytical method. Unfortunately, all these methods assume the knowledge of noise statistics.

Therefore, we have studied in this manuscript, soft BER estimation techniques by the mean of probability density function (PDF). These PDFs are estimated from soft channel observations, without any knowledge of the noise statistics.

In Chapter 2, we have introduced the non-parametric and semi-parametric methods to estimate the PDF of soft observations. The non-parametric Kernel method and the semi-parametric Gaussian Mixture Model have been used for supervised BER estimation.

In Chapter 3, we have shown that the performance of Kernel-based PDF estimation strongly depends on the smoothing parameter (the bandwidth). We have given the expression of the optimal bandwidth when the PDF to be estimated is Gaussian. When the PDF to be estimated is not Gaussian, the optimal value should be computed using the Maximum Likelihood criterion. Then we have tested the Kernel-based technique for different digital systems : *i*) BPSK signal over AWGN and Rayleigh channels ; *ii*) CDMA system ; *iii*) turbo coding system and *iv*) QC-LDPC coding system. It can be shown that the proposed Kernel method provides reliable PDF and BER estimates

using only a small number of soft observations compared to the MC case.

In Chapter 4, we have presented the Gaussian Mixture Model-based BER estimation technique using Expectation-Maximization (EM) method. The GMM-based estimator depends on several parameters, such as the number of Gaussian components $K$ and the maximum EM-iteration number $T$. The value of $K$ could be selected by using the mutual information to evaluate the independence of the mixture. The value of $T$ is determined by some initial values which should be carefully selected. In our works, we have proposed to use the K-means algorithm to initialize them. The GMM method has been simulated under the same conditions as in Chapter 3. It can be shown that, by carefully choosing the number of Gaussians and the maximum number of EM iterations, the performance of BER estimation can be as good as the Kernel-based one, in average. Moreover, the variance of the estimation is better.

The aim of the proposed supervised Kernel-based and GMM-based BER estimators is to smoothen the MC-based PDFs (histograms) and minimize the distortions due to the lack of data samples. Many advantages have been found :

- only a small number of soft observations is required, which means that the proposed methods are well suited for on-line BER estimation ;
- compared with the classical and modified Monte-Carlo methods which fail to perform BER estimation for high SNR when the number of samples is not high enough, the proposed estimators provide reliable estimators ;
- the transceiver scheme and channel model are not required, which means that the proposed estimators can be used for any digital communication systems ; moreover, many types of soft observations could be applied, e.g., for BPSK symbols, the soft outputs are composed of the transmitted data and the noise, whereas the soft information of Turbo coding and LDPC coding systems is respectively the LLR and the a posteriori probabilities.

In Chapter 5, we have presented the unsupervised BER estimation techniques. As the knowledge of a priori probabilities of transmitted information bits and the classification of receiver's soft outputs are required, we have introduced an iterative Stochastic Expectation-Maximization algorithm to iteratively compute these parameters. Combined with the Kernel and GMM methods, we have analyzed the behavior of the unsupervised BER estimator under the same conditions as in Chapter 3 and 4. It was shown that, with a suitable number of SEM iterations, the unsupervised estimator provides similar BER estimates as the supervised BER results estimated by using the corresponding Kernel or GMM method. This is true even in the high SNR regime or when using uniform or non-uniform sources. By taking into account the above advantages of the Kernel-based and GMM-based methods, the unsupervised estimators are able to meet the requirements of on-line and real-time BER estimation.

A first perspective could be to implement an on-line estimation of the BER, where the estimation is updated at each sample, especially with the GMM-based estimator which performs quite well, even if the number of samples is very low compared to the targeted BER.

As we only focused on the soft observations of binary information, we shall also investigate the extension to the estimation of the symbol error rate, where the symbols are taken in a given QAM or PSK constellation.

# Appendix

## Appendix A

### LDPC codes

LDPC codes were firstly developed by Gallager in 1963 [Gal62], but soon forgotten until Gallager's work was discovered in 1996 [MN96] by MacKay.

#### (1) Parity-check matrix and Tanner graph

LDPC codes are defined by a sparse parity-check matrix, which is often randomly generated and represented by a bipartite graph called Tanner graph [Tan81].

Fig. A.1 shows the Tanner graph of the sparse parity-check matrix given by Eq. A.1.

$$H = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \tag{A.1}$$

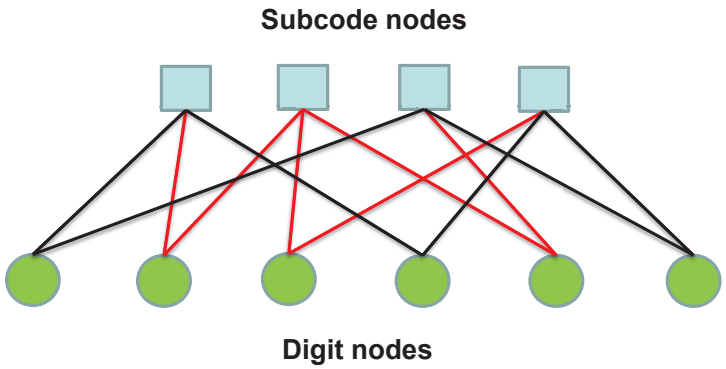


***Figure A.1*** — Tanner graph for the matrix of Eq. A.1

Tanner graph is partitioned into subcode nodes and digit nodes. For linear block codes, the subcode nodes denote the rows of the matrix $H$. The digit nodes represent the columns of the matrix $H$. An edge connects a subcode node to a digit node if a nonzero entry exists in the intersection of the corresponding row and column.

The girth of a Tanner graph is the length of the shortest cycle in the graph. In Fig. A.1, the black lines represent the shortest loop. Short cycles degrade the performance of LDPC decoders, because they affect the independence assumption (decoder side) of the extrinsic information exchanged in the iterative decoding [LFKL00]. Hence, LDPC codes with large girth are desired.

In our works, a Tanner QC-LDPC code [MYK05] without 4-cycles was used.

**(2) LDPC encoding**

Given a codeword $\mu$ and an $M$ *by* $N(N > M)$ parity check matrix $H$, we have :

$$\mu \cdot H^T = 0 \tag{A.2}$$

Assume that the message bits, $s$, are located at the end of the codeword and the check bits, $c$, occupy the beginning of the codeword, we have :

$$\mu = [c|s] \tag{A.3}$$

Also let :

$$H = [A|B] \tag{A.4}$$

where $A$ is an $M$ *by* $M$ matrix and $B$ is an $M$ *by* $N - M$ matrix.

If $A$ is an invertible matrix, the generator matrix $G$ can be given as :

$$G = \left[ [A^{-1}B]^T_{(N-M)\times M} I_{M\times M} \right] \tag{A.5}$$

Thus, we get :

$$c = A^{-1}Bs \tag{A.6}$$

Eq. A.6 can be used to compute the check bits as long as $A$ is non-singular.

**(3) LDPC decoding**

We used the Sum-Product method for LDPC decoding. This method is based on the Belief Propagation (BP) algorithm.

Let $\{r\}$ be the received signals and $X = \{X_1, \ldots X_j, \ldots, X_N\} \in \{0, 1\}$ be the codeword corresponding to the digit nodes of Tanner graph. Let $S$ be the state when $X$ satisfy all the parity-check constraints. If :

$$\frac{P(X_j = 0|\{r\}, S)}{P(X_j = 1|\{r\}, S)} \geq 1 \tag{A.7}$$

we can write :

$$\hat{X}_j = 0$$

Otherwise, we obtain :

$$\hat{X}_j = 1$$

Let $R_{i,j}^a$ be the conditional probability when the $i^{th}$ parity-check constraint is satisfied and the $j^{th}$ bit is equal to $a$ ($a = 0$ *or* $1$). We have :

$$\begin{cases} R_{i,j}^0 & = \frac{1}{2}\Big[1 + \prod_{j \in N(i) \backslash j}(1 - 2P_{j,i}^1)\Big] \\ R_{i,j}^1 & = 1 - R_{i,j}^0 \end{cases} \tag{A.8}$$

where N(i) is the set of information bits constrained by the $i^{th}$ subcode node.

If the transmitted bits are independent, we have :

$$\begin{aligned} P(S|X_j = 1, \{r\}) &= \prod_{k \in M(j) \backslash i} R_{k,j}^1 \\ P(S|X_j = 0, \{r\}) &= \prod_{k \in M(j) \backslash i} R_{k,j}^0 \end{aligned} \tag{A.9}$$

where M(j) is the set of checks for $X_j$.

Thus, we get :

$$\frac{P(X_j = 0|\{r\}, S)}{P(X_j = 1|\{r\}, S)} = \frac{1 - P_j}{P_j} \prod_{i=1}^{k} \frac{P(S|X_j = 0, \{r\})}{P(S|X_j = 1, \{r\})} \tag{A.10}$$

where $P_j$ is obtained by channel measurement at the receiver side.

Let $Q_{j,i}^a$ be the conditional probability that the $j^{th}$ bit is equal to a while using the information from subcode nodes except the $i^{th}$ node.

$$\begin{cases} Q_{j,i}^0 & = (1 - P_j) \prod_{k \in M(j) \backslash i} R_{i,j}^0 \\ Q_{j,i}^1 & = P_j \prod_{k \in M(j) \backslash i} R_{i,j}^1 \end{cases} \tag{A.11}$$

At each iteration, $R_{i,j}^a$ are firstly transmitted from the subcode nodes to the digit nodes, and then $Q_{j,i}^a$ are transferred from the digit nodes to the subcode nodes to update the values of $R_{i,j}^a$. If the iterative algorithm is convergent, we can finally obtain the two probabilities at iteration $t + 1$.

$$\begin{cases} Q_j^0(t+1) & = k_j(1 - P_j) \prod_{k \in M(j)} R_{i,j}^0(t+1) \\ Q_j^1(t+1) & = k_j P_j \prod_{k \in M(j)} R_{i,j}^1(t+1) \end{cases} \tag{A.12}$$

where $k_j$ is the normalization factor for $Q_j^0(t+1) + Q_j^1(t+1) = 1$.

Let $\lambda = \frac{Q_j^1(t+1)}{Q_j^0(t+1)}$, the decision can be given by :

$$\begin{cases} X_j = 1 & \text{if } \lambda \geq 1, \\ X_j = 0 & \text{otherwise.} \end{cases} \tag{A.13}$$

# Résumé de la thèse

Pour la majorité des systèmes de communications numériques, le taux d'erreurs binaires (BER : Bit Error Rate) est un paramètre clé. En général, le BER ne peut pas être calculé analytiquement et doit être estimé par les simulations de type Monte-Carlo (MC). Cependant, elles se révélent très côuteuses en nombre d'échantillons et en temps de simulation lorsque le BER est très faible. De plus, les données transmises doivent être connues par le récepteur, ceci signifie que ces méthodes ne sont pas applicables quand le taux d'erreurs doit être estimé de façon aveugle au niveau du récepteur.

Par conséquent, nous proposons de mettre en œuvre des techniques d'estimation de densités de probabilités (PDF : Probability Density Function) des observations souples en sortie du récepteur. Dans un premier temps, nous avons étudié l'estimation non-paramétrique appelée "méthode du noyau" (Kernel) pour estimer la PDF. Ensuite, le modèle de mélanges de gaussiennes (GMM : Gaussian Mixture Model) est utilisé.

## L'introduction de l'estimation souple du BER

Nous considerons $N$ échantillons des bits transmis I.I.D. (Indépendents et Identiquement Distribués), $(b_i)_{1\leq i\leq N}$, et les sorties souples du récepteur, $(X_i)_{1\leq i\leq N}$, qui ont la même PDF, $f_X(x)$. Alors le taux d'erreurs peut être exprimé par :

$$\begin{aligned} p_e &= P[\hat{b}_i \neq b_i] \\ &= P[X<0, b_i=+1] + P[X>0, b_i=-1] \\ &= P[X<0|b_i=+1]P[b_i=+1] + P[X>0|b_i=-1]P[b_i=-1] \end{aligned}$$

On note que $f_X^{b_+}(\cdot)$ et $f_X^{b_-}(\cdot)$ sont les PDFs des sorties souples $X$ dans les conditions $b_i=+1$ et $b_i=-1$. Le BER est :

$$p_e = P[b_i=+1]\int_{-\infty}^{0} f_X^{b_+}(x)dx + P[b_i=-1]\int_{0}^{+\infty} f_X^{b_-}(x)dx \tag{1}$$

En général, on note que $P[b_i=+1]=\pi_+$ et $P[b_i=-1]=\pi_-$, où $\pi_++\pi_-=1$. Alors la PDF des sorties souples est donnée par le mélange des deux PDFs conditionnelles :

$$f_X(x) = \pi_+ f_X^{b_+}(x) + \pi_- f_X^{b_-}(x) \tag{2}$$

Ce qui, en utilisant L'équation 1, donne :

$$p_e = \pi_+ \int_{-\infty}^{0} f_X^{b_+}(x)dx + \pi_- \int_{0}^{+\infty} f_X^{b_-}(x)dx \tag{3}$$

La figure 1 montre les PDFs conditionnelles des observations souples gaussiennes identiquement distribuées ($\pi_+ = \pi_- = 1/2$). Le BER est donné par la superficie de la zone hachurée sous les deux courbes.

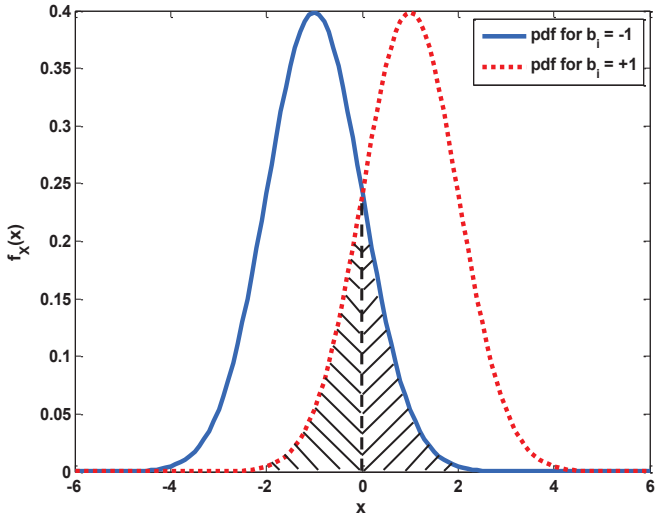


***Figure 1*** — Exemple des PDFs conditionnelles des observations souples

Par conséquent, l'estimation du BER est équivalent à l'estimation de la PDF conditionnelle des observations souples. Une fois que l'on obtient l'estimation de la PDF, le BER peut être analytiquement calculé à partir de l'équation 3.

Plusieurs techniques pourront être utilisées pour l'estimation de la PDF, telles que l'estimation paramétrique, la méthode non-paramétrique et la méthode semi-paramétrique. La méthode paramétrique n'est utilisée que pour calculer le BER théorique car il est souvent très difficile de trouver l'expression mathématique de la PDF inconnue pour le système pratique. Nous proposons d'utiliser la méthode non-paramétrique ou semi-paramétrique pour estimer la PDF des observations souples.

## L'estimation du BER par la méthode du noyau

La méthode du noyau est une technique non-paramétrique très souvent utilisée, généralisant astucieusement la méthode d'estimation par histogramme. Pour les observations souples $(X_i)_{1\leq i\leq N}$ classifiées en deux parties, $(X_i)_{1\leq i\leq N_+} \in C_+$ et $(X_i)_{1\leq i\leq N_-} \in C_-$, l'estimation des deux PDFs conditionnelles peut être donnée par :

$$\begin{aligned} \hat{f}^{b_+}_{X,N_+}(x) &= \frac{1}{N_+ h_{N_+}} \sum_{X_i \in C_+} K\Big(\frac{x - X_i}{h_{N_+}}\Big) \\ \hat{f}^{b_-}_{X,N_-}(x) &= \frac{1}{N_- h_{N_-}} \sum_{X_i \in C_-} K\Big(\frac{x - X_i}{h_{N_-}}\Big) \end{aligned} \tag{4}$$

où $K(\cdot)$ est la fonction du noyau, qui est souvent choisie comme étant la densité d'une fonction gaussienne standard avec une espérance nulle et une variance unitaire ; $h_{N_+}$ et $h_{N_-}$ sont des paramètres de lissage, qui régissent le degré de lissage de l'estimation.

En utilisant l'équation 3, on obtient l'expression ci-dessous pour le noyau gaussien :

$$\hat{p}_{e,N} = \frac{\pi_+}{N_+} \sum_{X_i \in C_+} Q\Big(\frac{X_i}{h_{N_+}}\Big) + \frac{\pi_-}{N_-} \sum_{X_i \in C_-} Q\Big(-\frac{X_i}{h_{N_-}}\Big) \tag{5}$$

où $Q(\cdot)$ vaut $\frac{1}{2}\mathrm{erfc}\Big(\frac{x}{\sqrt{2}}\Big)$.

### 1) Choix du paramètre de lissage

En général, le choix du Kernel est réputé comme peu influent sur l'estimateur, par contre le choix des paramètres de lissage est une question centrale dans l'estimation du BER. Pour une largeur $h_{N_+}$ ($h_{N_-}$) trop grande, la majorité des caractéristiques est effacée, au contraire une largeur trop faible provoque l'apparition de détails artificiels.

La figure 2 montre les résultats de l'estimation de la PDF par la méthode du noyau avec differents nombres d'observations et differentes valeurs de paramètres de lissage. On remarque que la PDF est très mal estimée si le paramètre de lissage est trop grand ou trop faible. Par conséquent, l'utilisation de la méthode du noyau requiert l'optimisation du paramètre de lissage.

Une façon répandue pour trouver la valeur optimale du $h_N$ consiste à minimiser l'erreur quadratique moyenne intégrale (IMSE : Integrated Mean Squared Error) de la PDF estimée. Pour le noyau normal et les PDFs gaussiennes $N(\mu_+, \sigma_+^2)$ et $N(\mu_-, \sigma_-^2)$, les paramètres de lissage des classes $C_+$ et $C_-$ sont :

$$h_{N_+} = \Big(\frac{4}{3N_+}\Big)^{\frac{1}{5}} \sigma_+ \qquad h_{N_-} = \Big(\frac{4}{3N_-}\Big)^{\frac{1}{5}} \sigma_- \tag{6}$$

Pour le système pratique, la forme de la distribution des observations souples est inconnue. Nous proposons d'annuler la dérivée de la fonction de vraisemblance conditionnelle.

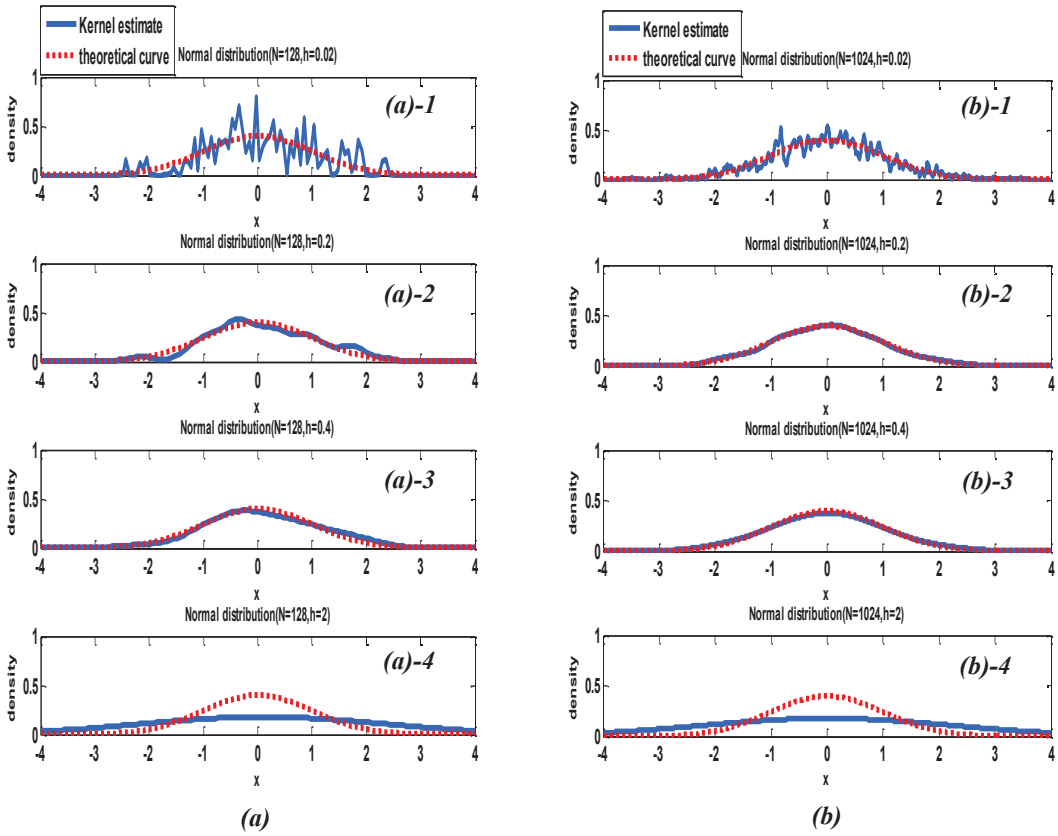


***Figure 2*** — Estimation par la méthode du noyau avec differents largeurs $h_N$ (0.02, 0.2, 0.4, 2) : (a). 128 observations ; (b). 1024 observations

$$\frac{d\left\{\left[L(h_{N_+}|X_1,\ldots,X_{N_+})\right]\right\}}{d(h_{N_+})} = -\frac{N_+}{h_{N_+}} + \sum_{i=1}^{N_+} \frac{\sum_{j=1}^{N_+} K\left(\frac{X_i-X_j}{h_{N_+}}\right)\left(\frac{(X_i-X_j)^2}{h_{N_+}^3}\right)}{\sum_{j=1,j\neq i}^{N_+} K\left(\frac{X_i-X_j}{h_{N_+}}\right)} \tag{7}$$

Cependant, le calcul de l'équation 7 est très difficile, alors nous proposons d'utiliser
- la méthode "curve fitting" : trouver la forme parfaitement adaptée à la courbe de la dérivée de la fonction de vraisemblance ;
- la méthode "Newton" : trouver le $h_N$ optimal en traçant itérativement la tangente de la dérivée de la fonction de vraisemblance.

### 2) Performances de l'estimation du BER par la méthode du noyau

Plusieurs systèmes de communications numériques ont été considérés pour tester la méthode du noyau :
- symboles BPSK transmis sur les canaux AWGN et Rayleigh ;
- système CDMA ;
- turbo codes ;
- codes LDPC.

La figure 3 montre la courbe du BER pour 1000 observations souples des bits BPSK transmis sur le canal Rayleigh. On remarque que les BERs obtenus par l'optimisation du paramètre de lissage sont les plus proches des valeurs théoriques. Le $h_N$ donné par l'équation 6 donne une mauvaise estimation du BER car la distribution est non-gaussienne. En outre, les courbes du BER estimé par la méthode du noyau sont plus lissées que celles estimées par la simulation MC.

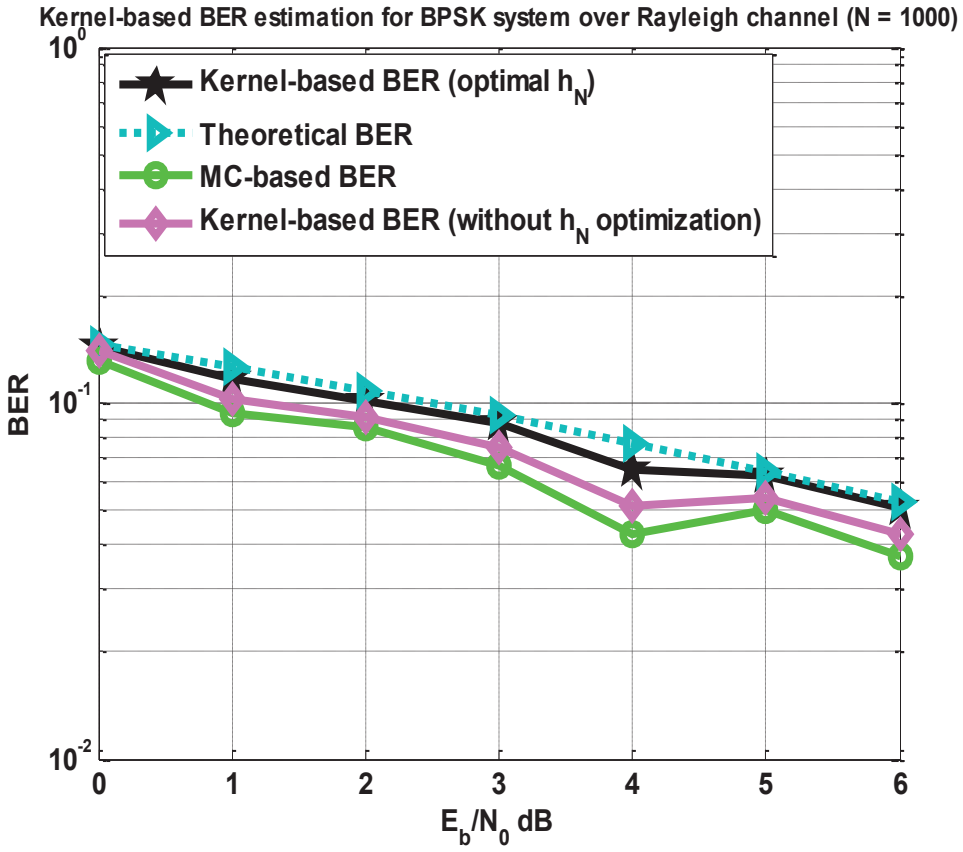


***Figure 3*** — L'estimation du BER par la méthode du noyau (symboles BPSK, canal Rayleigh, $N = 1000$)

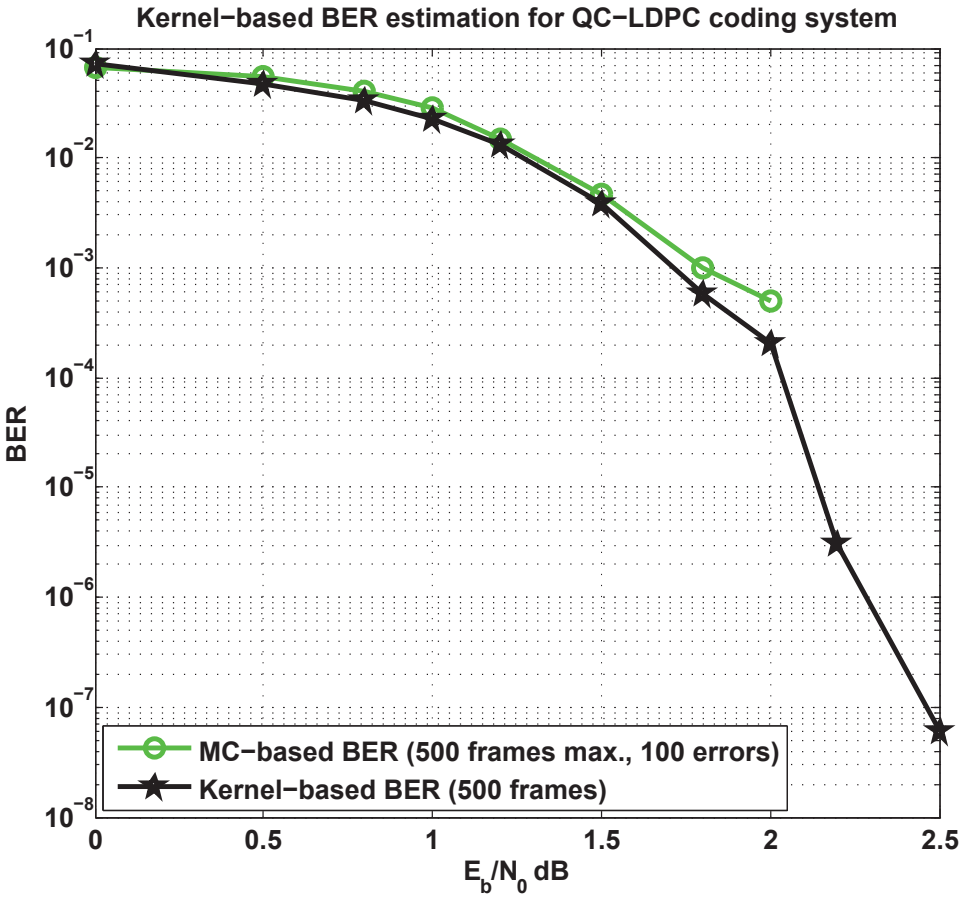


***Figure 4*** — L'estimation du BER par la méthode du noyau (LDPC codes, canal AWGN, $N = 317\ 500$)

L'intéret principal de la méthode du noyau est de réduire le nombre d'échantillons. La figure 4 montre les BERs estimés pour 317 500 observations souples des codes QC-LDPC transmis sur le canal AWGN. En comparaison avec le BER du MC qui est stoppé à $10^{-3}$, le BER estimé par Kernel peut atteindre $10^{-7}$ avec le même nombre d'échantillons.

## L'estimation du BER par le modèle de mélanges de gaussiennes

Le modèle de mélanges de gaussiennes (GMM) est une méthode très souvent utilisée pour la classification de données. Les PDFs conditionnelles des observations pour $C_+$ et $C_-$ sont définie par :

$$\begin{aligned} f_{X,N_+}(x) &= \sum_{k=1}^{K} \alpha_k^+ f_k(x; \mu_k^+, \sigma_k^{+2}) \\ f_{X,N_-}(x) &= \sum_{k=1}^{K} \alpha_k^- f_k(x; \mu_k^-, \sigma_k^{-2}) \end{aligned} \tag{8}$$

où les $(\alpha_k^+)_{1\leq k\leq K}$ (resp. $(\alpha_k^-)_{1\leq k\leq K}$) représentent les probabilités a priori de la $k^{\text{éme}}$ composante de $f_{X,N_+}$ (resp. $f_{X,N_-}$).

Le BER peut être estimé de la manière suivante :

$$\hat{p}_{e,N} = \pi_+ \sum_{k=1}^{K_+} \alpha_k^+ Q\Big(\frac{\mu_k^+}{\sigma_k^+}\Big) + \pi_- \sum_{k=1}^{K_-} \alpha_k^- Q\Big(-\frac{\mu_k^-}{\sigma_k^-}\Big) \tag{9}$$

**1) L'estimation des paramètres inconnus des composantes gaussiennes**

Les paramètres inconnus, $\theta_k^+ = (\alpha_k^+, \mu_k^+, \sigma_k^{+2})_{1\leq k\leq K}$ et $\theta_k^- = (\alpha_k^-, \mu_k^-, \sigma_k^{-2})_{1\leq k\leq K}$, sont estimés par l'algorithme EM (Expectation-Maximization) en utilisant les $N$ observations ainsi que la probabilité a posteriori où la $i^{\text{ème}}$ observations $X_i$ fait partie de la $k^{\text{ème}}$ composante gaussienne. La figure 5 montre l'organigramme de l'estimation du BER par la méthode GMM.

**2) Performances de l'estimation du BER par la méthode GMM**

La figure 6 montre les BERs estimés par la méthode GMM pour les symboles BPSK et le système CDMA. On remarque que :

- en comparaison avec la méthode MC, l'estimateur GMM permet de fournir des estimations très précises avec 2000 observations (cf. figure 6 (b)) ;
- pour la distribution non-gaussienne (eg. canal Rayleigh, figure 6 (a)), plusieurs composantes gaussiennes sont nécessaires ;
- plusieurs itérations EM sont nécessaires (cf. figure 6 (b)).

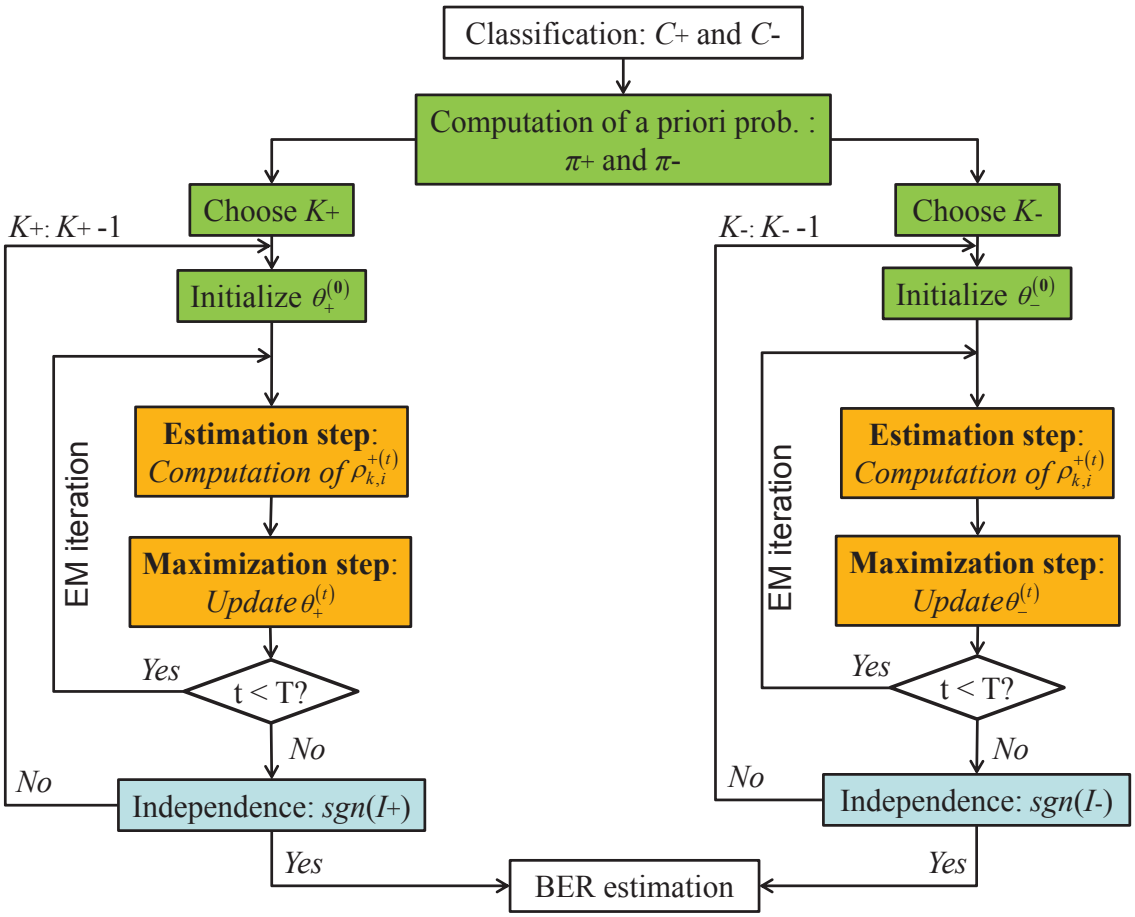


***Figure 5*** — L'organigramme de l'estimation du BER par la méthode GMM

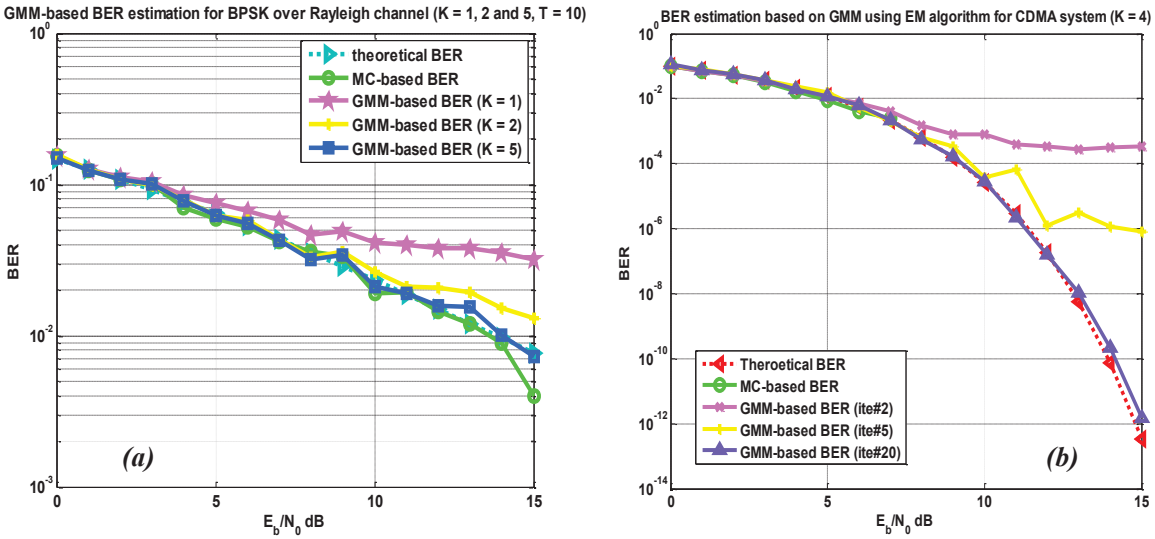


***Figure 6*** — BER estimé par la méthode GMM : (a). symboles BPSK (canal Rayleigh, $N = 2000$, $K = 1, 2, 5$, $T = 10$) (b). système CDMA (canal AWGN, $N = 2000$, $K = 4$, $T = 2, 5, 20$)

La méthode GMM est idéale pour la distribution gaussienne même avec très peu d'échantillons. Le tableau 1 montre les variances des BERs estimés du système CDMA pour differents SNR (Signal-to-Noise Ratio). En comparaison avec la simulation MC et la méthode du noyau, la méthode GMM permet de fournir la meilleure performance en terme de minimum de variance.

| | | SNR $= 0\ dB$ | SNR $= 4\ dB$ | SNR $= 6\ dB$ |
|---|---|---|---|---|
| Monte-Carlo | Variance | $5.0 \cdot 10^{-5}$ | $2.4 \cdot 10^{-5}$ | $10.3 \cdot 10^{-6}$ |
| Kernel using optimal $h_N$ | Variance | $4.3 \cdot 10^{-5}$ | $2.0 \cdot 10^{-5}$ | $1.0 \cdot 10^{-6}$ |
| GMM | Variance | $3.7 \cdot 10^{-5}$ | $1.6 \cdot 10^{-5}$ | $0.85 \cdot 10^{-6}$ |

**Table *1*** — Variance du BER estimé pour la méthode MC, Kernel et GMM (système CDMA, canal AWGN, $N = 2000$, 1000 épreuves)

## L'estimation en aveugle du BER

Pour une estimation au niveau du récepteur, les bits transmis sont inconnus. Autrement dit, il faut estimer le BER de façon aveugle sans connaître la classification des données $C_+$ et $C_-$, ainsi que les nombres d'échantillons associés $N_+$ et $N_-$. Pour cela, nous proposons d'utiliser l'algorithme Stochastic Expectation-Maximization (SEM). La classification est initialement réalisée par le signe des observations souples. La figure 8 montre l'organigramme pour l'estimation du BER en aveugle avec la méthode du noyau et la méthode GMM.

La figure 7 montre les BERs estimés en aveugle en utilisant la méthode du noyau pour les symboles BPSK sur le canal AWGN. Les bits transmis sont générés de façon non-equiprobable ($\pi_+ = 0.7$ et $\pi_- = 0.3$).

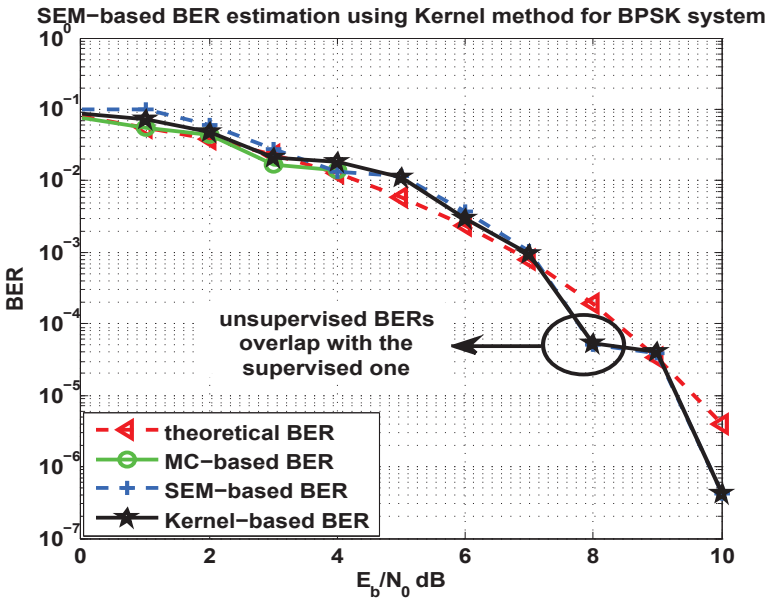


***Figure 7*** — BER estimé en aveugle en utilisant la méthode du noyau pour les symboles BPSK (canal AWGN, $N = 1000$, $\pi_+ = 0.7$ et $\pi_- = 0.3$)

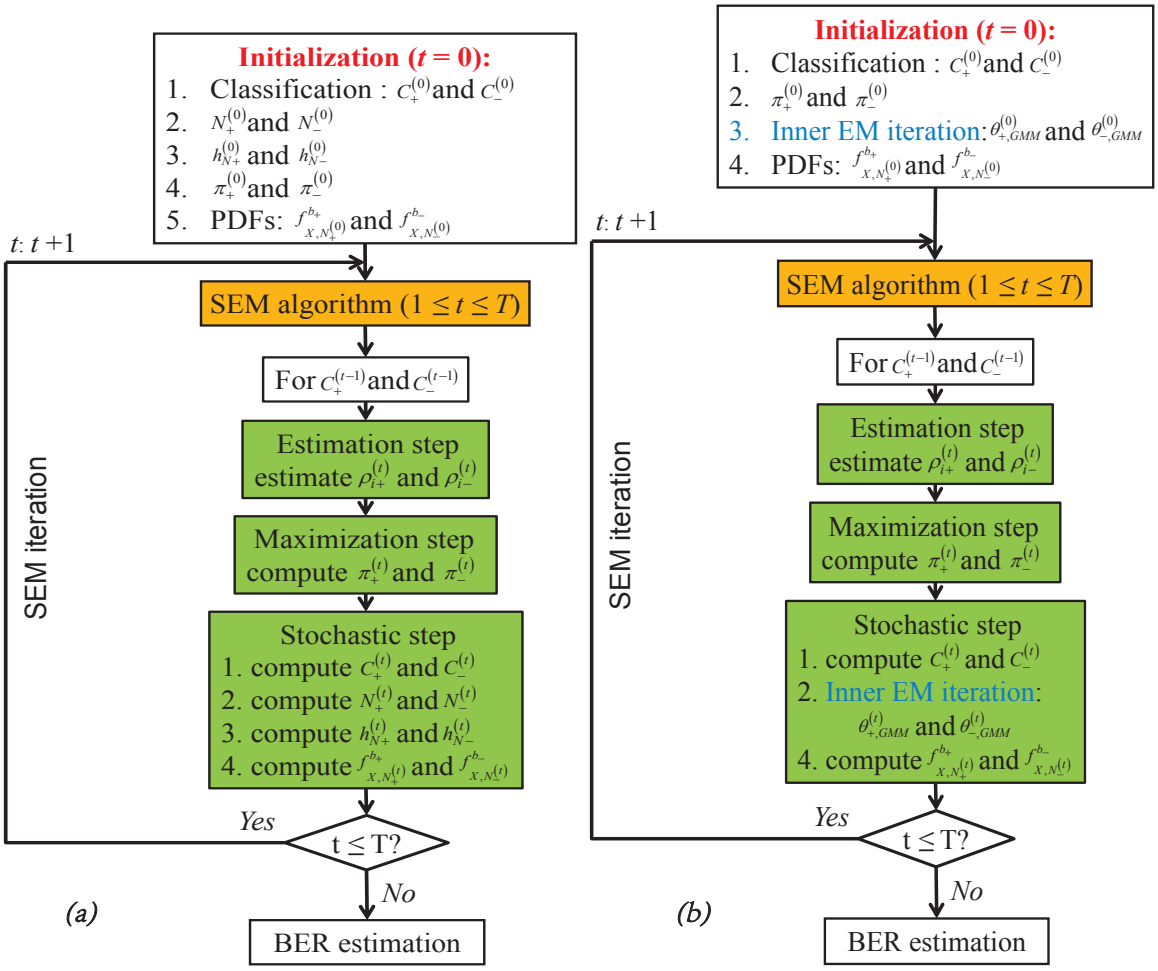


***Figure 8*** — L'organigramme pour l'estimation du BER en aveugle : (a). Kernel ; (b). GMM

On remarque que le BER estimé de façon aveugle devient de plus en plus proche de celui estimé par la méthode Kernel avec l'augmentation du SNR, surtout pour des SNR supérieur à 7 dB. Cela est dû au choix initial des classes $C_+$ et $C_-$. Quand le SNR est faible, la classification initiale provoque des erreurs. Au contraire, pour des valeurs de SNR élevé, très peu d'erreurs pourront être trouvés. Dans ce cas-là, l'estimation en aveugle est équivalente à l'estimation avec la connaissance des bits envoyés.

Plusieurs avantages de l'estimation du BER en aveugle peuvent être trouvés :

- elle est très adaptée à l'application pratique (eg. l'estimation du BER en ligne et en temps réel) compte tenu des exigences citées au début du résumé ;
- elle bénéficie des intérêts de la méthode du noyau ou de la GMM :
  1. réduire le nombre d'echantillons ;
  2. minimiser la variance de l'estimation ;
  3. servir pour n'importe quel type d'observations souples ;
  4. le temps de simulation ne dépend pas du SNR.

# Publications

**2011**

SAOUDI Samir, DONG Jia, ET TOLBA Mohamed. Estimation rapide de la probabilité d'erreur pour les systèmes de communicationos numériques. *GRETSI*, 05-08 septembre 2011, Bordeaux, France, 2011.

SAOUDI Samir, DONG Jia. La méthode du noyau pour estimer rapidement une très faible probabilité d'erreur. *TAIMA 2011 : Traitement et analyse de l'information, méthodes et applications*, 03-08 octobre 2011, Hammamet, Tunisie, 2011.

**2012**

DONG Jia, AIT IDIR Tarik, SAOUDI Samir. Unsupervised Bit Error Rate Estimation Using Maximum Likelihood Kernel Methods. *VTC 2012-Spring : IEEE 75th Vehicular Technology Conference, IEEE*, 06-09 may 2012, Yokohama, Japan, 2012, ISBN 978-1-4673-0989-9.

SAOUDI Samir, DONG Jia. Kernel method for Bit Error Rate Estimation. *ISIVC12 : International Symposium on signal, Image, Video and Communications*, 04-06 july 2012, Valenciennes, France, 2012.

SAOUDI Samir, DONG Jia. Joint Maximum Likelihood and Expectation Maximization methods for Unsupervised Iterative Soft Bit Error Rate Estimation. *VTC 2012-Fall : 76th IEEE Vehicular Technology Conference*, 03-06 september 2012, Québec, Canada, 2012.

# Bibliography


[AG99] M-S Alouini and Andrea J Goldsmith. Capacity of rayleigh fading channels under different adaptive transmission and diversity-combining techniques. *Vehicular Technology, IEEE Transactions on*, 48(4) :1165–1181, 1999.

[ÄKM07] Sami Äyrämö, Tommi Kärkkäinen, and Kirsi Majava. Robust refinement of initial prototypes for partitioning-based clustering algorithms. *Recent Advances in Stochastic Modeling and Data Analysis*, pages 473–482, 2007.

[And85] Larry C Andrews. *Special functions for engineers and applied mathematicians*. Macmillan New York, 1985.

[And99] Eric C Anderson. Monte carlo methods and importance sampling. *Lecture Notes for Statistical Genetics*, 578, 1999.

[AV07] David Arthur and Sergei Vassilvitskii. k-means++ : The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.

[BBW00] MJ Baxter, CC Beardah, and S Westwood. Sample size and related issues in the analysis of lead isotope data. *Journal of Archaeological Science*, 27(10) :973–980, 2000.

[BEF84] James C Bezdek, Robert Ehrlich, and William Full. Fcm : The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2) :191–203, 1984.

[BG96] Claude Berrou and Alain Glavieux. Near optimum error correcting coding and decoding : Turbo-codes. *Communications, IEEE Transactions on*, 44(10) :1261–1271, 1996.

[BGT93] Claude Berrou, Alain Glavieux, and Punya Thitimajshima. Near shannon limit error-correcting coding and decoding : Turbo-codes. 1. In *Communications, 1993. ICC 93. Geneva. Technical Program, Conference Record, IEEE International Conference on*, volume 2, pages 1064–1070. IEEE, 1993.

[CD+84] Gilles Celeux, Jean Diebolt, et al. Reconnaissance de mélange de densité et classification. un algorithme d'apprentissage probabiliste : l'algorithme sem. 1984.

[CD85] Gilles Celeux and Jean Diebolt. The sem algorithm : a probabilistic teacher algorithm derived from the em algorithm for the mixture problem. *Computational statistics quarterly*, 2(1) :73–82, 1985.

[CD+86] Gilles Celeux, Jean Diebolt, et al. Etude du comportement asymptotique d'un algorithme d'apprentissage probabiliste pour les melanges de lois de probabilite. 1986.

[CD87] G Celeux and J Diebolt. A probabilistic teacher algorithm for iterative maximum likelihood estimation. In *1. Conference of the International Federation of Classification Societies*, pages 617–624, 1987.

[CHD09] Enver Cavus, Charles L Haymes, and Babak Daneshrad. Low ber performance estimation of ldpc codes via application of importance sampling to trapping sets. *Communications, IEEE Transactions on*, 57(7) :1886–1888, 2009.

[Cra01] Kyle Cranmer. Kernel estimation in high-energy physics. *Computer Physics Communications*, 136(3) :198–207, 2001.

[CS09] S Chandrakala and C Chandra Sekhar. Model based clustering of audio clips using gaussian mixture models. In *Advances in Pattern Recognition, 2009. ICAPR'09. Seventh International Conference on*, pages 47–50. IEEE, 2009.

[Das91] Belur V Dasarathy. Nearest neighbor (NN) norms :NN pattern classification techniques. 1991.

[DFL95] John DiNardo, Nicole M Fortin, and Thomas Lemieux. Labor market institutions and the distribution of wages, 1973-1992 : A semiparametric approach. Technical report, National Bureau of Economic Research, 1995.

[DHS12] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 2012.

[DLR77] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.

[DWY09] Liang Dong, Wang Wenbo, and Li Yonghua. Monte carlo simulation with error classification for multipath rayleigh fading channel. In *Telecommunications, 2009. ICT'09. International Conference on*, pages 223–227. IEEE, 2009.

[DY79] Persi Diaconis and Donald Ylvisaker. Conjugate priors for exponential families. *The Annals of statistics*, 7(2) :269–281, 1979.

[Far90] Julian Faraway. Implementing semiparametric density estimation. *Statistics & Probability Letters*, 10(2) :141–143, 1990.

[For65] Edward W Forgy. Cluster analysis of multivariate data : efficiency versus interpretability of classifications. *Biometrics*, 21 :768–769, 1965.

[Gal62] Robert Gallager. Low-density parity-check codes. *Information Theory, IRE Transactions on*, 8(1) :21–28, 1962.

[Gna11] Ram Gnanadesikan. *Methods for statistical data analysis of multivariate observations*, volume 321. John Wiley & Sons, 2011.

[Gre90] Peter J Green. On use of the em for penalized likelihood estimation. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 443–452, 1990.

[Gre03] William H Greene. *Econometric Analysis, 5/e*. Pearson Education India, 2003.

[Han09] Bruce E Hansen. Lecture notes on nonparametrics. *Lecture notes*, 2009.

[Har75] John A Hartigan. *Clustering algorithms*. John Wiley & Sons, Inc., 1975.

[HC06] Dongwoo Hong and Kwang-Ting Cheng. Bit error rate estimation for improving jitter testing of high-speed serial links. In *Test Conference, 2006. ITC'06. IEEE International*, pages 1–10. IEEE, 2006.

[HW01] M Mostofa K Howlader and Brian D Woerner. Decoder-assisted frame synchronization for packet transmission. *Selected Areas in Communications, IEEE Journal on*, 19(12) :2331–2345, 2001.

[JAN13] JANG. Gaussian mixture model, 2013. [Online ; accessed 24-September-2013].

[JBS00] Michel C Jeruchim, Philip Balaban, and K Sam Shanmugan. *Simulation of communication systems : modeling, methodology and techniques*. Springer, 2000.

[Jer84] Michel Jeruchim. Techniques for estimating the bit error rate in the simulation of digital communication systems. *Selected Areas in Communications, IEEE Journal on*, 2(1) :153–170, 1984.

[JSM97] Jyh-Shing Roger Jang, Chuen-Tsai Sun, and Eiji Mizutani. Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [book review]. *Automatic Control, IEEE Transactions on*, 42(10) :1482–1484, 1997.

[Kel03] Carl T Kelley. *Solving nonlinear equations with Newton's method*, volume 1. Siam, 2003.

[LFKL00] Rainer Lucas, Marc PC Fossorier, Yu Kou, and Shu Lin. Iterative decoding of one-step majority logic deductible codes based on belief propagation. *Communications, IEEE Transactions on*, 48(6) :931–937, 2000.

[LH01] Ingmar Land and Peter A Hoeher. Using the mean reliability as a design and stopping criterion for turbo codes. In *Information Theory Workshop, 2001. Proceedings. 2001 IEEE*, pages 27–29. IEEE, 2001.

[LHS00] Ingmar Land, Peter Hoeher, and Ulrich Sorger. Log-likelihood values and monte carlo simulation-some fundamental results. In *Proc. Int. Symp. on Turbo Codes & Rel. Topics, Sept. 2000*, pages 43–46, 2000.

[LLF+04] Yi Lu, Shiyong Lu, Farshad Fotouhi, Youping Deng, and Susan J Brown. FGKA : A fast genetic k-means clustering algorithm. In *Proceedings of the 2004 ACM symposium on Applied computing*, pages 622–623. ACM, 2004.

[LLL06] Younjeong Lee, Ki Yong Lee, and Joohun Lee. The estimating optimal number of gaussian mixtures based on incremental k-means for speaker identification. *International Journal of Information Technology*, 12(7) :13–21, 2006.

[LRT03] Jeffery D Laster, Jeffrey H Reed, and WH Tranter. Bit error rate estimation using probability density function estimators. *Vehicular Technology, IEEE Transactions on*, 52(1) :260–267, 2003.

[M+67] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.

[Man06] Gaurav Pradeep Mandhare. *Distributed power control algorithms for WCDMA cellular systems*. PhD thesis, Concordia University, 2006.

[McL80] GJ McLachlan. Estimation of mixing proportions by the em algorithm. In *Proceedings of the Statistical Computing Section of the Annual Meeting of the American Statistical Association*, pages 140–143, 1980.

[MN96] David JC MacKay and Radford M Neal. Near shannon limit performance of low density parity check codes. *Electronics letters*, 32(18) :1645, 1996.

[Moo96] Todd K Moon. The expectation-maximization algorithm. *Signal processing magazine, IEEE*, 13(6) :47–60, 1996.

[MP04] Geoffrey McLachlan and David Peel. *Finite mixture models*. Wiley. com, 2004.

[MYK05] Seho Myung, Kyeongcheol Yang, and Jaeyoel Kim. Quasi-cyclic ldpc codes for fast encoding. *Information Theory, IEEE Transactions on*, 51(8) :2894–2901, 2005.

[MZB01] Hlaing Minn, Mao Zeng, and Vijay K Bhargava. On arq scheme with adaptive error control. *Vehicular Technology, IEEE Transactions on*, 50(6) :1426–1436, 2001.

[Par62] Emanuel Parzen. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3) :1065–1076, 1962.

[QGP99] Yonghong QIU, Zhongmin GAN, and Yahan PAN. Research on application of software simulation to spread spectrum communication systems. *Journal of System Simulation*, 11(6) :461–464, 1999.

[Que56] Maurice H Quenouille. Notes on bias in estimation. *Biometrika*, 43(3/4) :353–360, 1956.

[Ros56] Murray Rosenblatt. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, pages 832–837, 1956.

[RW84] Richard A Redner and Homer F Walker. Mixture densities, maximum likelihood and the em algorithm. *SIAM review*, 26(2) :195–239, 1984.

[SAIM11] Samir Saoudi, Tarik Ait-Idir, and Yukou Mochida. A novel non-parametric iterative soft bit error rate estimation technique for digital communications systems. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.

[SBG+66] Yulii Anatolevich Shreider, Nikolai Panteleimonovich Buslenko, DI Golenko, IM Sobol, VG Sragovich, GJ Tee, and DM Parkyn. *The Monte Carlo method : the method of statistical trials*. Pergamon Press New York, 1966.

[SGH97] S Saoudi, F Ghorbel, and A Hillion. Some statistical properties of the kernel-diffeomorphism estimator. *Applied stochastic models and data analysis*, 13(1) :39–58, 1997.

[Sha01] Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1) :3–55, 2001.

[SHG94] S Saoudi, A Hillion, and F Ghorbel. Non–parametric probability density function estimation on a bounded support : Applications to shape classification and speech coding. *Applied Stochastic models and data analysis*, 10(3) :215–231, 1994.

[Sil86] Bernard W Silverman. *Density estimation for statistics and data analysis*, volume 26. CRC press, 1986.

[SLS03] Emina Soljanin, R Liu, and P Spasojevic. Hybrid arq in wireless networks. In *DIMACS Workshop on Network Information Theory*, 2003.

[SPKK99] SK Shin, SK Park, JM Kim, and SC Ko. New quasi-analytic ber estimation technique on the nonlinear satellite communication channels. *IEE Proceedings-Communications*, 146(1) :68–72, 1999.

[SSB05] E Calvanese Strinati, Sébastien Simoens, and Joseph Boutros. Ber and per estimation based on soft output decoding. In *9th International OFDM-Workshop*, 2005.

[STG09] Samir Saoudi, Molka Troudi, and Faouzi Ghorbel. An iterative soft bit error rate estimation of any digital communication systems using a non-parametric probability density function. *EURASIP Journal on Wireless Communications and Networking*, 2009 :4, 2009.

[STTP10] Saoudi Samir, Derham Thomas, Ait-Idir Tarik, and Coupe Patrice. A fast soft bit error rate estimation method. *EURASIP Journal on Wireless Communications and Networking*, 2010, 2010.

[TA03] Emili Tortosa-Ausina. Bank cost efficiency as distribution dynamics : controlling for specialization is important. *Investigaciones económicas*, 27(1) :71–96, 2003.

[Tan81] R Tanner. A recursive approach to low complexity codes. *Information Theory, IEEE Transactions on*, 27(5) :533–547, 1981.

[TAS07] Molka Troudi, Adel M Alimi, and Samir Saoudi. Fast plug-in method for parzen probability density estimator applied to genetic neutrality study. In *EUROCON, 2007. The International Conference on" Computer as a Tool"*, pages 1034–1039. IEEE, 2007.

[TJBB06] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476), 2006.

[TS92] George R Terrell and David W Scott. Variable kernel density estimation. *The Annals of Statistics*, pages 1236–1265, 1992.

[TSM+85] D Michael Titterington, Adrian FM Smith, Udi E Makov, et al. *Statistical analysis of finite mixture distributions*, volume 7. Wiley New York, 1985.

[VRV94] William N Venables, Brian D Ripley, and WN Venables. *Modern applied statistics with S-PLUS*, volume 250. Springer-verlag New York, 1994.

[VWR+01] Gerd Vandersteen, Piet Wambacq, Yves Rolain, Johan Schoukens, Stéphane Donnay, Marc Engels, and Ivo Bolsens. Efficient bit-error-rate estimation of multicarrier transceivers. In *Design, Automation and Test in Europe, 2001. Conference and Exhibition 2001. Proceedings*, pages 164–168. IEEE, 2001.

[Wik13a] Wikipedia. Importance sampling — wikipedia, the free encyclopedia, 2013. [Online ; accessed 24-September-2013].

[Wik13b] Wikipedia. Ordinary least squares — wikipedia, the free encyclopedia, 2013. [Online ; accessed 24-September-2013].

[WJ95] M Matt P Wand and MM Chris Jones. *Kernel smoothing*, volume 60. Crc Press, 1995.

[WL05] T Warren Liao. Clustering of time series data-a survey. *Pattern Recognition*, 38(11) :1857–1874, 2005.

[Wol80] Joseph J Wolcin. Maximum a posteriori estimation of narrow-band signal parameters. *The Journal of the Acoustical Society of America*, 68 :174, 1980.

[WW78] MS Waterman and DE Whiteman. Estimation of probability densities by empirical density functions. *International Journal of Mathematical Education in Science and Technology*, 9(2) :127–137, 1978.

[YZ01] Zheng Rong Yang and Mark Zwolinski. Mutual information theory for adaptive mixture models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(4) :396–403, 2001.