DIPLOMARBEIT

QUANTIZATION FOR MULTITERMINAL INFERENCE

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Diplomingenieurs

unter der Leitung von

Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Gerald Matz
Univ.Ass. Dipl.-Ing. Andreas Winkelbauer
Institute of Telecommunications

eingereicht an der Technischen Universität Wien
Fakultät für Elektrotechnik und Informationstechnik

von

Thomas Kropfreiter
Marinelligasse 18/19
1020 Wien

Wien, im November 2014 _____

—— To my parents ——

Abstract

This thesis deals with the topic of quantization for multiterminal detection or distributed hypothesis tests. We extend the concept of Neyman-Pearson detection and Bayesian detection to a multiterminal setting. Specifically, the scenario consists of a number of sensors which collect and process local measurements. Each sensor calculates a local statistic of the measurements, e.g., a log-likelihood ratio (LLR). The LLR value is quantized and afterwards reported to a central processing unit, the fusion center (FC). The FC gathers all LLRs from the sensors and makes a final decision.

Special emphasizes is put on the issue of quantization. The local LLR values have to be quantized before they can be transmitted to the FC. Because it is not evident how to design the quantizer to maximize the overall system performance, we consider three different quantizer design strategies: The Lloyd-Max quantizer, the maximum output entropy (MOE) quantizer, and a quantizer based on the information bottleneck method (IBM).

In the last part of this thesis we corroborate our theoretical findings by numerical simulation results. More precisely, we evaluate the performance of distributed hypothesis tests for a variety of different settings. Furthermore, we discuss several possible extensions which could yield improved performance.

Kurzfassung

Diese Diplomarbeit beschäftigt sich mit Quantisierung für Hypothesentests in verteilten Detektionsszenarien. Dafür erweitern wir die Grundkonzepte für klassische Neyman-Pearson-Detektoren und Bayes'sche-Detektoren auf verteilte Szenarien. Genauer gesagt besteht das Szenario aus mehreren Sensoren, die lokale Messungen vornehmen und verarbeiten. Aus diesen Messungen werden anschließend *log-likelihood ratios* (LLRs) berechnet. Diese werden quantisiert und an die globale Verarbeitungseinheit, das *fusion center* (FC), übermittelt. Das FC sammelt alle LLR-Werte und fällt eine Entscheidung.

Besondere Aufmerksamkeit wird dem Thema der Quantisierung gewidmet, da die ermittelten LLR Werte quantisiert werden müssen, bevor sie dem FC übermittelt werden können. Da es im Vorhinein nicht klar ist, wie die Quantisierer entworfen werden müssen, damit die Gesamtleistung des Systems maximiert wird, stellen wir drei verschiedene Quantisiererentwurfstrategien vor: Den Lloyd-Max-Quantisierer, den *maximum output entropy* (MOE) Quantisierer und einen Quantisierer basierend auf der *information bottleneck-Methode* (IBM).

Im letzten Teil der Diplomarbeit untermauern wir unsere theoretischen Aussagen mit Hilfe von numerischen Simulationsergebnissen. Dabei ermitteln wir die Leistungsfähigkeit eines verteilten Hypothesentests für eine Vielzahl an verschiedenen Szenarien. Abschließend diskutieren wir Möglichkeiten und Erweiterungen zur Verbesserung des verteilten Detektionssystems.

Acknowledgements

I would like to express my gratitude to a number of people who supported me during my studies and especially who helped me to finish my work on this thesis.

First of all I want to thank Gerald Matz for offering me the opportunity to compose this thesis. Thank you for your interest in my work and your patience in finishing this thesis.

Very special thanks I want to express to my supervisor Andreas Winkelbauer. He always answered my questions and requests immediately and his profound advices helped me a lot. Additionally, he spent a lot of time in proofreading this thesis which improved the quality of this work tremendously. I am very grateful for your support.

At last, I want to express my honest gratitude to my family and all my friends who supported me during my studies. Especially, I want to say thank you to my parents, Renate and Josef, who gave me the opportunity to study at the university. Thank you for let me choose my own way of live.

Contents

1

Introduction

In this thesis we deal with the topic of *distributed sensor networks* (DSNs). A DSN consists of a number of sensors, which obtain measurements from the environment, abstract relevant information out of the measurement, and report that information to a central processing unit, known as the fusion center (FC).

We differentiate between *centralized* networks and *decentralized* networks. In a centralized network each sensor just transmits its raw measurement sample to the FC without further processing at the sensor. In contrast, in a decentralized network, data processing is applied at the sensors. Furthermore, the sensors have the ability to communicate with each other to exchange their measurement samples or some other relevant information. Afterwards a preprocessed, often condensed, version of the measurement samples is reported to the FC. In the field of DSN this is known as *intelligence at the node*. Most DSNs are implemented as decentralized network because the benefits are reduced communication bandwidth, increased reliability, and faster response to rapidly changing channel conditions. Due to the fact that the FC may not have all information of each sensor, the decentralized network achieves a lower performance than the centralized network in general [1]. However, in this thesis we deal with a decentralized network with FC and intelligence at each node and we assume that each sensor is able to report its information to the FC.

Often DSNs are used to perform statistical inference given some measurement data. We differentiate between *distributed hypothesis testing* or *distributed detection*, and *distributed estimation*. We will deal with the topic of distributed detection.

Indeed, in the last years distributed detection with multiple sensors has become an important issue. Due to the increased computational capabilities, the availability of high speed communication and the decreased costs of sensors there was a huge effort in this research field. Actually, this topic has its origin in military applications like surveillance of enemy targets, but nowadays there are also a variety of civilian applications available [1].

One application of distributed detection is a *cognitive radio system*. A cognitive radio system consists of a variety of users, the sensors, which sense the frequency bands if there is traffic or not. Afterwards the sensors report some related data to the FC. There the information of all sensors is collected and the FC makes an overall decision, whether the frequency band is free or not. Then the FC is able to allocate free frequency bands to users that want to transmit data over the channel. Furthermore, this strategy enables to adapt parameters like carrier frequency and power dynamically. So one goal might be to maximize the overall data throughput by dynamically adapting the transmission parameters to the temporally changing channel conditions [2].

1.1 Outline

In this thesis, we deal with a distributed binary hypothesis testing scenario with intelligence at the sensors. Each sensor extracts relevant information from the measurements and reports it to the FC. The FC decides which hypothesis is in force. We evaluate the performance of the overall system and show opportunities to improve the performance of the test. Special consideration is put on the quantization of the information which is reported from each sensor to the FC. The structure of this thesis is as follows:

- In Chapter 2, we introduce the basics of distributed hypothesis testing systems according to two concepts: The Neyman-Pearson (NP) detector and the Bayesian detector. We discuss LLR tests and their connection to hypothesis tests and show

that LLR tests are optimum if the detection scenario satisfies some constraints.

- Chapter 3 is devoted to the topic of quantizer design. The information extracted from the measurements needs to be quantized before it can be reported to the FC. Therefore, we introduce three different approaches of quantizer design strategies and discuss their main properties.

- In Chapter 4 we evaluate the performance of our distributed detection system. We discuss several ways to increase the performance of our system and specify hypothesis tests for different testing environments.

- In the last Chapter 5 we summarize our results and give some motivation for future research topics.

2

Distributed Detection with Multiple Sensors

2.1 Likelihood Ratio

A hypothesis test can be depicted as a parameter space Θ consisting of a finite number of subsets θ_i, the so called hypotheses, which are mapped via a probability density function (pdf) $p_\theta(x)$ to an observation space $\mathcal{X}$. The task of the hypothesis test is to make a decision which hypothesis θ_i is in force using the observation x. We focus on to the case of *simple binary hypothesis tests*. *Binary* refers to the fact that there are only two hypothesis $\mathcal{H}_0 : \theta \in \Theta_0$ and $\mathcal{H}_1 : \theta \in \Theta_1$. $\mathcal{H}_0$ is called the *null hypothesis* and $\mathcal{H}_1$ the *alternative hypothesis*. Furthermore, each of the two hypotheses consists of a single value. Therefore the hypotheses can be written as $\mathcal{H}_0 : \theta \in \theta_0$ and $\mathcal{H}_1 : \theta \in \theta_1$. This is known as a *simple* test. $\mathcal{H}_0$ will be represented by the value -1 and $\mathcal{H}_1$ by the value $+1$. The mapping from the parameter space Θ to the observation space $\mathcal{X}$ is random and is modeled by the introduction of an additive noise term n. Using a sample $x \in \mathcal{X}$ we try to detect, which hypothesis is in force. The problem can now be summarized as follows:

$$\begin{aligned} \mathcal{H}_0 : \quad x &= -1 + n, \\ \mathcal{H}_1 : \quad x &= \quad 1 + n. \end{aligned} \tag{2.1}$$

In most parts of this thesis the noise is modeled as Gaussian with zero mean and known variance, i.e., $n \sim \mathcal{N}(0, \sigma^2)$. In later chapters we drop the Gaussian noise assumption and replace it by Laplace and Cauchy distributed noise. Returning to the Gaussian case this problem is sometimes called the *mean-shifted Gauss-Gauss* problem [3]. The hypothesis test (2.1) will be treated throughout the whole thesis, further important properties will be derived from (2.1) and will also be the basis of the simulation part. Note that x and n in (2.1) are considered as scalars. The case where more than one measurement sample is taken is treated in the simulation part in Chapter 4.

Although we have introduced a random mapping from the parameter space Θ to the observation space $\mathcal{X}$, we first consider the case where no observation is available at all. Under such circumstances we can only make a meaningful decision if we have knowledge about the prior probabilities of the hypothesis $P\{\mathcal{H}_0\}$ and $P\{\mathcal{H}_1\}$. In literature the ratio of the prior probabilities $\frac{P\{\mathcal{H}_1\}}{P\{\mathcal{H}_0\}}$ is known as the the *prior probability ratio*. A quite intuitive decision can be made if we decide for $\mathcal{H}_1$, if the ratio is greater than one and for $\mathcal{H}_0$ if the ratio is less than one. This result can be expressed by a so called *decision function* $\phi(\cdot)$

$$\phi(x) = \begin{cases} -1, & \frac{P\{\mathcal{H}_1\}}{P\{\mathcal{H}_0\}} < 1 \\ 1, & \frac{P\{\mathcal{H}_1\}}{P\{\mathcal{H}_0\}} \geq 1 \end{cases}. \tag{2.2}$$

The decision function $\phi(x)$ outputs -1 if the prior probability ratio is less than 1, and it outputs 1 otherwise. If the priors $P\{\mathcal{H}_0\}$ and $P\{\mathcal{H}_1\}$ are unknown or the hypothesis are distributed equally likely, the best thing we can do is a pure guess.

Next we consider the case where an observation x is available. We try to incorporate x to find a more reliable decision rule. Therefore, we define the *posterior probability ratio* as

$$R(x) = \frac{P\{\mathcal{H}_1|x\}}{P\{\mathcal{H}_0|x\}}. \tag{2.3}$$

The conditional probability $P\{\mathcal{H}_i|x\}$ is the probability that $\mathcal{H}_i$ is in force after x was observed. We again favor $\mathcal{H}_1$ if $R(x)$ is greater than one and $\mathcal{H}_0$ if $R(x)$ is less than one. Applying Bayes' theorem we can define another quantity which is known as the

famous *likelihood ratio*:

$$L(x) = \frac{p(x|\mathcal{H}_1)}{p(x|\mathcal{H}_0)}. \tag{2.4}$$

The conditional pdf $p(x|\mathcal{H}_i)$ is the pdf of the observation x given that hypothesis $\mathcal{H}_i$ is in force. The likelihood ratio $L(x)$ contains all information that carries the data x about the two hypotheses. The relationship between the posterior probability ratio and the likelihood ratio is

$$R(x) = L(x)\frac{P\{\mathcal{H}_1\}}{P\{\mathcal{H}_0\}}. \tag{2.5}$$

Depending on the specific conditional pdfs it may often be more convenient to take the logarithm of the $L(x)$. This quantity is known as the *log-likelihood ratio* (LLR):

$$l(x) = \log L(x). \tag{2.6}$$

Recalling from (2.3) that we favor $\mathcal{H}_1$, if the posterior probability ratio R is greater than 1 and $\mathcal{H}_0$ if R is less than 1, we can equivalently express this test by the decision function

$$\phi(x) = \begin{cases} -1, & l(x) < \log\frac{P\{\mathcal{H}_0\}}{P\{\mathcal{H}_1\}} \\ 1, & l(x) \geq \log\frac{P\{\mathcal{H}_0\}}{P\{\mathcal{H}_1\}} \end{cases}. \tag{2.7}$$

The above decision function is known as log-likelihood ratio test (LRT). The hypothesis test simplifies to a simple threshold test where the threshold is given by the logarithm of the ratio of the prior probabilities. The procedure of finding a decision is as follow: We first calculate the prior probability ratio and the log-likelihood ratio $l(x)$, where we have assumed that we know the underlying probability distributions. Next, we take an observation x and evaluate $l(x)$. If the result is smaller than the threshold we decide for $\mathcal{H}_0$, if it is greater we decide for $\mathcal{H}_1$. The case where $l(x)$ is equal to the threshold is only possible for discrete distributed random variables and will further specified in Chapter 4 when we deal with quantized variables.

Even though this LRT is a nice result we haven't proven yet if the result is optimum

in any sense. That it is optimum and under which circumstances will be shown in the next two sections.

2.2 Neyman Pearson Detector

2.2.1 Fundamentals

We now continue our investigation on the hypothesis test in (2.1). As stated earlier we deal with a binary problem. The parameter space Θ consists of the two elements -1 and 1. After observing a data sample x we make a decision according the LRT (2.7). The result is again -1 or 1. We notice that there are four possible decisions depending which hypotheses is in force and which we favor. This can be summarized as follows:

$$\begin{aligned}
&\theta \in \Theta_0, \quad x \in \mathcal{X}_0 \quad \text{correctly accepted,} \\
&\theta \in \Theta_0, \quad x \in \mathcal{X}_1 \quad \text{type I error or false alarm,} \\
&\theta \in \Theta_1, \quad x \in \mathcal{X}_0 \quad \text{type II error or miss,} \\
&\theta \in \Theta_1, \quad x \in \mathcal{X}_1 \quad \text{correct detection.}
\end{aligned} \tag{2.8}$$

In the Neyman Pearson (NP) detector case θ is modeled as a deterministic variable and x as random variable. Therefore the prior probabilities $P\{\mathcal{H}_0\}$ and $P\{\mathcal{H}_1\}$ do not exist and can not be incorporated into the test to achieve a better performance. The mapping from the parameter space Θ to the observation space $\mathcal{X}$ is random and can be expressed as $P\{x \in \mathcal{X}_i|\mathcal{H}_i\}$, $i = \{-1, 1\}$. Therefore we can assign a probability mass to each of the four cases of (2.8):

$$\begin{aligned}
\text{correctly accepted:} \quad & P_{\mathrm{A}} = P\{x \in \mathcal{X}_0|\mathcal{H}_0\}, \\
\text{false alarm:} \quad & P_{\mathrm{FA}} = P\{x \in \mathcal{X}_1|\mathcal{H}_0\}, \\
\text{miss:} \quad & P_{\mathrm{M}} = P\{x \in \mathcal{X}_0|\mathcal{H}_1\}, \\
\text{correct detection:} \quad & P_{\mathrm{D}} = P\{x \in \mathcal{X}_1|\mathcal{H}_1\}.
\end{aligned} \tag{2.9}$$

Usually we aim to minimize the probabilities of an error and maximize the probability of a correct detection. But it shall be noted that both error types can not be minimized simultaneously. Typically in communication scenarios both error types are treated equally. That means we do not prefer any of the two errors. In contrast for instance in radar systems a miss is much worse than a false alarm error. Therefore the goal is to minimize the miss probability P_{M} while keeping the false alarm probability P_{FA} fixed. This latter approach leads to the famous NP theorem:

Theorem 1.1 To maximize P_{D} for a given $P_{\mathrm{FA}} = \alpha$ decide $\mathcal{H}_1$ if

$$L(x) = \frac{p(x|\mathcal{H}_1)}{p(x|\mathcal{H}_0)} > \gamma, \tag{2.10}$$

where the threshold γ is found from

$$P_{\mathrm{FA}} = \int_{\{x:L(x)>\gamma\}} p(x|\mathcal{H}_0)dx = \alpha. \tag{2.11}$$

Here α is known as the *size* of the test. The proof for Theorem 1.1 can be found, e.g., in [3]. The theorem states that if we want to maximize P_{D} for a given P_{FA} then we have to perform a LRT with appropriate choice of the threshold γ. It can also be shown that a solution exists for every α and that solution is optimum and unique. In Chapter 2.1 we have introduced the fundamentals of LRT as a quite intuitive concept. Now Theorem 1.1 states that LRT is indeed optimum for a NP detector. Note, the case where $L(x)$ is equal to the threshold is only possible for discrete distributed random variables. This case is treated in Chapter 4 for quantized variables.

We are now returning to our mean-shifted Gauss-Gauss problem, for which we want to find the maximum P_{D} for a given $P_{\mathrm{FA}} = \alpha$ according Theorem 1.1. Therefore we first have to evaluate $l(x)$. The conditional distributions for both hypotheses are given as:

$$\begin{aligned} p(x|\mathcal{H}_1) &= \tfrac{1}{\sqrt{2\pi\sigma^2}} \exp(-\tfrac{1}{2}\tfrac{(x-1)^2}{\sigma^2}), \\ p(x|\mathcal{H}_0) &= \tfrac{1}{\sqrt{2\pi\sigma^2}} \exp(-\tfrac{1}{2}\tfrac{(x+1)^2}{\sigma^2}). \end{aligned} \tag{2.12}$$

Inserting (2.12) into (2.4) we can find the LLR

$$\begin{aligned} l(x) &= \log \frac{p(x|\mathcal{H}_1)}{p(x|\mathcal{H}_0)} \\ &= \log \frac{\exp(-\frac{1}{2}\frac{(x-1)^2}{\sigma^2})}{\exp(-\frac{1}{2}\frac{(x+1)^2}{\sigma^2})} \\ &= -\frac{1}{2\sigma^2}((x-1)^2 - (x+1)^2) \\ &= x\frac{2}{\sigma^2}. \end{aligned} \tag{2.13}$$

As we can see above in the Gaussian case $l(x)$ simplifies to a linear transformation of the observation x. According (2.1), x is equal to $\theta \in \{-1, 1\}$ corrupted by Gaussian noise and therefore the resulting distribution of x is the sum of two shifted Gaussian distributions. Due to the linear transformation (2.13) the distribution of l consist of two Gaussian distributions as well. Moreover the conditional distributions $p(l(x)|\mathcal{H}_0)$ and $p(l(x)|\mathcal{H}_1)$ are again Gaussian distributed

$$\begin{aligned} l|\mathcal{H}_0 &\sim \mathcal{N}\left(-\frac{2}{\sigma^2}, \frac{4}{\sigma^2}\right), \\ l|\mathcal{H}_1 &\sim \mathcal{N}\left(\frac{2}{\sigma^2}, \frac{4}{\sigma^2}\right). \end{aligned} \tag{2.14}$$

Figure 2.1 shows the underlying distributions.

Theorem 1.1 tells us that we favor $\mathcal{H}_1$ if $L(x)$ is greater than a particular threshold or if $l(x)$ is greater than a transformed threshold. This can be expressed by the decision function

$$\phi_{\mathrm{NP}}(x) = \begin{cases} -1, & x < \gamma' \\ 1, & x \geq \gamma' \end{cases}, \qquad \gamma' = \log(\gamma)\frac{\sigma^2}{2}. \tag{2.15}$$

Now we want to evaluate the exact value of the detection probability P_{D}. The false alarm probability P_{FA} is given in the Gaussian case:

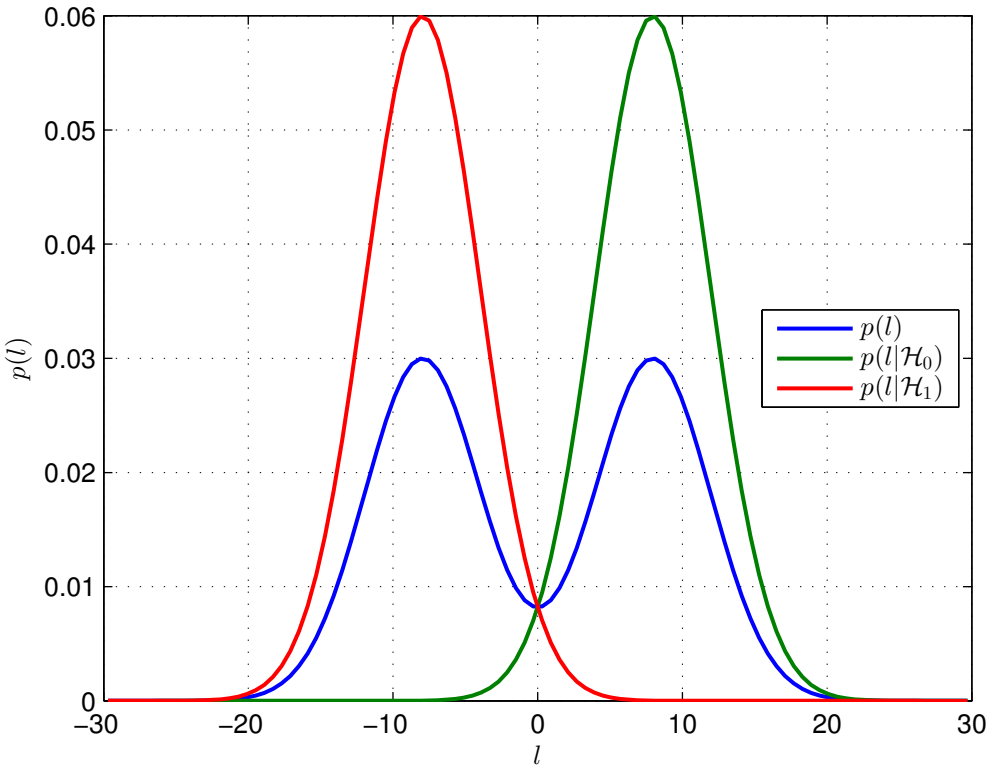


Figure 2.1: Distribution of l for the Gaussian case.

$$P_{\mathrm{FA}} = P\{x > \gamma' | \mathcal{H}_0\} = Q\left(\frac{\gamma' + 1}{\sigma}\right). \tag{2.16}$$

Expressing γ' from (2.16) and inserting into the definition of the detection probability we find:

$$\begin{aligned} P_{\mathrm{D}} &= P\{x > \gamma' | \mathcal{H}_1\} \\ &= Q\Big(\frac{\gamma' - 1}{\sigma}\Big) = Q\Big(\frac{-1 + \sigma Q^{-1}(P_{\mathrm{FA}}) - 1}{\sigma}\Big) \\ &= Q\Big(Q^{-1}(P_{\mathrm{FA}}) - \frac{2}{\sigma}\Big). \end{aligned} \tag{2.17}$$

This result can be further generalized if we introduce the *deflection coefficient* d^2

$$d^2 = \frac{(E(x|\mathcal{H}_1) - E(x|\mathcal{H}_0))^2}{\mathrm{var}(x|\mathcal{H}_0)} = \frac{4}{\sigma^2} \tag{2.18}$$

The final result for the detection probability equals

$$P_{\mathrm{D}} = Q(Q^{-1}(P_{\mathrm{FA}}) - \sqrt{d^2}). \tag{2.19}$$

2.2.2 Distributed Detection

In the previous section we have considered a detection problem with a single NP detector. Now we extend our investigation to the case where several sensors take local observations x_i and transmit a compressed version of x_i to a FC. At the FC the information of each sensor S_i is combined appropriately and an overall decision is made. The goal of this distributed NP detector is to maximize the global P_{D} while the global P_{FA} is bounded. In this thesis we present two approaches of such an implementation. The first way is to calculate local LLR at each sensor, quantize the value and transmit the obtained bit sequence to the FC, where a final decision is made. This case shall be denoted as soft decision (SD) [4]. The other case is where we locally decide if $\mathcal{H}_0$ or $\mathcal{H}_1$ is in force and transmit these local decisions to the FC where a global decision is made [1]. This shall be denoted as hard decision (HD). We will compare these two approaches later on in the simulation part. Furthermore, we have to distinguish the cases where the sensor observations x_i are conditionally independent and when there is correlation involved. In most parts of this thesis we will treat the observations x_i as conditionally independent i.e., $p(x_1, \dots x_N|\mathcal{H}) = \prod_{i=1}^{N} p(x_i|\mathcal{H})$.

Next, we give a short overview of possible configurations of sensor networks [1].

Parallel configuration. A parallel configuration consisting of N sensors with each of them taking a measurement x_i is depicted in Figure 2.2. Depending whether SD or HD is used, each sensor S_i transmits a quantized version of $l(x_i)$ or one bit based on the HD to the FC. This compressed information is denoted as u. The links between S_i shall be modeled as error-free channels for simplicity. Later in the simulation part of this thesis we will drop this simplification and model the links as binary symmetric channels (BSC) with a specific bit inversion probability P_b. The FC receives the compressed information u_i which can be collected in a vector $\mathbf{u} = (u_1\, u_2 \cdots u_{N-1}\, u_N)^T$. Based on $\mathbf{u}$, the FC

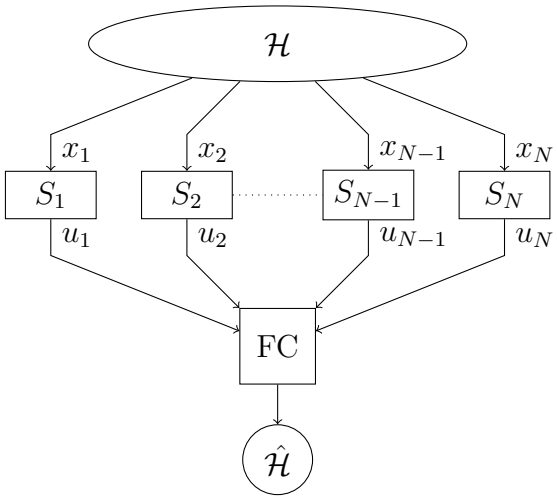


Figure 2.2: Sensor network with parallel configuration.

makes an overall decision $\hat{\mathcal{H}}$. Furthermore we assume that there is no communication between the sensors and that the sensors receive no feedback from the FC. Throughout this thesis we will use the parallel configuration for the simulation part. Therefore we will treat the SD and HD case for the parallel configuration in some more detail.

Soft or Hard decisions. In decentralized detection scenarios we wish to transmit a compressed version of x_i from S_i to the FC. As mentioned before, in the HD case each sensor just reports one bit to the FC corresponding to its local decision. It is quite intuitive that, if we compress the relevant information within more than one bit, the overall performance should be improved. Under that assumption the question arises, how to choose the partition of the outcome of each sensor to achieve maximum performance. Therefore we introduce $\mathcal{U} = \{1, \ldots, D\}$ as the set of all possible outcomes of each sensor. S_i observes x_i and decides for $u_i \in \mathcal{U}$, where u_i can be expressed using $\lceil \log_2 |\mathcal{U}| \rceil$ bits. Another question is how the FC should combine all received u_i's to maximize the overall performance. Assuming that the observations x_i are conditionally statistically independent, we can extend the observation space from Section 2.1 to $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_N$, where each $\mathcal{X}_i$ is the observation space of S_i. We are now ready to formulate P_{FA} and P_{D} for the case of a distributed NP detector [5]:

$$\begin{aligned} P_{\mathrm{FA}} &= P\{\gamma_{\mathrm{FC}}(\gamma_1(x_1)\cdots\gamma_N(x_N)) = 1 \mid \mathcal{H}_0\} \\ P_{\mathrm{D}} &= P\{\gamma_{\mathrm{FC}}(\gamma_1(x_1)\cdots\gamma_N(x_N)) = 1 \mid \mathcal{H}_1\} \end{aligned} \qquad x_i \in \mathcal{X}_i. \tag{2.20}$$

The equations above express the probability, that the FC favors $\mathcal{H}_1$ when either $\mathcal{H}_0$ or $\mathcal{H}_1$ is in force. Thereby the decision is made by γ_{FC} which incorporates all local decisions $\gamma_i(x_i)$. The function γ_{FC} is often denoted as *fusion rule*. As in the centralized case we aim to maximize P_{D} for bounded P_{FA}. The author of [5] has shown that, if we assume statistically independent observations and if there exists a global solution of the NP problem, then a local threshold test is optimum for each sensor S_i. This can be stated as follows:

$$\max \sum_{d=1}^{D} f_d P\{\gamma_i(x_i) = d \mid \mathcal{H}_1\}, \qquad \text{subject to} \quad \sum_{d=1}^{D} h_d P\{\gamma_i(x_i) = d \mid \mathcal{H}_0\} = \alpha. \tag{2.21}$$

Here, γ_i represents the optimal threshold rule for S_i. Notice the similarity to the classical NP problem, where $\mathcal{X}$ is just split into two subsets whereas here $\mathcal{X}_i$ is divided into D subsets. The scalars f_d and h_d can be considered as weights which have to be chosen appropriately to fulfill the constraint in (2.21). Note that if we set $D = 2$, $f_1 = h_1 = 0$, and $f_2 = h_2 = 1$, the problem simplifies to the classical NP detector. Assuming that every sensor decides according its own decision rule and reports its result to the FC, then the FC is confronted with a classical NP problem where the optimal solution is a threshold test as well [5].

Finding the optimal partition of $\mathcal{X}$ is somehow related to the traditional theory of quantization. Although in quantization theory we aim to minimize mean-square error, whereas here we try to find the optimal partition for maximizing P_{D}.

We can now summarize the following facts. If we assume statistically independent observations and there exist an optimal solution for the global NP problem then a threshold test with appropriate choice of γ_i at S_i and γ_{FC} at the FC is optimum. Even if the property of statistical independence is fulfilled, there may not exist a globally optimum solution for every type of problem. The author of [5] has shown that the NP problem can equivalently be considered as Lagrangian minimization problem, where

the solution space need not be convex. Although an optimal solution exist, it is highly nontrivial to find it. In fact, the weights **f** and **h** depend on the thresholds of the other sensors. So if we optimize one sensor, the weights for all the other sensors change. Therefore this ends up in an iterative optimization scheme, where it is not guaranteed that it converges to the global optimum.

In this thesis we are not trying to find a threshold test which maximizes the global P_{D}; instead we investigate the two special cases of SD and HD. As mentioned earlier, HD is the case where the partition is restricted to two disjoint sets and for SD we just quantize the local LLRs and report them to the FC.

Soft decision. Soft decision refers to the fact that every sensor S_i reports a quantized version of its test statistic, i.e., the LLR l_i, to the FC. This is in contrast to the HD case, where each sensor S_i transmits a single bit to the FC, regarding S_i favors $\mathcal{H}_1$ or $\mathcal{H}_0$. The quantization can be performed using different strategies which are discussed in the next chapter. Each sensor S_i evaluates l_i, quantizes it and passes the obtained sequence of bits on to the FC. The FC then recovers the quantized LLRs, combines them in an appropriate way, and outputs an overall decision $\hat{\mathcal{H}}$.

Now we will derive the performance of the distributed NP detector for statistically independent observations. In this case we have

$$p(x_1, \dots x_N|\mathcal{H}_l) = p(\mathbf{x}|\mathcal{H}_l) = \prod_{i=1}^{N} p(x_i|\mathcal{H}_l). \tag{2.22}$$

This assumption is commonly used in distributed detection problems where the goal is to detect a known signal. It is satisfied if the sensors are far enough away from each other, so that the sensor noise can be considered statistically independent. In contrast, the condition is violated if the sensor noise is correlated or the system aims to detect an unknown signal in noise [5].

We further assume that no quantization is performed, i.e., the FC knows all LLRs exactly. Equivalently, this can be seen as the case where the quantization is performed with infinite small quantization intervals and can be considered as an upper bound. In addition we assume for simplicity that the link between S_i and FC is error-free.

The question arises now what is the optimal way to combine the N l_i's to achieve the maximum performance. According Theorem 1.1, we know that a LRT is optimal for an NP detector. If we use the independence relation (2.22) we can develop $L(\mathbf{x})$ as follows:

$$L(\mathbf{x}) = \frac{p(\mathbf{x} \mid \mathcal{H}_1)}{p(\mathbf{x} \mid \mathcal{H}_0)} = \frac{\prod_{i=1}^{N} p(x_i \mid \mathcal{H}_1)}{\prod_{i=1}^{N} p(x_i \mid \mathcal{H}_0)} = \prod_{i=1}^{N} \frac{p(x_i \mid \mathcal{H}_1)}{p(x_i \mid \mathcal{H}_0)} = \prod_{i=1}^{N} L(x_i). \tag{2.23}$$

If we insert (2.23) into the definition of LLR (2.6) we obtain

$$l(\mathbf{x}) = \log L(\mathbf{x}) = \log \prod_{i=1}^{N} L(x_i) = \sum_{i=1}^{N} \log L(x_i) = \sum_{i=1}^{N} l(x_1). \tag{2.24}$$

We have just shown that the optimal fusion rule of a distributed NP detector is the sum of all local LLRs. Therefore we obtain the test statistic for the SD case as

$$T_{\text{sd,opt}} = \sum_{i=1}^{N} l_i. \tag{2.25}$$

It shall be emphasized that this test statistic is only optimum in the case of statistically independent observations, cf. (2.22). If there is correlation present then the optimum test statistic is not as trivial to find [6].

We now continue with our investigation of the distributed NP detector. The summation of statistical independent variables entails that the resulting pdf of the transformed variable is the convolution of the pdfs of the single variables. Moreover, the resulting conditional pdf is the convolution of the individual conditional pdfs [4].

$$p(T_{\text{sd,opt}}) = p(l_1) * p(l_2) * \cdots * p(l_N), \tag{2.26}$$

$$p(T_{\text{sd,opt}}|\mathcal{H}_0) = p(l_1|\mathcal{H}_0) * p(l_2|\mathcal{H}_0) * \cdots * p(l_N|\mathcal{H}_0), \tag{2.27}$$

$$p(T_{\text{sd,opt}}|\mathcal{H}_1) = p(l_1|\mathcal{H}_1) * p(l_2|\mathcal{H}_1) * \cdots * p(l_N|\mathcal{H}_1). \tag{2.28}$$

Now we return to our mean-shifted Gauss-Gauss problem (2.1). We have stated in the previous chapter that l_i is a linear transformation of x_i and that the corresponding

pdf $p(l_i)$ is a superposition of two Gaussian distributed variables with shifted mean and transformed variance. We can now derive the decision function

$$\phi_{\text{sd,opt}}(x) = \begin{cases} -1, & T_{\text{sd,opt}} < \gamma_{opt} \\ 1, & T_{\text{sd,opt}} \geq \gamma_{opt} \end{cases}, \qquad \gamma_{opt} = \min_{y \in T_{\text{sd,opt}}} y : p(T_{\text{sd,opt}} > y | \mathcal{H}_0) < \alpha. \tag{2.29}$$

Here, α is the size of the test and specifies an upper bound on P_{FA}. So we have extended the NP Theorem for the case of a distributed NP detector with statistically independent observations. In Chapter 3 we introduce quantization strategies and in Chapter 4 we analyze how the quantization schemes degrade the performance in comparison to the unquantized case.

Hard decision. Now we consider the case of HD in some more detail. As mentioned before, in the HD scenario each S_i makes a local decision whether $\mathcal{H}_0$ or $\mathcal{H}_1$ is in force according the NP Theorem. The result can be represented as one bit and is afterwards transmitted to the FC. Again we assume an error-free link from S_i to the FC. The FC makes an overall decision according to the so-called *K-out-of-N rule*. That means we favor $\mathcal{H}_1$ if the amount of 'one' bits received are greater or equal K, where K can be chosen appropriately [7]. Note that the K-out-of-N rule is not optimal in general, but is often used in practice for its simple structure [4]. The test statistic is given by

$$T_{\text{hd,opt}} = \sum_{i=1}^{N} u_i, \tag{2.30}$$

and the decision function can be found as

$$\phi_{\text{hd,opt}}(x) = \begin{cases} -1, & T_{\text{hd,opt}} < \gamma \\ 1, & T_{\text{hd,opt}} \geq \gamma \end{cases}, \qquad \gamma = K. \tag{2.31}$$

Next, we want to evaluate the false alarm probability at the FC, which shall be denoted as P_{FA}^{fc}. We assume that every sensor S_i exhibits the same local false alarm probability P_{FA}. According to [4], the global false alarm probability can be found as

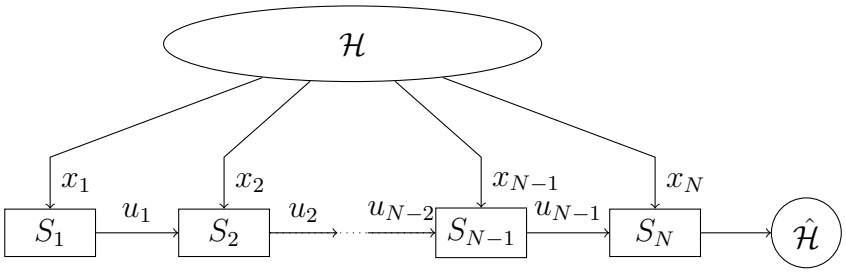


Figure 2.3: Sensor network with serial configuration.

$$P_{\mathrm{FA}}^{fc} = \sum_{i=K}^{N} \binom{N}{i} P_{\mathrm{FA}}^{i} (1 - P_{\mathrm{FA}})^{N-i} = 1 - \mathcal{B}(K-1, N, P_{\mathrm{FA}}), \tag{2.32}$$

where $\mathcal{B}$ denotes the binomial cumulative distribution function (cdf) and is given as

$$\mathcal{B}(k, n, p) = \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i}. \tag{2.33}$$

We have now derived the principle concepts of SD and HD. All further extensions are presented in Chapter 4. We continue our investigation on other configurations that are sometimes used in practice.

Serial configuration. A serial configuration also consisting of N sensors is depicted in Figure 2.3. Every sensor S_i receives the compressed information u_{i-1} from sensor S_{i-1}, which again can be a quantized version of l_i or a HD-based decision bit. In addition sensor S_i incorporates its own measurement x_i and generates compressed information u_i based on the SD or HD principle. The first sensor S_1 just takes an observation sample x_1 and pass u_1 on to S_2. The last sensor S_N can also be interpreted as a FC which makes an overall decision which hypothesis is in force.

There are several problems connected to the serial configuration. First due to the concatenation of several sensors the delay of each sensor accumulates and the latency is approximately N times bigger than in the parallel case. Another and even bigger problem is that the whole system collapses if a single link between two sensors S_i and S_{i-1} breaks. These major problems can be circumvented if we extend or change the structure of the network [1].

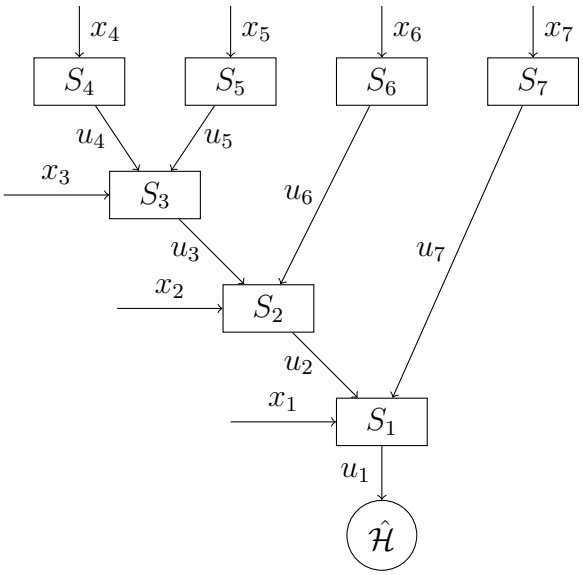


Figure 2.4: Sensor network with tree configuration.

Tree configuration. Another configuration which sometimes appears in literature is the so called *tree configuration*. In this configuration the sensors S_1 to S_N are connected to form a tree which entails that the graph contains no cycles. Indeed, there exists a unique path through the graph. One possible realization is depicted in Figure 2.4. The graph consists of two sensors S_4 and S_5 which observes x_4 and x_5. They make their own decisions u_4 and u_5 and forward them to S_3. Now S_3 receives the decisions from S_4 and S_5, incorporates its own observation x_3, makes its own decision u_3, and passes it to S_2. This principle can be extended to other sensors where S_6 and S_7 forward their own decisions and sensor S_2 and S_1 also incorporates the decisions from the antecedent sensors. S_1 can be considered as FC which may or may not take an own observation x_1 and tries to find an overall decision whether $\mathcal{H}_0$ or $\mathcal{H}_1$ is in force.

2.3 Bayesian Detector

In the case of an NP detector we only have knowledge of the conditional distributions $p(x|\mathcal{H}_i), i = \{0, 1\}$ and the hypothesis is considered as deterministic. The behavior changes if we model the hypothesis as random variable with prior probability $P\{\mathcal{H}_0\}$

and $P\{\mathcal{H}_1\}$, respectively. Furthermore, if we know the probability distribution of the priors we can use it to design a detector with better performance. This is the approach of a Bayesian detector. In typical communication scenarios the priors will be equally likely distributed, in the case of a radar or sonar problem this will not hold true. For the derivation of the Bayesian detector we have to assign costs to the four types of errors in (2.8). The costs are denoted as C_{ij} and account for $\mathcal{H}_i$ although $\mathcal{H}_j$ is in force. After assigning costs and with the knowledge of the prior probabilities we are able to define the *Bayesian risk* R_{B}:

$$R_{\mathrm{B}} = \mathbb{E}\{C\} = \sum_{i=0}^{1}\sum_{j=0}^{1} C_{ij} P(\mathcal{H}_i|\mathcal{H}_j) P(\mathcal{H}_j). \tag{2.34}$$

The Bayesian risk R_{B} is defined as the mean cost C averaged over both Θ and $\mathcal{X}$. The goal of a Bayesian detector is to minimize the Bayesian risk R_{B}. This can be expressed using the decision function $\phi(\cdot)$ as

$$\phi_{\mathrm{B}}(x) = \arg\min_{\phi} R_{\mathrm{B}}(\phi). \tag{2.35}$$

It can be shown (cf. [3]) that the detector minimizing the Bayesian risk R_{B} is again a LRT which can be found as:

$$\phi_{\mathrm{B}}(x) = \begin{cases} 0, & L(x) < \gamma \\ 1, & L(x) \geq \gamma \end{cases}, \qquad \gamma = \frac{(C_{10} - C_{00}) P\{\mathcal{H}_0\}}{(C_{01} - C_{11}) P\{\mathcal{H}_1\}}. \tag{2.36}$$

The threshold γ is fully determined by the assigned costs C_{ij} and the prior probability distribution. Now we consider the special case with

$$C_{ij} = 1 - \delta_{ij}. \tag{2.37}$$

Here, we penalize detection errors with $C = 1$ and correct detection with $C = 0$. In this case the decision function simplifies to

$$\phi_B(x) = \begin{cases} 0, & L(x) < \gamma \\ 1, & L(x) \geq \gamma \end{cases}, \qquad \gamma = \frac{P\{\mathcal{H}_0\}}{P\{\mathcal{H}_1\}}. \tag{2.38}$$

This is also known as the *Minimum Error Probability* detector and is most suitable for data transmission problems.

As in the NP case it is possible to evaluate decision rules for a distributed scenario. Indeed, finding these rules is again highly nontrivial and ends up in a complicated optimization problem. Nevertheless this problem was studied extensively in literature [1], [5]. It can be shown that under the assumption of statistically independent observations, the local decision rules simplifies to an LRT. But it is again not guaranteed that a global optimum can be achieved.

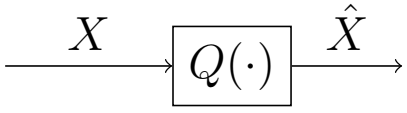
In this thesis we will mostly deal with the distributed NP case, therefore we do not explain Bayesian detection in more detail.

3

Quantizer Design

In Chapter 2 we have studied the case of SD. Every sensor S_i evaluates a local LLR value and reports it to the FC. For that reason we have to quantize the continuous LLR value l_i with d bits. Hence, l_i gets partitioned into $D = 2^d$ discrete values. This quantization can be performed using several strategies, which shall be the content of this chapter.

The setting is for every quantizer the same. We consider a discrete-time but continuous-valued input variable x with known pdf $f(x)$. The task of the quantizer Q is to map the continuous-valued input signal x to a discrete output signal $\hat{x}$. The set of outputs consists of D different values which can be represented with d bits. The question is now how to choose the mapping of the quantizer. Finding a mapping can equivalently be seen of finding the right decision boundaries g_i and appropriate reproducer values $\hat{x}_i$. If the input variable lies between g_i and g_{i+1} the quantizer maps x to $\hat{x}_i$. Next, we introduce a distortion measure D_{Q} which the quantizer tries to minimize. For every different choice of D_{Q} we obtain another quantizer strategy which entails another quantizer mapping. These strategies shall be presented in the following.



$$Q^\star = \arg\min_Q \mathbb{E}\{(X - \hat{X})^2\}$$

Figure 3.1: Lloyd-Max quantizer.

3.1 Lloyd-Max Algorithm

First we start our investigation with the very famous and widely used Lloyd-Max algorithm. The principle of Lloyd-Max quantization is based on two famous papers of Stuart P. Lloyd [8] and Joel Max [9]. As mentioned in the beginning of this chapter ever distortion measure entails another quantizer, i.e. another choice of set of decision boundaries g_i and reproducer values $\hat{x}_i$. In the case of Lloyd-Max quantization we choose the mean-square error (MSE) as distortion measure:

$$D_{\text{LM}} = \mathbb{E}\{(X - \hat{X})^2\} = \sum_{i=0}^{D-1} \int_{g_i}^{g_{i+1}} (\hat{x}_i - x)^2 p(x) dx. \tag{3.1}$$

The goal is now to find the set of decision boundaries g_i and reproducer values $\hat{x}_i$ that minimizes distortion measure (3.1). The setup is illustrated in Figure 3.1. Therefore we take the first derivative of (3.1) with respect to g_i and set it to zero

$$\frac{\partial D_{\text{LM}}}{\partial g_i} = (\hat{x}_{i-1} - g_i)^2 p(g_i) - (\hat{x}_i - g_i)^2 p(g_i) = 0, \tag{3.2}$$

where we have used the identity $\frac{d}{dt}\int_a^t f(\tau)d\tau = \frac{d}{dt}(F(t) - F(a)) = f(t)$. Rewrite (3.2) we obtain the optimal decision boundaries with respect to MSE:

$$g_i = \frac{\hat{x}_i + \hat{x}_{i+1}}{2}. \tag{3.3}$$

The optimal decision boundary g_i lies exactly in the middle of two adjacent reproducer values $\hat{x}_i$ and $\hat{x}_{i-1}$. Although we can express g_i in terms of the adjacent reproducer values we do not know them. Therefore we take the first derivative of (3.1) but now with respect to the reproducer value $\hat{x}_i$

$$\frac{\partial D_{\mathrm{LM}}}{\partial \hat{x}_i} = \frac{\partial}{\partial \hat{x}_i} \int_{g_i}^{g_{i+1}} (\hat{x}_i - x)^2 p(x) dx = 2 \int_{g_i}^{g_{i+1}} (\hat{x}_i - x) p(x) dx = 0. \tag{3.4}$$

From (3.4) we obtain the optimal reproducer value

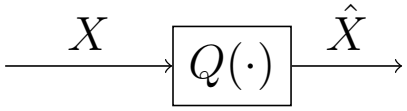
$$\hat{x}_i = \frac{\int_{g_i}^{g_{i+1}} x p(x) dx}{\int_{g_i}^{g_{i+1}} p(x) dx}. \tag{3.5}$$

Hence, the optimal reproducer value (3.5) is dependent from g_i and g_{i+1}. The resulting problem is evident. The optimal decision boundaries g_i are dependent from $\hat{x}_i$ according (3.3) and the optimal reproducer values $\hat{x}_i$ are dependent from g_i according (3.5). The solution to this problem is an iterative algorithm. We start with a guess of the reproducer values $\hat{x}_i$. Often they are assumed equally space over the whole possible range of the input signal x. After the initialization we calculate the optimal decision boundaries for our first guess of reproducer values according (3.3) which are exactly in the middle of two reproducer values. After we have obtained the decision boundaries g_i we compute again new reproducer values $\hat{x}$. The process pursues until the algorithm converges, i.e. the difference of g_i and g_{i-1} and the difference of $\hat{x}_i$ and $\hat{x}_{i-1}$ is smaller than a predefined threshold. Note the difficulty of the choice of the threshold. The smaller the threshold the more accurate is the solution for g_i and $\hat{x}_i$ but the runtime of the algorithm will increase.

If the probability distribution of the input signal $p(x)$ is not bounded, e.g. in the Gaussian case, the decision boundary g_0 is set to $-\infty$ and g_D is set to ∞. Note that the set of decision boundaries and reproducer values is only optimal for a particular distribution of X. If the distribution changes, e.g., if the variance of the input signal changes, the quantizer is not optimal anymore.

3.2 Maximum Output Entropy

In Section 3.1, the quantizer aims to minimize the MSE of quantizer input x and reproducer value $\hat{x}$. Therefore we specified the distortion measure according (3.1). Although this quantizer approach minimizes the MSE there is no reason why it should



$$Q^{\star} = \arg\max_{Q} I(X;\hat{X})$$

Figure 3.2: Maximum output entropy quantizer.

maximize the detection probability P_{D} as well. Therefore we will formulate two other quantization strategies. The first one is the maximum output entropy (MOE) principle [10]. Instead of minimizing the MSE of quantizer input and quantizer output the MOE quantizer aims to maximize the mutual information between quantizer input and quantizer output:

$$D_{\mathrm{MOE}} = I(X;\hat{X}), \tag{3.6}$$

where $I(X;\hat{X})$ specifies the mutual information of X and $\hat{X}$. The mutual information can be interpreted as the amount of information that a random variable carries about another random variable. In the case of X and $\hat{X}$ the mutual information can be written as:

$$I(X;\hat{X}) = \mathbb{E}\left\{\log\frac{p(X,\hat{X})}{p(X)p(\hat{X})}\right\} = \sum_{x}\sum_{\hat{x}} p(x,\hat{x})\log\frac{p(x,\hat{x})}{p(x)p(\hat{x})}. \tag{3.7}$$

The MOE quantizer tries to maximize the information that the output carries about the input of the quantizer. The distortion measure (3.6) can be simplified to:

$$I(X;\hat{X}) = H(\hat{X}) - \underbrace{H(\hat{X}|X)}_{=0} = H(\hat{X}), \tag{3.8}$$

where $H(\hat{X})$ denotes the entropy of the quantizer output and $H(\hat{X}|X)$ refers to the conditional entropy of the quanitzer output for fixed quantizer input. The entropy, here in terms of $\hat{X}$, is defined as the expectation of the information and is given as:

$$H(\hat{X}) = \mathbb{E}\left\{\log\frac{1}{p(\hat{X})}\right\} = \sum_{i=0}^{D} p(\hat{x}_i)\log\frac{1}{p(\hat{x}_i)} = -\sum_{i=0}^{D} p(\hat{x}_i)\log(\hat{x}_i). \tag{3.9}$$

The conditional entropy $H(\hat{X}|X)$ in (3.9) can be set to zero because a quantizer represents a deterministic device. The entropy of such device is zero because it introduces no further uncertainty [11]. This entails that maximizing the mutual information between input signal and output signal is equivalent in maximizing the entropy of the output of the quantizer. Therefore the distortion measure (3.6) can be rewritten as:

$$D_{\text{MOE}} = H(\hat{X}). \tag{3.10}$$

The quantizer scenario is illustrated in Figure 3.2. The question is now how to design the quantizer to achieve maximum entropy of the quantizer output. The authors of [11] have shown that a random variable which is uniform distributed achieves maximum entropy. Therefore the probability that the quantizer chooses $\hat{x}_i$ has to be D

$$P(\hat{X} = \hat{x}_i) = \int_{g_i}^{g_{i+1}} f(x)dx = \frac{1}{D} \qquad \text{for} \quad i = 1, \cdots, D. \tag{3.11}$$

As in the case of the Lloyd-Max quantizer (3.1) the optimal reproducer values are the centroids of the region between the decision boundaries and (3.5) simplifies to

$$\hat{x}_i = \frac{\int_{g_i}^{g_{i+1}} xp(x)dx}{\int_{g_i}^{g_{i+1}} p(x)dx} = D \int_{g_i}^{g_{i+1}} xp(x)dx. \tag{3.12}$$

In contrast to the Lloyd-Max quantizer the MOE quantizer is an information theoretic approach. However the authors of [10] and [12] have shown that the concepts of Lloyd-Max and MOE are related to each other. More precisely, both quantizer design strategies just rely on the probability distribution $p(x)$. If an MOE quantizer is designed for $p(x)$, then the Lloyd-Max quantizer achieves the same performance if the quantizer is designed for $p(x)$ scaled by an appropriate factor. Note this fact must not be true for every kind of probability distribution, but however it is true for the Gaussian case and some others [10]. Moreover, both quantizers outputs then exhibit the same entropy.

3.3 Information Bottleneck Method

The general Information Bottleneck Method (IBM) principle was introduced by Naftali Tishby in 1999 [13] and its possible applications exceed the topic of simple data quantization. The main idea of IBM is to maximize the relevant information that a random variable $X \in \mathcal{X}$ carries about another variable $Y \in \mathcal{Y}$ and compress this information within less bits as possible. The quantized variable is denoted as $\hat{X} \in \hat{\mathcal{X}}$. The observation space $\mathcal{X}$ can be either continuous-valued or discrete-valued whereas the space $\hat{\mathcal{X}}$ is discrete. $\hat{\mathcal{X}}$ can also be considered as a codebook where each entry represents a quantized value. The mapping from $\mathcal{X}$ to $\hat{\mathcal{X}}$ is modeled via a the conditional pdf $p(\hat{x}|x)$. Note that this mapping is of random nature. That means every value of x can be mapped to every value of $\hat{x}$ with some probability. This is quite different from a quantizer, where we are interested in a discrete mapping. The output variable can be expressed via the total probability theorem

$$p(\hat{x}) = \sum_{x} p(x)p(\hat{x}|x), \tag{3.13}$$

where we have assumed that the pdf of the observation $p(x)$ is known.

As mentioned before, the IBM maximizes the relevant information of Y in X on the one hand and on the other hand minimizes the rate of the quantization. The rate or equivalently the average number of bits of one entry in the codebook is lower bounded by $I(X;\hat{X})$. This is true because the average length of one codebook entry is given by $\frac{2^{H(X)}}{2^{H(X|\hat{X})}} = 2^{I(X;\hat{X})}$. $2^{H(X)}$ is the average bitlength that is needed to represent $\hat{X}$ and $2^{H(X|\hat{X})}$ can be considered as the average number of elements of $\mathcal{X}$ that are mapped to one element of $\hat{\mathcal{X}}$. Therefore if we minimize $I(X;\hat{X})$, we represent $\hat{X}$ with a minimum number of bits. Note the difference to (3.9), where mutual information simplifies to entropy of $\hat{X}$. This was true because a quantizer can be considered as a discrete mapping whereas here we have introduced a random mapping.

Beside minimizing the rate we also want to maximize the amount of information that X carries about Y. We shall briefly mention the difference to a traditionally rate distortion problem where the goal is to minimize the rate under a minimum distortion

measure. The expected distortion measure and the minimization of the rate is given as

$$\mathbb{E}\{d(X,\hat{X})\} = \sum_{x\in\mathcal{X}}\sum_{\hat{x}\in\hat{\mathcal{X}}} p(x,\hat{x})d(x,\hat{x}), \tag{3.14}$$

$$R(D_Q) = \min_{p(\hat{x}|x):\mathbb{E}\{d(x,\hat{x})\}\leq D_q} I(X;\hat{X}). \tag{3.15}$$

This minimization ends up in a converging iterative algorithm which is known as Blahut-Arimoto (BA) algorithm [14]. The BA algorithm just finds the optimal partition of $\mathcal{X}$ given the partition of $\hat{\mathcal{X}}$ and not the optimal partition of $\hat{\mathcal{X}}$ that minimizes the expected distortion. Another difficulty is to find the right distortion measure $d(X,\hat{X})$ that fits to the particular problem.

Therefore instead of utilizing the rate-distortion theory we try another approach. We assume that there exists a statistical dependence between Y and X which can be expressed by the mutual information $I(X;Y)$. As mentioned before, the main concept of IBM is to preserve as much information as possible of Y within the quantized variable $\hat{X}$ with a minimum number of bits. That means we want to maximize the preserved information $I(\hat{X};Y)$ while minimizing the rate $I(X;Y)$. The variables $Y \leftrightarrow X \leftrightarrow \hat{X}$ form a Markov chain, where the mutual information of $I(X;Y)$ is always greater or equal than $I(\hat{X};Y)$ due to the data processing inequality [11]. This is euqivalent to the fact that lossy source coding cannot increase the amount of mutual information between two variables. The optimization problem can be formulated as follow

$$\min_{p(\hat{x}|x)} I(\hat{X};X) - \beta I(\hat{X};Y). \tag{3.16}$$

The parameter β allows us to adjust the amount of preserved information. When β is set to zero, the quantization is most inaccurate and when it is set to ∞ it is most precise.

The mutual information of Y and $\hat{X}$ can be simplified as [15]

$$I(Y;\hat{X}) = I(Y;X) + \underbrace{I(Y;\hat{X}|X)}_{=0} - I(Y;X|\hat{X})$$

$$
\begin{aligned}
&= I(Y;X) - \sum_{y,x,\hat{x}} p(y,x,\hat{x}) \log \frac{p(y,x|\hat{x})}{p(y|\hat{x})p(x|\hat{x})} \\
&= I(Y;X) - \sum_{x,\hat{x}} p(x,\hat{x}) \sum_{y} p(y|x) \log \frac{p(y|x)}{p(y|\hat{x})} \\
&= I(Y;X) - \sum_{x,\hat{x}} p(x,\hat{x}) D_{\mathrm{KL}}(p(y|x)||p(y|\hat{x})) \\
&= I(Y;X) - \mathbb{E}[D_{\mathrm{KL}}(p(y|x)||p(y|\hat{x}))], \qquad (3.17)
\end{aligned}
$$

where we have used that $p(y,x|\hat{x}) = p(y|x)p(x|\hat{x})$ and $p(y,x,\hat{x}) = p(x,\hat{x})p(y|x)$ which is valid due to the Markov chain property. In (3.17) we have introduced the Kullback-Leibler divergence

$$
D_{\mathrm{KL}} = \mathbb{E}\left\{ \log \frac{p(x)}{q(x)} \right\} = \sum_{x} p(x) \log \frac{p(x)}{q(x)}, \qquad (3.18)
$$

which is a measure of the difference of two probability mass functions (pmf). Since $I(Y;X)$ is constant for given $p(y,x)$ (3.17) reduces to $-\mathbb{E}[D_{\mathrm{KL}}(p(y|x)||p(y|\hat{x}))]$.

The authors of [13] have shown that the optimal solution that fulfills minimization (3.16) satisfies the equation

$$
p(\hat{x}|x) = \frac{p(\hat{x})}{\psi(x,\beta)} \exp(-\beta \mathbb{E}[D_{\mathrm{KL}}(p(y||x)p(y|\hat{x})))], \qquad (3.19)
$$

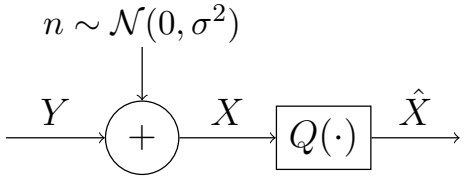
where $\psi(x,\beta)$ is introduced as normalization function. Now we are able to state the IBM algorithm

Theorem 2: The pmfs that minimize (3.16) are also minimizing the functional

$$
\mathcal{F}[p(\hat{x}|x); p(\hat{x}); p(y|\hat{x})] = I(X;\hat{X}) + \beta \mathbb{E}[D_{\mathrm{KL}}(p(y|x)||p(y|\hat{x}))], \qquad (3.20)
$$

where the minimization is performed over the convex sets of pmfs $\{p(\hat{x})\}$, $\{p(\hat{x}|x)\}$ and $\{p(y|\hat{x})\}$ according

$$
\min_{p(y|\hat{x})} \min_{p(\hat{x})} \min_{p(\hat{x}|x)} \mathcal{F}[p(\hat{x}|x); p(\hat{x}); p(y|\hat{x})]. \qquad (3.21)
$$



$$Q^\star = \arg\max_Q I(Y; \hat{X})$$

Figure 3.3: Information bottleneck quantizer.

The computation of the pmfs is carried out using the following alternating algorithm:

$$\begin{aligned} p_t(\hat{x}|x) &= \frac{p_t(\hat{x})}{\phi_t(x,\beta)} \exp(-\beta \mathbb{E}[D_{\mathrm{KL}}(p(y|x)||p(y|\hat{x}))]), \\ p_{t+1}(\hat{x}) &= \sum_x p(x) p_t(\hat{x}|x), \\ p_{t+1}(y|\hat{x}) &= \sum_y p(y|x) p_t(x|\hat{x}), \end{aligned} \tag{3.22}$$

where $\psi_t(x, \beta)$ is a normalization function as before.

Theorem 2 describes the IBM algorithm in its general form, where the mapping from X to $\hat{X}$ is modeled via the conditional pmf $p(\hat{x}|x)$. This algorithm is often used in applications like pattern recognition. However we are more interested to use the IBM algorithm for quantization problems. Here the mapping from X to $\hat{X}$ is of discrete nature and the rate of the quantizer is assumed fixed. Therefore we choose the parameter β in (3.22) as very large, which entails that $p(\hat{x}|x)$ is either one or zero. In information theoretic sense this can be seen that we try to preserve as much information as possible, while the alphabet size is kept constant. In our simulation scenario the distortion of Y is modeled as Gaussian noise with zero-mean and variance σ^2. The setup is visualized in Figure 3.3. Note that for the IBM design we have to know the variance σ^2 to obtain $p(x)$.

We have just presented the basic concept of the IBM quantizer. Some extensions for performance improvement will be made in Chapter 4.

Summarizing, we have introduced three different quantizer design strategies. The Lloyd-Max quantizer tries to minimize the MSE of the quantizer output and the quantizer input whereas the MOE and IBM are information theoretic approaches. It is not obvious which of the three quantizers performs best and achieves the highest detection probability. Therefore one part of the next chapter is the comparison of the performance of all three quantizers.

4

Simulation Results

4.1 Simulation Setup

In Chapter 2 we provided the theoretical framework for distributed hypothesis testing and in Chapter 3 we introduced three different quantizer design strategies. We are now able to discuss the simulation part. As mentioned in Chapter 2, we use the parallel configuration scenario of Figure 2.2 as simulation setup. Therefore, we will next investigate this setup in some more detail.

Most of the simulations are performed for the SD case. This configuration is depicted in Figure 4.1. In contrast to the general setting of Figure 2.2 we have restricted the simulation scenario to five branches. The sensors S_1 to S_5 are modeled as a concatenation of an additive white Gaussian noise term n_i, a device which calculates the LLR $l(x_i)$ of each branch and a quantizer which discretizes the continuous valued LLR. The discretized LLR value is denoted as $\hat{l}(x_i)$. The additive white Gaussian noise term n_i can be considered as sensor noise, which gets introduced when sensor S_i performs its measurement. The noise has zero-mean and exhibits a variance of σ_i^2. Note that the noise variance can be different for every branch. After quantization with D levels the discrete value $\hat{l}(x_i)$ is mapped to a bit sequence $\mathbf{c}_k, k \in \{1, \ldots, D\}$, where the length of $\mathbf{c}_k$ is $\lceil \log_2 D \rceil$. This sequence is reported to the FC, where the channel between quantizer and FC is modeled as a BSC with bit inversion probability P_b. Due to the

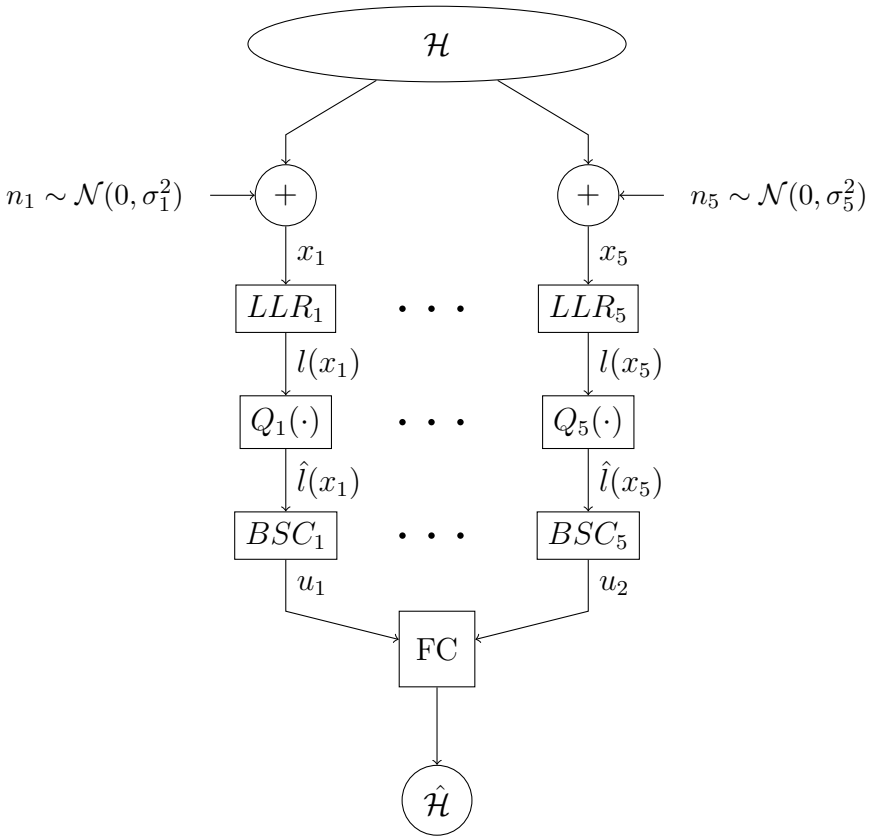


Figure 4.1: Simulation setup for the SD case.

errors introduced by the BSC, the received LLR value may be different from the transmitted one. Therefore, we denote the LLR received at the FC as $\hat{l}_{\mathrm{FC}}(x_i)$ from sensor S_i. Afterwards the FC collects all the LLRs of each branch and makes its final decision which hypothesis is in force.

Although we have chosen a simulation setup of five branches, for some simulations it will be necessary to restrict the setup to only three branches. It will be mentioned in the text when this restriction is executed.

LLR correction. Due to the fact that we have modeled the channels between quantizers and FC as BSCs with bit inversion probability P_b, the received LLR value may be different from the transmitted one. It is obvious that bit errors can degrade the performance of the whole system. Every quantized LLR value is mapped to a different bit sequence $\mathbf{c}_k$. Hence, after receiving an erroneous bit sequence this sequence is mapped back to a wrong LLR value $\hat{l}_{\mathrm{FC}}(x_i)$. The conditional pmfs of the quantized LLR and the conditional pmfs of the quantized LLR at the FC is denoted as $p(\hat{l}|\mathcal{H}_i)$ and $p(\hat{l}_{\mathrm{FC}}|\mathcal{H}_i)$, respectively. The conditional probabilities at the FC can be expressed as [4]

$$P\left(\hat{l}_{\mathrm{FC}} = \hat{l}_j|\mathcal{H}_i\right) = \sum_{k=1}^{D} P_b^{d_{j,k}}(1-P_b)^{d-d_{j,k}} P\left(\hat{l} = \hat{l}_k|\mathcal{H}_i\right), \quad j \in \{1,\ldots,D\}, \tag{4.1}$$

where D is the number of quantization intervals and $d_{j,k}$ denotes the Hamming distance between the bit sequences $\mathbf{c}_j$ and $\mathbf{c}_k$. Thus, $p(\hat{l}_{\mathrm{FC}}|\mathcal{H}_i)$ takes into account each possible transition between two bit sequences. Note that we have assumed i.i.d. channel errors, which is usually fulfilled for binary phase shift keying (BPSK) and M-ary frequency shift keying (M-FSK) [4]. Now we are able to derive the test statistic for the pmf in (4.1):

$$T_{\mathrm{sd}} = \sum_{i=1}^{N} \lambda_i, \quad \lambda_i = \log \frac{p(\hat{l}_{\mathrm{FC},i}|\mathcal{H}_1)}{p(\hat{l}_{\mathrm{FC},i}|\mathcal{H}_0)}. \tag{4.2}$$

Here, i denotes the index of the sensor and N specifies the total number of sensors. As described in Chapter 2, the summation of statistically independent variables entails

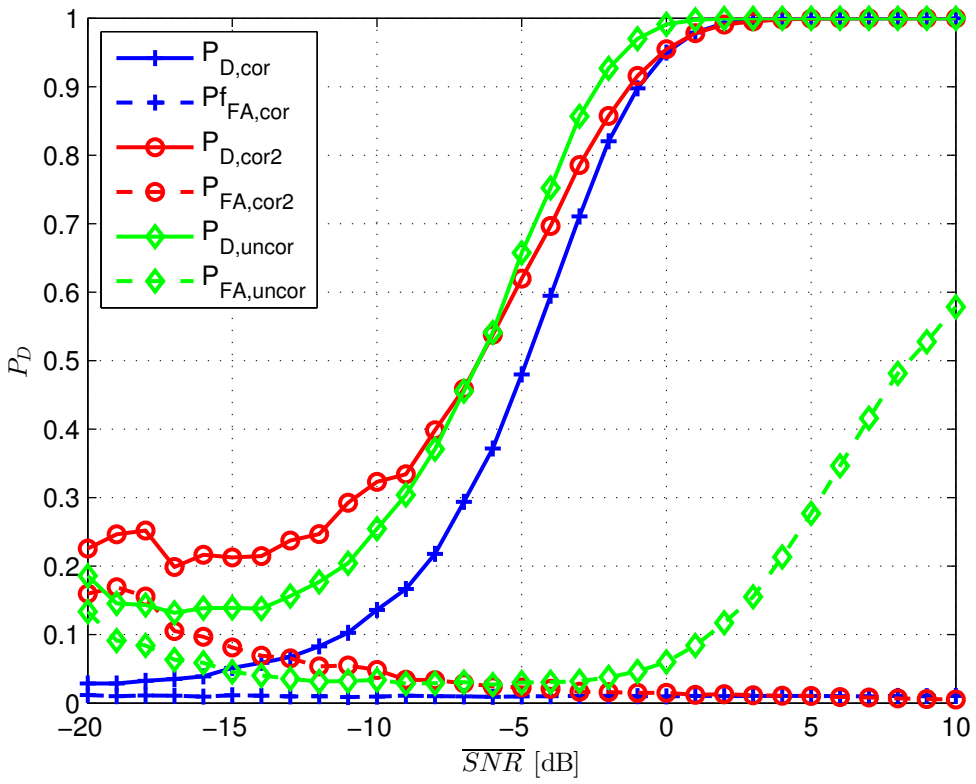


Figure 4.2: Influence of LLR correction on performance.

that the resulting pmf is just the convolution of the single pmfs. The resulting decision function can be written as:

$$\phi_{\text{sd}} = \begin{cases} T_{\text{sd}} < \gamma_{\text{sd}}, & -1 \\ T_{\text{sd}} = \gamma_{\text{sd}}, & \text{Decide}\,\mathcal{H}_1 \text{with prob}\,\eta_{\text{sd}}\ , \\ T_{\text{sd}} > \gamma_{\text{sd}}, & 1 \end{cases} \tag{4.3}$$

where the threshold is given by

$$\gamma_{\text{sd}} = \min_{z \in T_{\text{sd}}} z : p(T_{\text{sd}} > z | \mathcal{H}_0) < \alpha, \tag{4.4}$$

and the probability deciding for $\mathcal{H}_1$, when the test statistic is equal to the threshold, is

$$\eta_{\text{sd}} = \frac{\alpha - P(T_{\text{sd}} > \gamma_{\text{sd}} | \mathcal{H}_0)}{P(\gamma_{\text{sd}} = \gamma_{\text{sd}} | \mathcal{H}_0)}. \tag{4.5}$$

Here, The size of the test ϕ_{sd} is denoted by α and represents an upper bound on P_{FA}. Note the difference to the continuous case. If we are dealing with pmfs instead of pdfs, it may be possible that the test statistic is exactly equal to the threshold. If that case occurs, we decide with probability η_{sd} for $\mathcal{H}_1$. This randomization is not necessary for continuous distributions. Finally, the probability of detection can be given as

$$P_{\text{D}} = P(T_{\text{sd}} > \gamma_{\text{sd}}|\mathcal{H}_1) + \eta_{\text{sd}} P(T_{\text{sd}} = \gamma_{\text{sd}}|\mathcal{H}_1). \tag{4.6}$$

The difference of the performance when the LLR correction is applied and when not, is illustrated in Figure 4.2. Here, we have used the simulation setup from Figure 4.1 with five sensors. The variance of the sensor noise is different by 3 dB for every branch. Hence, there is a total difference of 12 dB from the first sensor compared to the last one. As quantizer strategy we have chosen the IBM method from Section 3.3 with a resolution of 3 bits per sensor. The bit inversion probability P_b is set to 0.1 and the global size α is chosen as 0.01. After the quantization the LLR value $\hat{l}$ is mapped via the natural binary coding (NBC) scheme to the bit sequence $\mathbf{c}_k$. The y axis represents the detection probability P_{D} and the false alarm probability P_{FA} and the x axis shows the median signal-to-noise ratio (SNR) value, which is equal to the SNR value of the third sensor. The SNR is defined as the ratio of the power of the signal and the power of the noise, which is given in our case as $10 \log\left(\frac{P_S}{P_N}\right) = 10 \log\left(\frac{1}{\sigma^2}\right)$. The signal power is denoted as P_S and the noise power as P_N. In our scenario P_S is given as 1. The upper solid curves show the detection probability whereas the lower dashed curves represent the false alarm probability. The two blue curves illustrates the case where LLR correction is applied. The graph indicates that the false alarm probability is bounded indeed at 0.01 and the detection probability increases with higher SNR values. The green curves show the case where no LLR correction is done at all. This means we just use the test statistic (2.25) and (2.29). This is the case when we have no knowledge of the BSC at all. Hence, we apply the wrong test statistic at the NP detector. Therefore, P_{FA} is not bounded at 0.01 and is dependent on the SNR. The red curves illustrate the case where we incorporate P_b into the test statistic, but we do not perform the LLR correction afterwards. Although the detection probabilities are

higher for the uncorrected cases we have to consider that this is only possible because we allow a higher false alarm probability. We can summarize that a realization of a NP detector for our distributed scenario is only feasible if we incorporate P_b of the BSC into the test statistic, otherwise the false alarm probability P_{FA} is not bounded at α.

4.2 Performance Evaluation

Throughout this section we investigate the performance of the hypotheses testing scenario. The simulation setup is the same as in the previous section. Changes and modifications in the setup are stated at the appropriate parts in the text. LLR correction is applied for all simulations.

Influence of BSC. As mentioned in Section 4.1, the channel between sensor S_i and the FC is modeled as BSC with bit inversion probability P_b. This is only a coarse and simple assumption and might not be true for real world channels. However, the probability that a bit error occurs is P_b and the probability that the bit is received without error is given as $1 - P_b$. The simulations are done for different bit inversion probabilities P_b and the results are shown in Figure 4.3. The blue curve represents the case where P_b is set to zero. This is equivalent to the assumption that there is no BSC at all. Further simulations were performed with higher values for P_b. We can observe that the performance degrades with increasing P_b which is a quite intuitive result. Note that although we have applied the prescribed LLR correction the performance decreases for higher P_b. So the LLR correction cannot equalize the effects of the BSC. It just incorporates the BSC into the test statistic and ensures that a NP detector with a given false alarm probability P_{FA} can be realized at all.

Number of sensors. It seems quite obvious that the performance can be improved if the number of sensors increases. Equivalently, this can be interpreted as increasing the diversity order of the system. For this reason, we choose for every sensor S_i the same noise variance to enable a fair comparison. Figure 4.4 shows the behavior of the

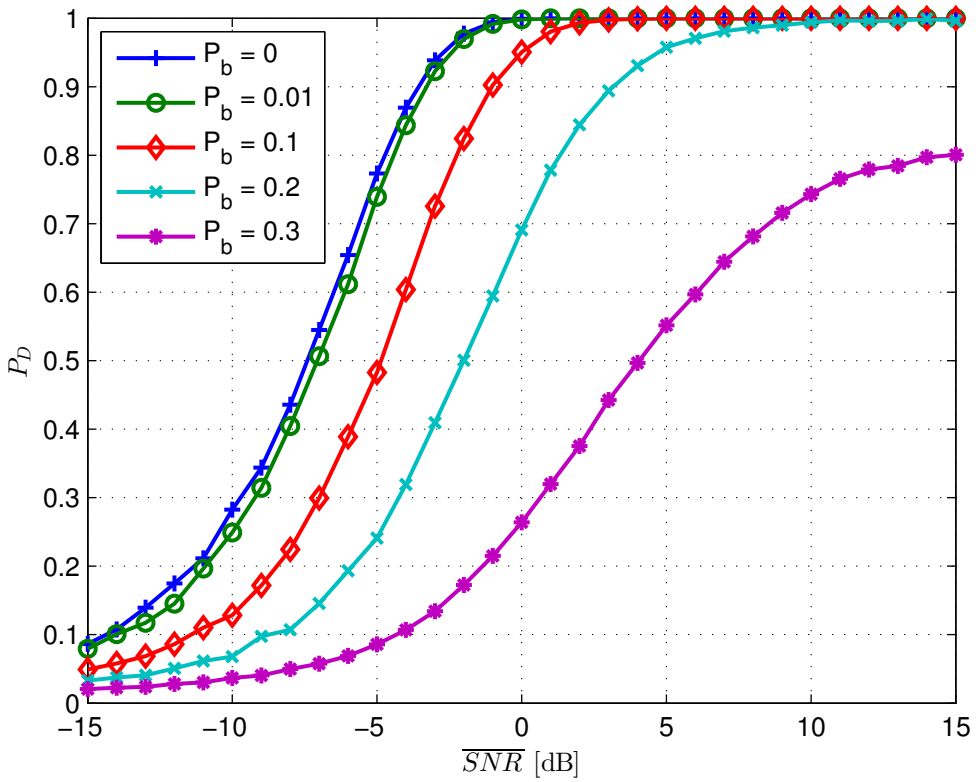


Figure 4.3: Influence of the BSC.

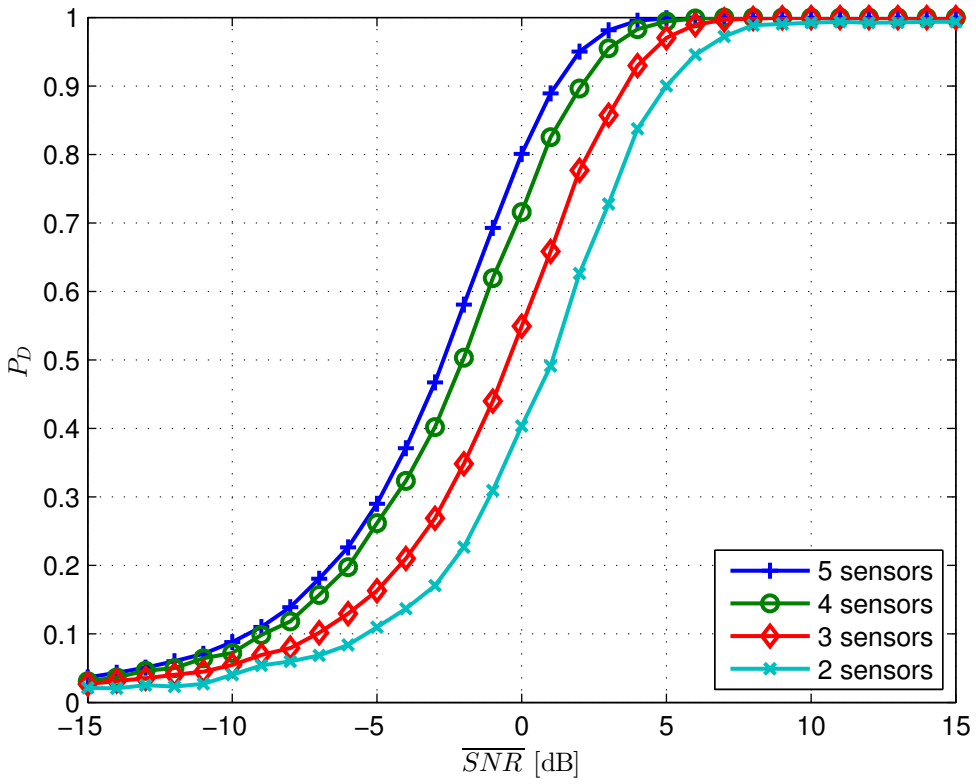


Figure 4.4: Detection probability for different number of sensors.

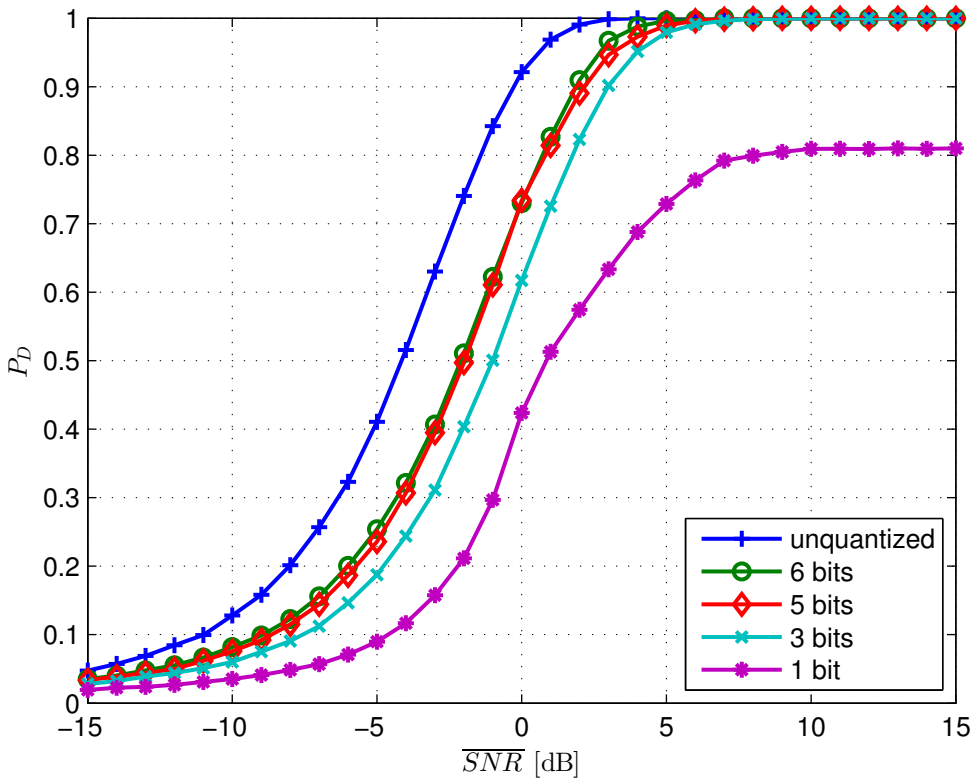


Figure 4.5: Detection probability for different quantizer resolution.

detection probability P_D with an increasing number of sensors. The graph shows that the performance improves for a larger number of sensors. Note that the computation time of the test statistic (4.2) also increases with the number of sensors.

Quantizer resolution. Next we investigate the performance for different quantizer resolutions. Because simulations for higher resolutions are computationally expensive, we restricted this simulation to three branches. Hence, the variance of sensor S_1 and S_3 is different by 6 dB. The simulation results are shown in Figure 4.5. Note that the x-axis shows the median SNR, which is in this case the SNR of the second sensor. The behavior is again quite obvious. The detection probability increases for higher resolution. According the IBM principle from Section 3.3, we can express more relevant information with a higher number of bits. Hence, we can make a more reliable decision, which yields a performance improvement. The purple curve shows the poorest performance with a resolution of 1 bit and the blue curve attains the highest detection probability

P_{D}. Note, there is a huge improvement if we increase the resolution from one bit to three bits but we also notice a saturation effect on the performance. This means the detection probability is saturating at a resolution of approximately 5 bits. Due to this saturation effect it makes no sense to increase to resolution above 5 bits. In addition, the blue curve represents the case where no quantization is considered at all and the BSC is omitted. This case is an upper bound on the overall system performance. Note, the loss of performance between the unquantized case and the quantized case with 5 bit resolution is almost introduced by the high bit inversion probability P_b of the BSC.

Simple channel coding scheme. Previously, we have seen that the performance improves if we increase the resolution of the quantizer. We know that the BSC introduces bit errors which degrades the overall performance. But a higher resolution entails a larger amount of data which has to be transmitted to the FC. Hence, the number of errors in one data word will increase as well. The question arises now which strategy performs better: Increasing the resolution of the quantizer or transmitting a quantized value with lower resolution several times. For this reason we introduce the so called *repetition code*, where a *codeword* $\mathbf{c}$ is given as:

$$\mathbf{c} = (u \underbrace{u \dots u}_{N-1 \text{ repetitions}}). \tag{4.7}$$

Here, the bit u is transmitted N times in total. Note that with an N-bit repetition code $t = \lfloor \frac{N-1}{2} \rfloor$ bit errors can be corrected, where t is called the *error-correction capability* [16]. For simplicity, we just compare a quantizer with a resolution of one bit to a quantizer with a resolution of N. In the case of the one bit quantizer the transmitted bit is repeated $N-1$ times.

Figure 4.6 shows the comparison between the quantizers with a resolution of three and five bits and the corresponding repetition codes. In the case of three bits we use a setup with five sensors and in the case of five bits a setup with three sensors. We observe that a quantizer with a resolution of one bit and a 3-bit repetition code performs slightly better than a quantizer with a resolution of 3 bits. But if we increase the resolution to 5 bits, we can compress more relevant information and attain a higher detection

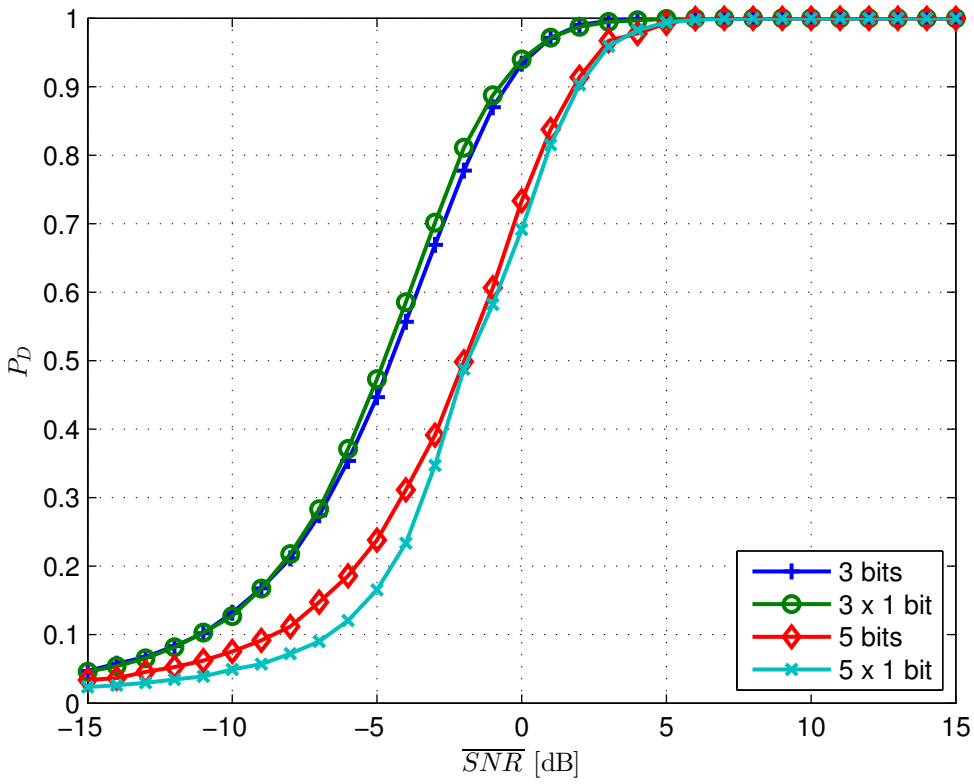


Figure 4.6: Detection probability for applied repetition code.

probability, c.f. Figure 4.5. In this case a quantizer with five bit resolution performs better than a quantizer with one bit resolution and the corresponding repetition code.

Note that before the mapping from data bit to code bit was just a one-to-one mapping, this means there was no mapping at all. But now due to the introduced channel coding scheme one data bit is mapped to N code bits. For that reason we differentiate between the bit inversion probability for the data bits $P_{b,d}$, which is a symbol error probability actually, and the bit inversion probability for the code bits $P_{b,c}$. Intuitively $P_{b,d}$ must be lower than $P_{b,c}$ due to the added redundancy bits. In fact we get for $P_{b,d}$ in case of the three bit repetition code:

$$P_{b,d} = \underbrace{P_{b,c}^3}_{\text{3 bit errors}} + \underbrace{3(1-P_{b,c})P_{b,c}^2}_{\text{2 bit errors}}. \tag{4.8}$$

In that case $t = \lfloor \frac{N-1}{2} \rfloor = \lfloor \frac{3-1}{2} \rfloor = 1$ bit can be corrected. Therefore the received bit sequence $\mathbf{c}$ is mapped to the wrong data bit if two or three code bits are received erro-

neously. If we plug in for $P_{b,c} = 0.1$, which is equal to our simulation setup assumption, the symbol error probability reduces to $P_{b,d} = 0.028$. The behavior is quite similar in the case of the five bit repetition code. The only difference is that the error-correction capability t increases to $\lfloor \frac{N-1}{2} \rfloor = \lfloor \frac{5-1}{2} \rfloor = 2$. Therefore $P_{b,d}$ changes to:

$$P_{b,d} = \underbrace{P_{b,t}^5}_{5\,\text{bit errors}} + \underbrace{5(1-P_{b,t})P_{b,t}^4}_{4\,\text{bit errors}} + \underbrace{10(1-P_{b,t})^2 P_{b,t}^3}_{3\,\text{bit errors}}, \tag{4.9}$$

and the symbol error probability reduces from $P_{b,c} = 0.1$ to $P_{b,d} = 8.56 \cdot 10^{-3}$. Although the symbol error probability decreases for both cases, Figure 4.6 shows that a quantizer with higher resolution might perform better than the repetition code for some simulation setups. Finally, note that for the LLR correction P_b has to be replaced by $P_{b,d}$ in (4.1).

Number of measurement samples. As explained in Chapter 2, it is possible to extend the hypothesis testing scenario (2.1) to the case where more than one noisy measurement sample is taken. This can be written as:

$$\begin{aligned} \mathcal{H}_0 : \quad \mathbf{x} &= -\mathbf{1} + \mathbf{n}, \\ \mathcal{H}_1 : \quad \mathbf{x} &= \mathbf{1} + \mathbf{n}. \end{aligned} \tag{4.10}$$

We assume that there is no temporal correlation involved. This means that the measurement samples are drawn from a statistically independent source. This can be expressed in terms of the conditional pdfs as:

$$\begin{aligned} p(\mathbf{x}|\mathcal{H}_1) &= p(x_1|\mathcal{H}_1)p(x_2|\mathcal{H}_1)\ldots p(x_N|\mathcal{H}_1) = \textstyle\prod_{i=1}^N p(x_i|\mathcal{H}_1), \\ p(\mathbf{x}|\mathcal{H}_0) &= p(x_1|\mathcal{H}_0)p(x_2|\mathcal{H}_0)\ldots p(x_N|\mathcal{H}_0) = \textstyle\prod_{i=1}^N p(x_i|\mathcal{H}_0). \end{aligned} \tag{4.11}$$

Here, the conditional pdfs $p(\mathbf{x}|\mathcal{H}_j)$ can be expressed as the product of the marginal pdfs $p(x_i|\mathcal{H}_j)$. If we now insert the Gaussian noise assumption into (4.11) we get the final result for the conditional pdf's:

$$\begin{aligned} p(\mathbf{x}|\mathcal{H}_1) &= \tfrac{1}{\sqrt{(2\pi\sigma^2)^N}} \exp\left(-\tfrac{1}{2}\tfrac{\|\mathbf{x}-\mathbf{1}\|_2^2}{\sigma^2}\right), \\ p(\mathbf{x}|\mathcal{H}_0) &= \tfrac{1}{\sqrt{(2\pi\sigma^2)^N}} \exp\left(-\tfrac{1}{2}\tfrac{\|\mathbf{x}+\mathbf{1}\|_2^2}{\sigma^2}\right), \end{aligned} \tag{4.12}$$

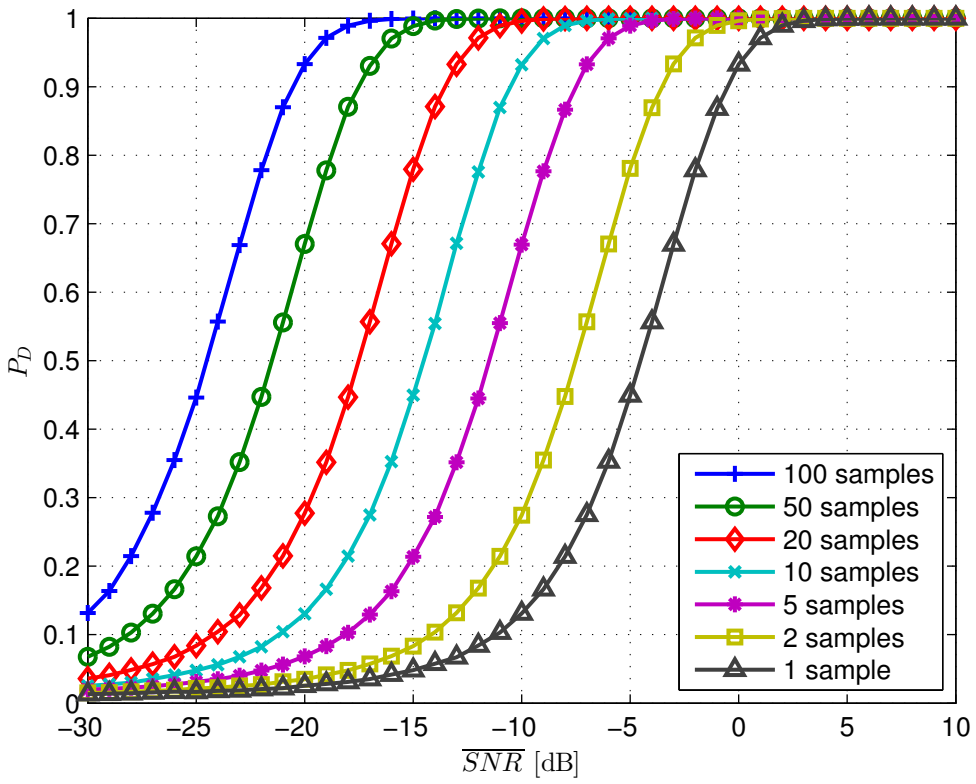


Figure 4.7: Detection probability for different number of samples.

where we have used the Euclidian norm $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{N} x_i^2}$. We are now ready to evaluate the LLR for multiple measurement samples:

$$\begin{aligned} l(\mathbf{x}) &= \log \frac{p(\mathbf{x}|\mathcal{H}_1)}{p(\mathbf{x}|\mathcal{H}_0)} \\ &= \log \frac{\exp(-\frac{1}{2}\frac{\|\mathbf{x}-\mathbf{1}\|^2}{\sigma^2})}{\exp(-\frac{1}{2}\frac{\|\mathbf{x}+\mathbf{1}\|^2}{\sigma^2})} \\ &= -\frac{1}{2\sigma^2}(\|\mathbf{x}-\mathbf{1}\|^2 - \|\mathbf{x}+\mathbf{1}\|^2) \\ &= \frac{1}{2\sigma^2}(\|\mathbf{x}\|^2 + 2\mathbf{x}^T\mathbf{1} + \|\mathbf{1}\|^2 - \|\mathbf{x}\|^2 + 2\mathbf{x}^T\mathbf{1} - \|\mathbf{1}\|^2) \\ &= 2\frac{\mathbf{x}^T\mathbf{1}}{\sigma^2} = \frac{2}{\sigma^2}\sum_{i=1}^{N} x_i = \frac{2}{\sigma^2/N}\overline{x}. \end{aligned} \tag{4.13}$$

Here we have used the norm identities $\|\mathbf{x}+\mathbf{y}\|^2 = \|\mathbf{x}\|^2 + 2\langle x, y\rangle + \|\mathbf{y}\|^2$ and $\|\mathbf{x}-\mathbf{y}\|^2 = \|\mathbf{x}\|^2 - 2\langle x, y\rangle + \|\mathbf{y}\|^2$, where $\langle x, y\rangle$ defines the inner product, which can be expressed in

a vector space as $\sum_{i=1}^{N} x_i y_i$. Note that we have expressed the result of (4.13) in three different ways, where we have introduced the *sample mean* as $\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$. This is the final result for the LLR in the case of multiple data samples under the assumption of statistically independent Gaussian noise. If we choose N as one, the LLR simplifies to (2.13) from Chapter 2. The procedure is exactly the same as before. The only difference is that we have replaced the LLR by expression (4.13). The LLR for the Gaussian case for multiple data samples simplifies to a linear transformation of the sample mean.

The behavior of the detection probability P_{D} for different number of measurement samples is illustrated in Figure 4.7. The black curve represents the detection probability for different SNR values and is identical to the curve from Figure 4.3. The performance can improved by taking more measurement samples. Indeed, the SNR improves up to 3 dB if we double the number of samples. This improvement can be achieved due to noise averaging effects. The SNR can be expressed as:

$$\mathrm{SNR} = \frac{P_S}{P_N} = \sum_{i=1}^{N} \frac{A_i^2}{\sigma_i^2} = \frac{A^2}{\sigma^2/N}, \tag{4.14}$$

where A represents the amplitude of the sample and the power P_S is given as A^2. We have assumed that the noise is zero-mean, therefore the noise power P_N is equivalent to the noise variance σ^2. Moreover we have assumed that the signal power and noise power is equal for every measurement sample. We summarize that the SNR increases linear with N, which is equivalent in a 3 dB SNR increase for doubling the samples. This means that the same detection probability P_{D} is achieved for 3 dB less SNR when the number of samples is doubled.

Comparison of different quantizers. We next consider the comparison of the three different quantizer methods from Chapter 3. Although we have introduced these strategies we can not really predict which of them performs best. The Lloyd-Max quantizer tries to minimize the MSE of the output and the input of the quantizer, whereas the MOE and IBM strategies are information theoretic approaches and minimizes the entropy of the output of the quantizer and the mutual information of the binary signal and the output of the quantizer, respectively. We can only guess that the IBM per-

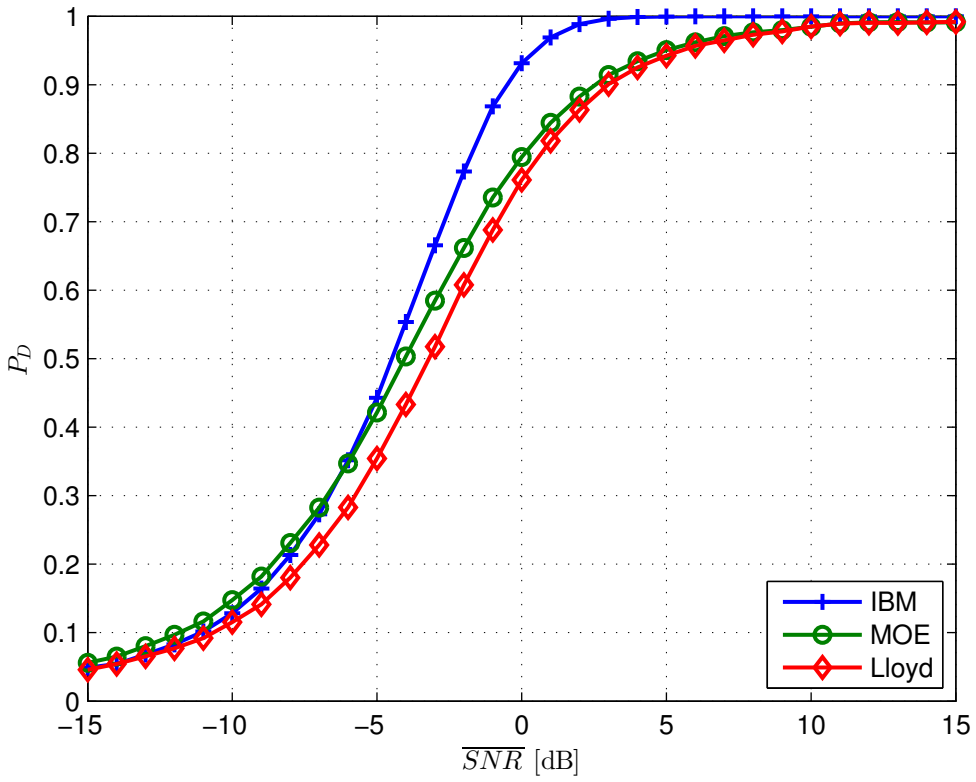


Figure 4.8: Detection probability for different quantizers.

forms better than the others because in the minimization there is also the Gaussian channel incorporated and this additional channel knowledge could improve the overall performance.

The results are shown in Figure 4.8. We have used the same simulation setup for all three quantizers as before. It shows that the IBM quantizer, which is indicated as the blue curve, performs best for most practical SNR values. Only for very low SNRs the MOE quantizer achieves a slightly better performance (green curve). The Llyod-Max quantizer (red curve) achieves the lowest performance of all three strategies. We can conclude that the IBM method, which incorporates the Gaussian noise in the quantizer design as well, achieves a better performance for such distributed detection problems.

Receiver Operating Characteristic (ROC). So far we have only analyzed the detection probability for variable SNR values and the false alarm probability was fixed at 0.01. Now we change the simulation setup and fix the SNR and evaluate the detection

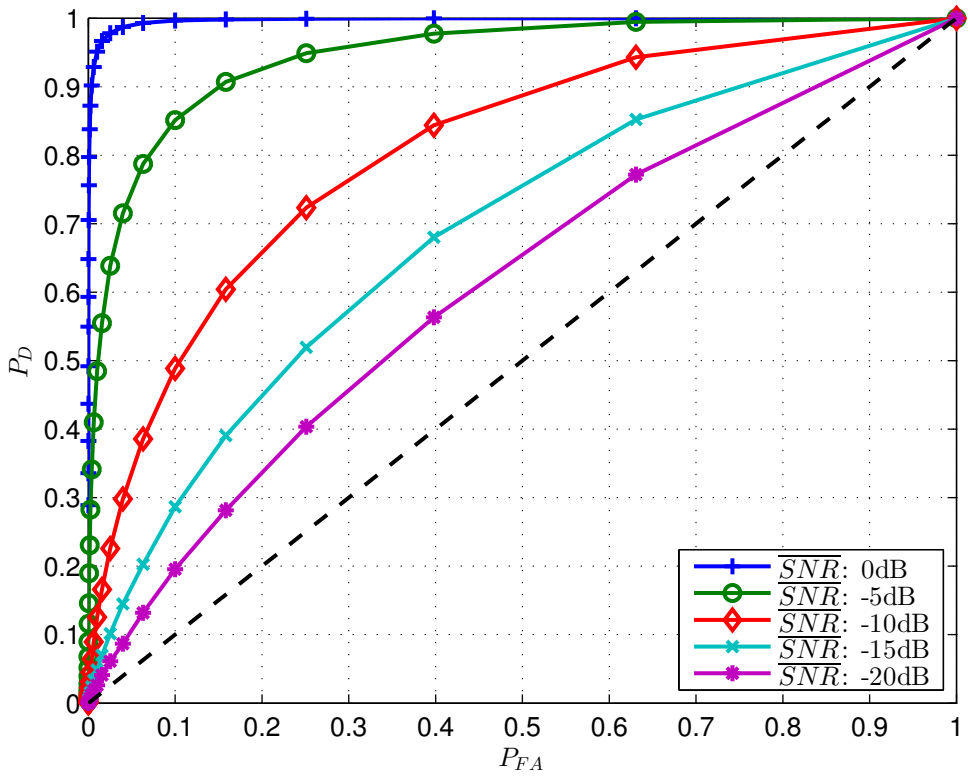


Figure 4.9: Receiver operating characteristic.

probability for variable false alarm probability for a setup with five branches and where the SNR of every branch is different in 3 dB. This leads to the ROC. The ROC is a graph, where the detection probability P_{D} is plotted versus the false alarm probability P_{FA}. The simulation results are shown in Figure 4.9. The point $(0, 0)$, where P_{D} and P_{FA} are zero, can be obtained, if we always decide for $\mathcal{H}_0$ and the point $(1, 1)$ if we always decide for $\mathcal{H}_1$. The dashed black line indicates the worst case, where the detector makes a pure guess which hypothesis is in force. That means all curves above the black line indicate detectors which perform better than a pure guess. Indeed, for higher SNR values the detector achieves a better performance.

4.3 Bayesian Detector

So far, we have performed all our simulations for the NP case. This section is devoted to the Bayesian case. Therefore we repeat some simulations from Section 4.2 for the

Bayesian detector. We have already introduced the basic concepts about the Bayesian detector in Section 2.3. Remember, in the Bayesian case we assume that we know the prior probabilities $P\{\mathcal{H}_0\}$ and $P\{\mathcal{H}_1\}$. For the following simulations we assume a communication scenario where the priors are equally likely distributed: $P\{\mathcal{H}_0\} = P\{\mathcal{H}_1\} = 0.5$. Moreover, we employ the cost function (2.37). This means an error is penalized with 1 and a correct detection is not penalized at all. Hence, the decision function can be written according (2.38) as:

$$\phi_{\mathrm{B}}(x) = \begin{cases} -1, & l(x) < 0, \\ \\ 1, & l(x) \geq 0, \end{cases} \tag{4.15}$$

where the threshold γ of (2.38) simplifies to zero because the priors are distributed equally likely. Now we extend the Bayesian detector to the case of distributed detection for statistically independent sensor noise. We can use the same test statistic as in the NP case (2.25):

$$T_{\mathrm{sd,opt}} = \sum_{i=1}^{N} l_i. \tag{4.16}$$

The decision function (4.15) is only valid for a single branch. If we apply (4.16) for N branches, the global threshold results in zero as well because the threshold of every single branch is zero. Therefore the global decision function is given as:

$$\phi_{\mathrm{B}}(x) = \begin{cases} -1, & T_{\mathrm{sd,opt}} < 0, \\ \\ 1, & T_{\mathrm{sd,opt}} \geq 0. \end{cases} \tag{4.17}$$

Note that in the case of a Bayesian detector with cost $C_{ij} = 1 - \delta_{ij}$ both error types (2.8), false alarm and miss, are treated equally. That is different to the NP case, where we bounded the amount of false alarm errors and try to minimize the amount of miss errors. Hence, P_{FA} and $P_{\mathrm{M}} = 1 - P_{\mathrm{D}}$ are equal because of our chosen cost function (2.37) and because of the symmetry of the problem. So we use as y-axis in our plots the *Bit Error Rate (BER)*, which is defined as the ratio of incorrectly detected bits divided by the total number of bits.

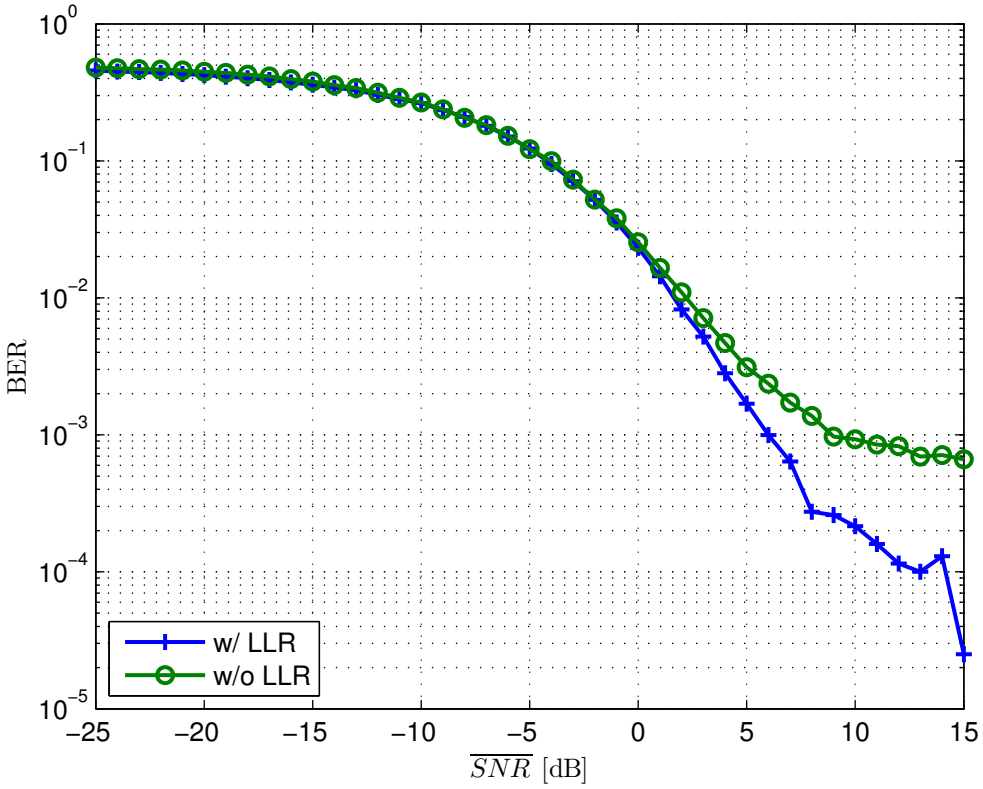


Figure 4.10: Impact of LLR correction for Bayesian detector.

Although we do not need the distribution of the global test statistic as in the NP case to find the correct decision threshold γ, we apply a LLR correction at the FC to incorporate the bit inversion probability of the BSC which improves the performance. The effect of the LLR correction is shown in Figure 4.10. The green curve indicates the case without LLR correction and the blue curve the case where the LLR correction is applied. We observe a huge performance improvement for the high SNR range. In this regime the performance is mostly determined by the bit inversion probability P_b of the BSC because the higher SNR entails that the introduced uncertainty by the sensor noise is of minor importance.

Figure 4.11 shows the BER curves for different bit inversion probabilities of the BSC. The behavior of the detector is quite intuitive. For higher values of P_b more LLR values are received erroneously. Thus, the detector decides for the wrong hypothesis and the performance degrades. The blue curve indicates the case where P_b is equal to zero and the cyan curve the case where P_b is equal to 0.1.

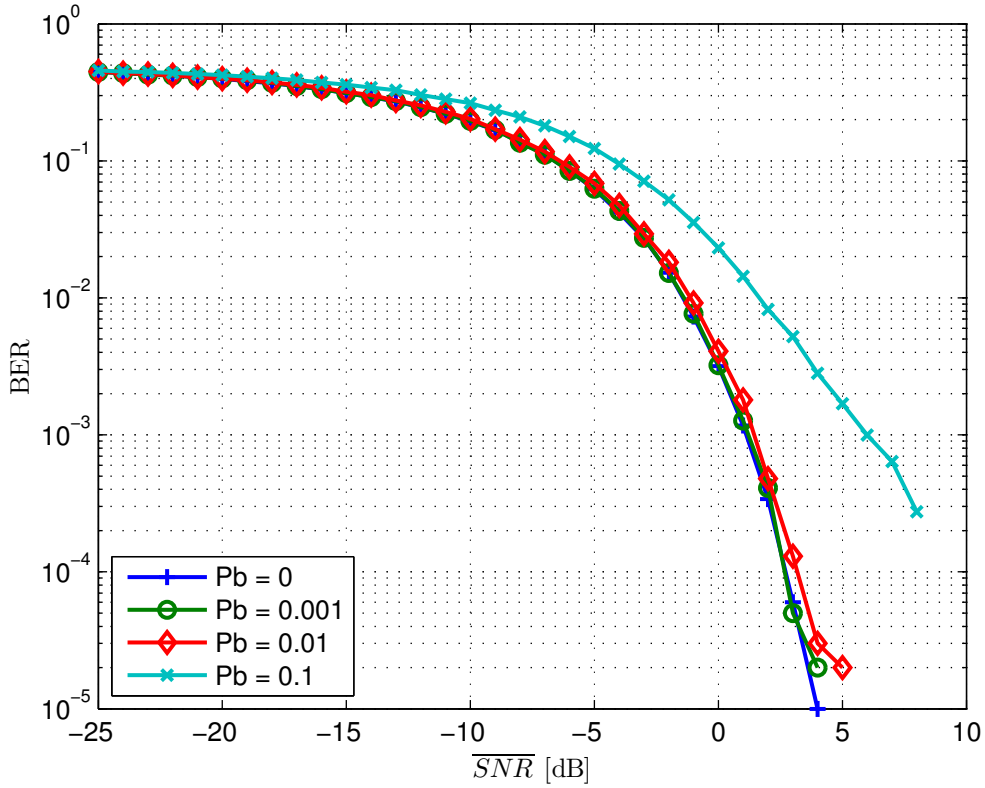


Figure 4.11: BER of Bayesian detector for different bit inversion probabilities.

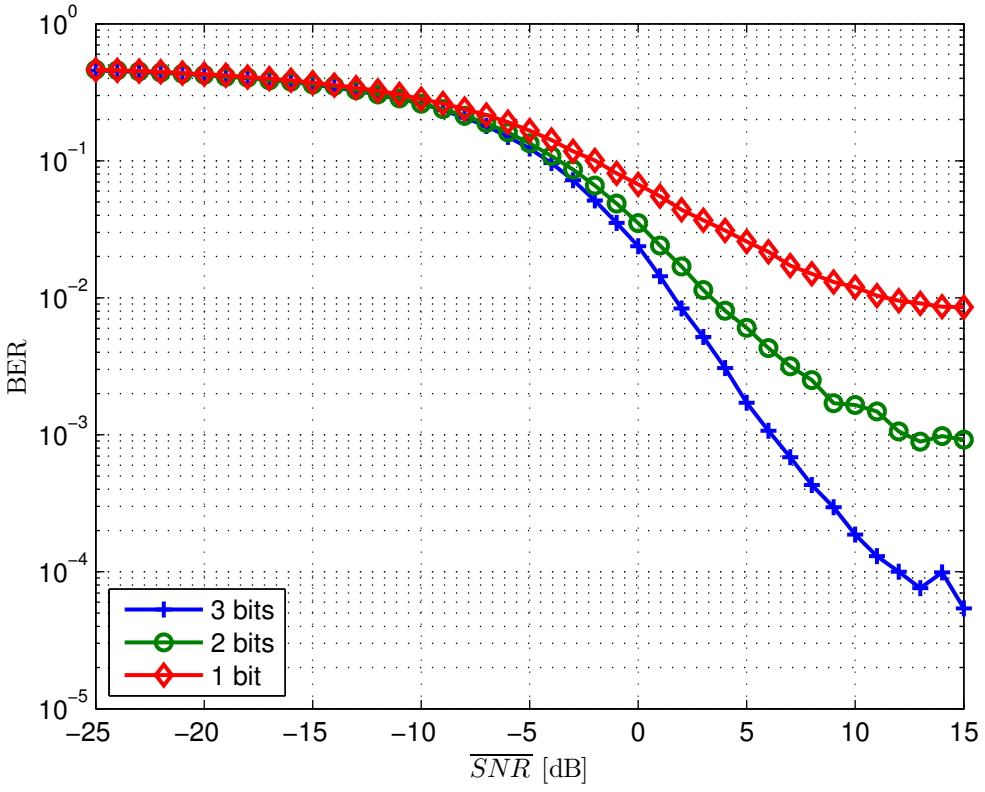


Figure 4.12: BER of Bayesian detector for different quantizer resolution.

Next, we investigated the performance for different quantizer resolutions. The behavior is shown in Figure 4.12. Here again, the behavior is similar to the NP case. For higher quantizer resolutions lower BER can be achieved. Whereas for a resolution of 1 bit only a BER of 10^{-2} can attained, for the case of a resolution of 3 bits a BER of 10^{-4} can be reached.

4.4 Hard Decision

We return to the HD scenario which we have introduced in subsection 2.2.2. As we mentioned there, in the HD case every sensor S_i performs its own NP test and reports its decision to the FC. The FC collects all the local decisions, which may be corrupted by a BSC, and makes an overall decision according the K-out-of-N rule (2.31). We stated that the K-out-of-N rule is not optimum in general but is often used in practice due to its simple structure. We developed the false alarm probability at the FC according to (2.32) as

$$P_{\mathrm{FA}}^{fc,g} = \sum_{i=K}^{N} \binom{N}{i} P_{\mathrm{FA}}^{fc,l_i} (1 - P_{\mathrm{FA}}^{fc,l_i})^{N-i}, \tag{4.18}$$

where we denote the global false alarm probability at the FC as $P_{\mathrm{FA}}^{fc,g}$ and the local false alarm probability of sensor S_i at the FC as P_{FA}^{fc,l_i}. Note the difference to Subsection 2.2.2: Due to the introduced bit errors of the BSC, the local false alarm probabilities at the sensors and at the FC are different. The question remains how to design the local NP detectors to achieve the global false alarm probability $P_{\mathrm{FA}}^{fc,g}$. Therefore we assume that all local false alarm probabilities $P_{\mathrm{FA}}^{S_i}$ at each sensor S_i are equal and denoted as P_{FA}. Furthermore we assume that each BSC exhibits the same bit inversion probability. Therefore $P_{b,i}$ is simply denoted as P_b. The local false alarm probability of each sensor at the FC can be developed as:

$$\begin{aligned} P_{\mathrm{FA}}^{fc,l} &= P(u^{fc} = 1|\mathcal{H}_0) \\ &= P(u^{fc} = 1|u^S = 1)P(u^S = 1|\mathcal{H}_0) + P(u^{fc} = 1|u^S = 0)P(u^S = 0|\mathcal{H}_0) \end{aligned}$$

$$= (1 - P_b)P_{\mathrm{FA}} + P_b(1 - P_{\mathrm{FA}}), \tag{4.19}$$

where u^S denotes the decision at the sensor and u^{fc} the received decision at the FC. The procedure is now as follows: For a given global false alarm probability $P_{\mathrm{FA}}^{fc,g}$ we can calculate the local false alarm probability $P_{\mathrm{FA}}^{fc,l}$ at the FC by reformulating (4.18). Note that (4.18) is of order N and due to the *fundamental theorem of algebra* (4.18) has exactly N solutions in $\mathbb{C}$. According the fact that the solution is a probability measure, the value must lie in the interval of $[0, 1]$. Therefore most of the solutions are not suitable. However, theoretically there can be more than one solution which achieves the same performance. After evaluating $P_{\mathrm{FA}}^{fc,l}$ we can reformulate (4.19) to obtain the local false alarm probability P_{FA} at the sensor. Because we have assumed that the sensor noise is Gaussian, we can use (2.16) to find the decision threshold for sensor S_i as:

$$\gamma_{S_i} = Q^{-1}(P_{\mathrm{FA}})\sqrt{\frac{4}{\sigma_{S_i}^2} - \frac{2}{\sigma_{S_i}^2}}. \tag{4.20}$$

Here, the noise variances $\sigma_{S_i}^2$ are just important for the calculation of the local thresholds. For the computation of the local false alarm probabilities they are immaterial. Note that there might be some combinations of $P_{\mathrm{FA}}^{fc,g}$ and P_b that permit no solution for $P_{\mathrm{FA}} \in [0, 1]$.

We now investigate the performance of the HD scheme compared to the SD scheme. As mentioned before there exist some combinations of $P_{\mathrm{FA}}^{fc,g}$ and P_b that have no solution. Hence, $P_{\mathrm{FA}}^{fc,g}$ is also determined by K, the order of the fusion rule. There might be some fusion rules which have no solution. Such a scenario is shown in Figure 4.13. Here, we compare the IBM quantizer with a resolution of one bit to the HD case where the order of the fusion rule K is varied. Note that we have chosen P_b as 0.1 and there exist no solution for $K = 1, 2$. This means it is not possible to realize a NP detector with a false alarm probability of 0.1 for that specific scenario. The blue line indicates the IBM case, which performs best of all setups. The green curve shows the case $K = 3$, which is often denoted as *majority rule*. The red curve indicates the case where $K = 4$. For lower SNR $K = 4$ performs better than the majority rule. This is due the fact that

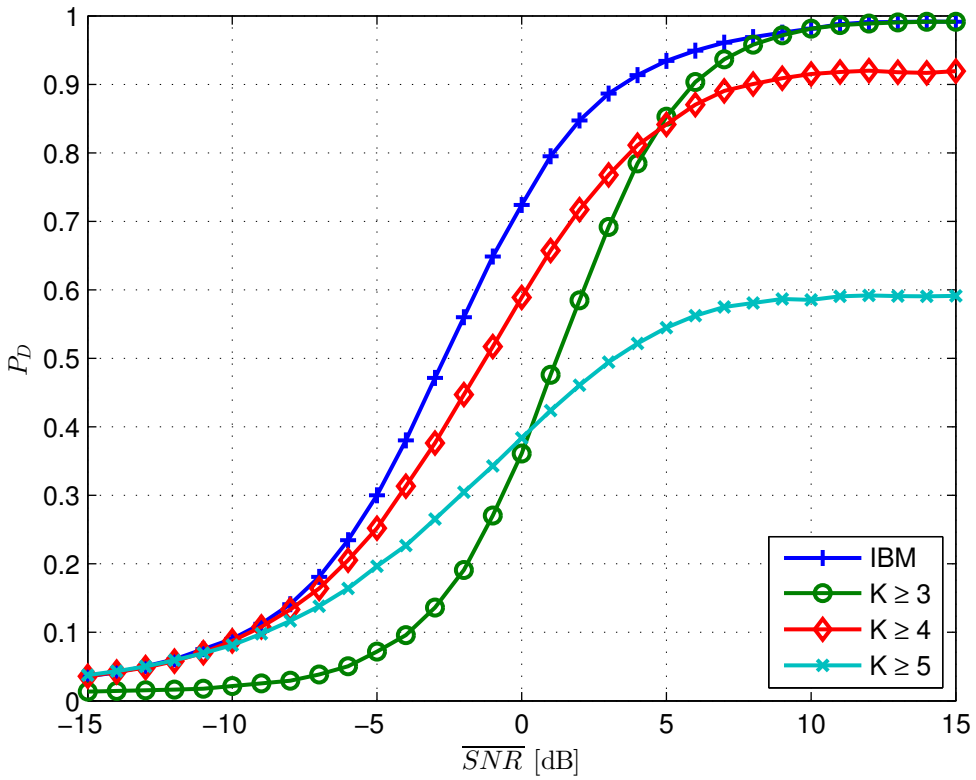


Figure 4.13: Comparison of HD with SD with $P_b = 0.1$.

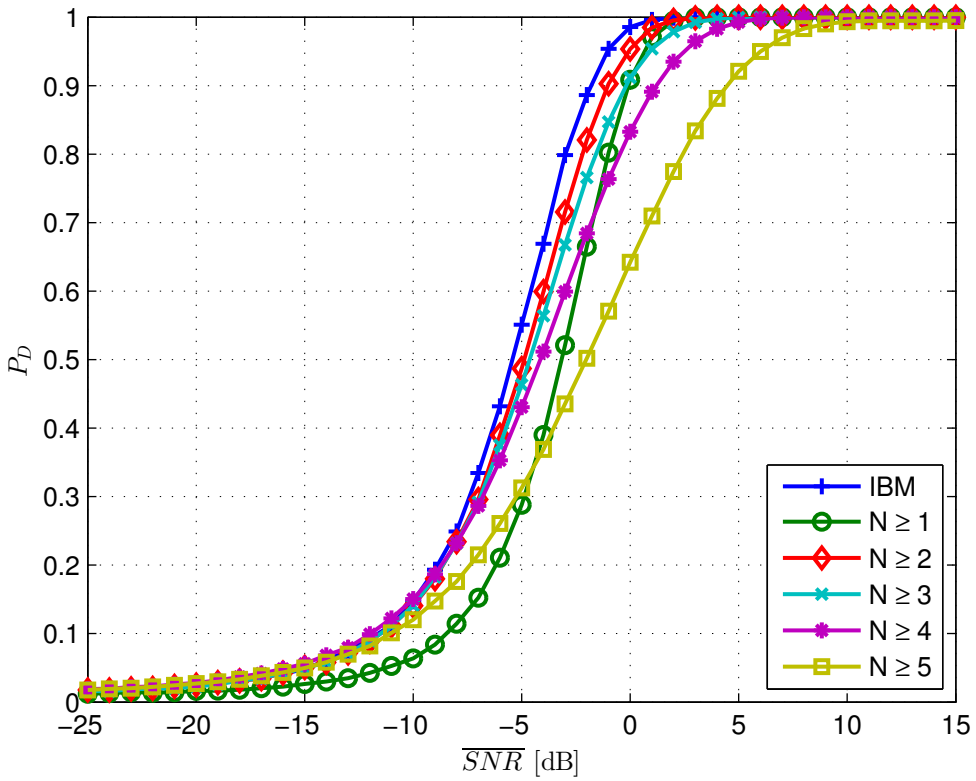


Figure 4.14: Comparison of HD with SD with $P_b = 10^{-3}$.

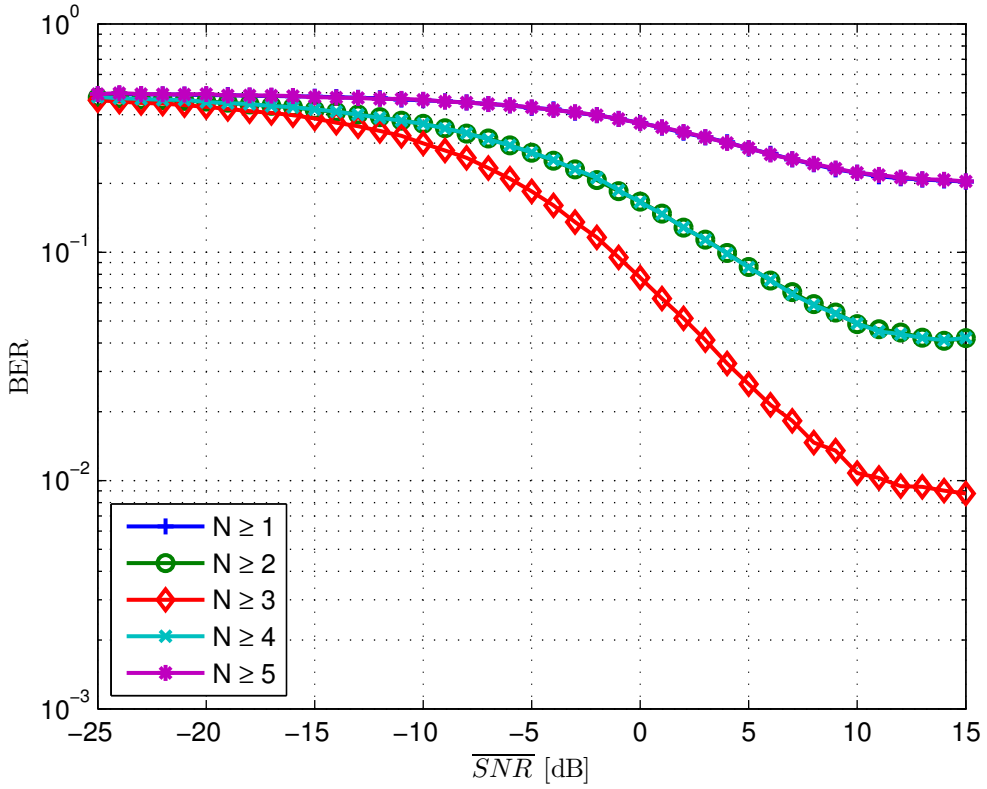


Figure 4.15: Comparison of different fusion rules for HD for the Bayesian detector.

the reported decisions from the sensors are quite unreliable and it is better to choose a fusion rule which is more conservative. This restriction can relaxed for higher SNR cases. Here, it is enough that only 3-out-of-5 received bits are 1.

The behavior changes if we lower the bit inversion probability from 0.1 to 10^{-3}. The results for this scenario are plotted in Figure 4.14. In this case the performance is mostly determined by the variance of the sensor noise and not by the introduced bit errors of the BSC. Here, the blue curve indicates the IBM case with a resolution of one bit, which performs best in most SNR ranges. Due to the low bit inversion probability it is sufficient to choose the 2-out-of-5 rule (red curve) to achieve the best performance of all hard detection rules. Summarizing, we can state that the SD case performs better than the HD case for the NP detector.

So far we have just considered the NP case for the HD scenario. Now we will apply the HD scheme also for the Bayesian detector. Remember from Section 4.3 that for the Bayesian detector both error types, false alarm and miss, are treated

equally. For the simulations the priors are assumed to be distributed equally likely: $P\{\mathcal{H}_0\} = P\{\mathcal{H}_1\} = 0.5$. First we investigate the performance of the different decision rules. We choose the bit inversion probability as 0.1. Note that due to the simpler structure of the Bayesian detector we have no problems for a bounded solution space like in the NP case. Figure 4.15 shows the results for the different fusion rules. We notice that the fusion rules for $K = 1$ and $K = 5$ and the cases where $K = 2$ and $K = 4$ achieves the same performance and the majority rule performs best. The difference to the NP case is due to the fact, that both error types are treated equally. That means both error types appear equally often and the best thing we can do, is to chose the middle value as threshold.

Figure 4.16 shows the performance evaluation for different bit inversion probabilities for the BSC. The behavior is quite obvious: The performance degrades for higher bit inversion probabilities. The blue curve indicates the case of a bit inversion probability of 0.1 and achieves the lowest detection probabilities. The red curve achieves the best performance, where the simulation was done with $P_b = 10^{-3}$.

An interesting result is shown in Figure 4.17. This simulation was done for a scenario where all sensors have the same noise variance. The plot indicates that the IBM quantizer with a resolution of 1 bit performs as good as the HD case employing the majority rule. At a first glance this result seems quite remarkable but can be explained with the introduction of a sufficient statistic [17]. In statistics every function of a vector $\mathbf{x}$ is called a *statistic* of $\mathbf{x}$. For instance the decision function (2.25) is a statistic of the LLR and therefore of the data $\mathbf{x}$ too. A statistic is *sufficient*, if no other statistic of the data provides more information about the data. In fact, the decision function (2.25) is a sufficient statistic. Note that the statistic is not sufficient anymore due to the quantization operation. Now consider the HD test statistic (2.30) applied on the Bayesian case. Here, we just take the sign of the LLR and report it to the FC. Hence, we lose information because the actual value of the LLR is not of importance. Therefore the HD decision function (2.30) is not a sufficient statistic. The behavior changes if we assume all variances of the sensors $\sigma_{S_i}^2$ are equal. That means every SNR value is quantized by the same value whereas before every bit has represented a different LLR

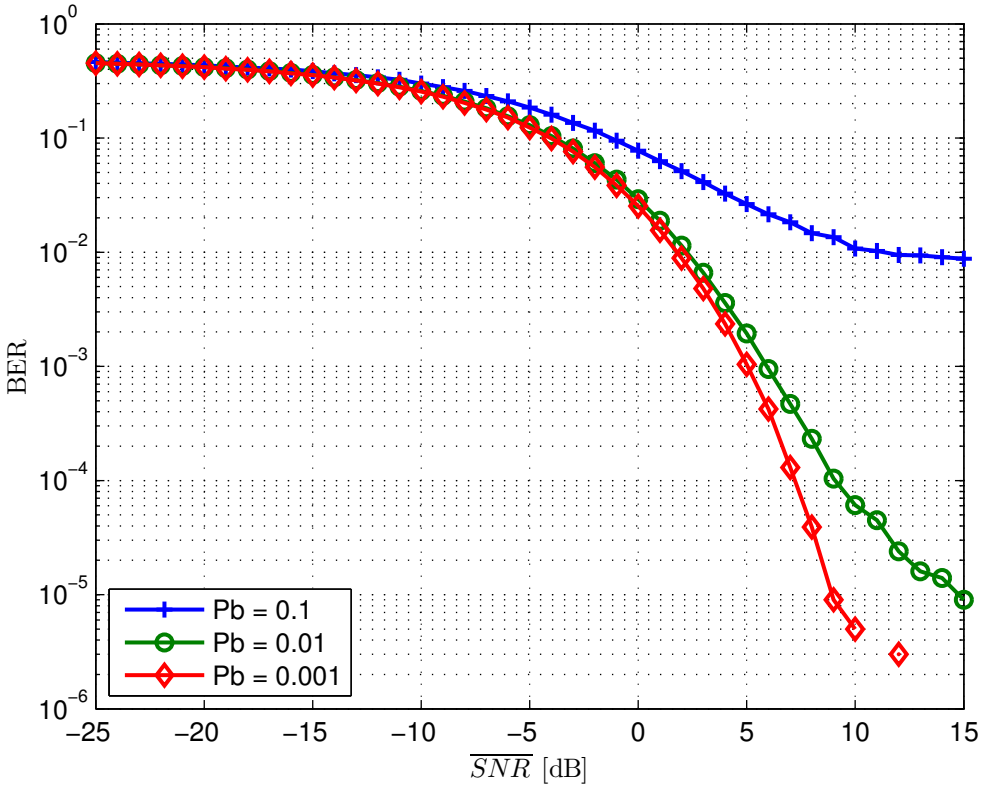


Figure 4.16: Comparison of HD for different bit inversion probabilities.

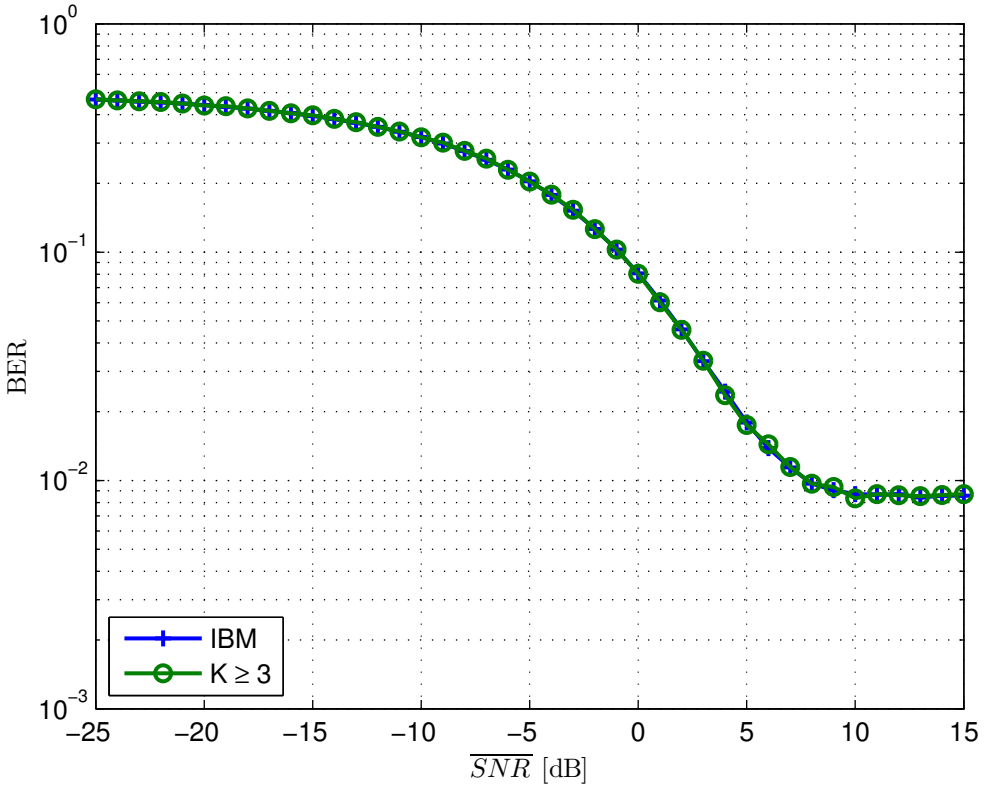


Figure 4.17: Comparison of HD and SD for equal channel SNR.

value. Hence, we lose no information if we use the HD scheme and both strategies achieves the same performance (cf. Figure 4.17).

4.5 Bit Labeling

We have mentioned in Section 4.1 that the quantized LLR value $\hat{l}$ gets mapped to a bit sequence $\mathbf{c}_k$. This sequence is then transmitted to the FC via the BSC. So far we have always chosen the NBC as bit labeling without further explanation. In this section, we investigate the issue of bit labeling and show a modification how we can improve the performance if we incorporate the bit labeling and the BSC in our quantizer design.

For our simulations we have chosen a quantizer resolution of three bits in most of the cases. Eight different LLR values can be represented with these three bits and there are $8! = 40320$ different possibilities how the assignment from these LLR values to the bit sequence can be chosen. Moreover the bit assignment has an impact on the overall performance of the whole system because every bit in each mapping has a different weight. So if a certain bit flips it can be worse for some mappings. Now summarizing, the bit mapping is a very important issue for the whole system performance, although it has nothing to do with the quantization. In general for D bits we have $2^D!$ possibilities of choosing the bit labeling. The goal is to choose the bit label that maximizes the detection probability for bounded false alarm probability:

$$\mathbf{c} = \max_{\mathbf{c}_k \in \mathcal{C}} P_{\mathrm{D}}, \qquad k = 1, \ldots, 2^D!, \tag{4.21}$$

where we have introduced the set of all possible bit labels $\mathcal{C}$. There is actually no satisfying solution for that maximization problem. One obvious solution is to try every bit label and choose that which achieves the highest detection probability. This is exactly what we have done for a quantizer resolution of 3 bits. The solution is visualized in Figure 4.18. The green curve shows the performance for the NBC labeling, what we have used so far. The purple curve represents the performance for the Gray labeling [18] and was introduced by Frank Gray 1953. The Gray mapping for three bits is depicted in Table 4.1. The main idea of the Gray code is that two neighboring values

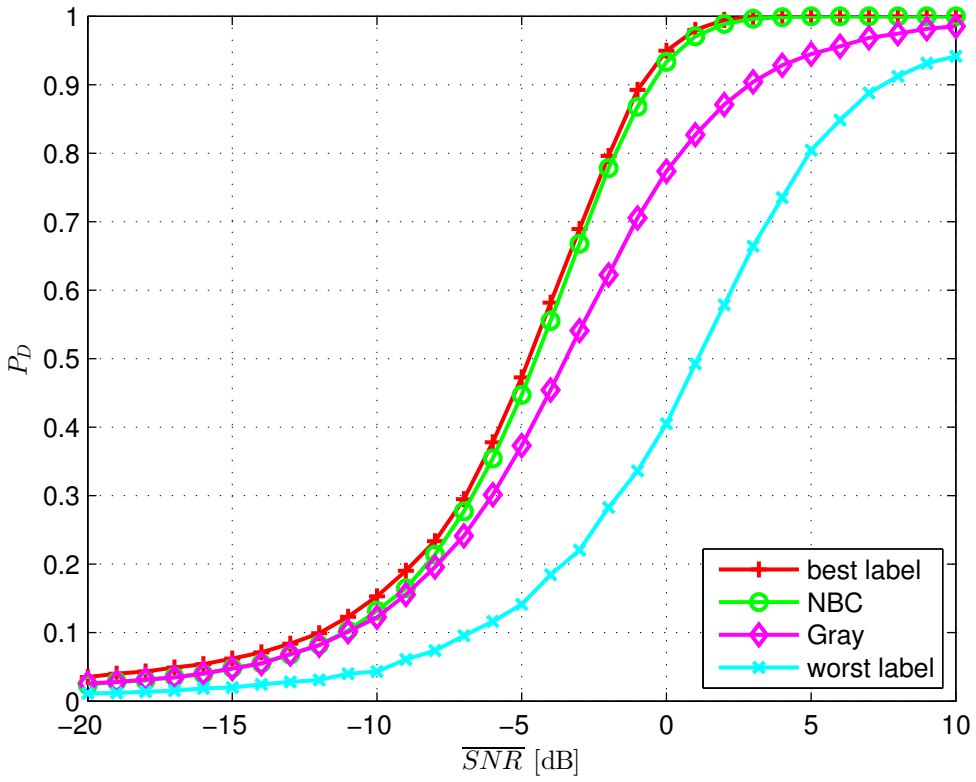


Figure 4.18: Influence of bit labeling.

are represented with bit labels that differ in just one bit. Although the Gray mapping is very famous in practice, it is not convenient in our scenarios. Gray coding is more suitable for applications where the bit sequence is mapped to a higher constellation diagram, here 8 PSK would be possible. Two adjacent symbols are then represented with a bit sequence that differ in one bit. Hence, each occurred symbol error to a neighbor symbol, which is most likely, entails just one bit error. As mentioned before we have also performed an exhaustive search over all possible bit labels for each SNR value. The bit label that achieves the highest detection probability per SNR value is indicated as the red line, whereas the label that has the worst performance is indicated by the cyan line.

Note that the optimal bit label can be different for each SNR value because the SNR defines the pdf of the LLR and a different pdf entails different reproducer values at the quantizer design. If the BSC introduces a bit error the weight is different for every SNR value. This is the reason why the optimum bit label is dependent of the

	NBC	Gray
1	000	000
2	001	001
3	010	011
4	011	010
5	100	110
6	101	111
7	110	101
8	111	100

Table 4.1: NBC and Gray mapping.

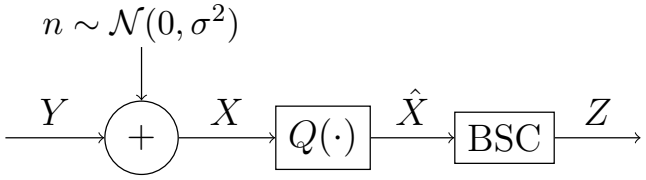
SNR value. Note that an exhaustive search over all bit labels for every SNR value is computationally expensive. In the case of a quantizer resolution of three bits we have to search over $8! = 40320$ different bit labels per SNR value. Although this search is possible, it is not feasible for higher resolutions.

Finding the optimal bit label is a difficult task but nevertheless performance critical. The authors of [19] suggested a novel approach, where the bit labeling and the BSC is already incorporated into the quantizer design.

The difference in the setup compared to the conventional IBM quantizer is illustrated in Figure 4.19. Before we tried to maximize $I(Y;\hat{X})$, whereas now we want to maximize $I(Y;Z)$ where we have incorporated the BSC into our maximization. Here, the Markov chain property is extended to the variables $Y \leftrightarrow X \leftrightarrow \hat{X} \leftrightarrow Z$. In analogy to (3.17) the mutual information between Y and Z can be developed as follows:

$$\begin{aligned} I(Y;Z) &= I(Y;\hat{X}) + \underbrace{I(Y;Z|\hat{X})}_{=0} - I(Y;\hat{X}|Z) \\ &= I(Y;X) - I(Y;X|\hat{X}) - I(Y;X|Z) + I(Y;X|\hat{X},Z) \\ &= I(Y;X) - I(Y;X|Z), \end{aligned} \tag{4.22}$$

where we exactly obtain the same result, except the term $I(Y;X|\hat{X})$ from (3.17) is replaced by $I(Y;X|Z)$. Since $I(Y;X)$ is again constant for given $p(\hat{x}|x)$ the minimization is performed with respect to $I(Y;X|Z)$, which can be expressed as



$$Q^\star = \arg\max_Q I(Y;Z)$$

Figure 4.19: Optimized information bottleneck quantizer.

$$\min_{p(\hat{x}|x)} \mathbb{E}[D_{\mathrm{KL}}(p(y|x)||p(y|z))], \tag{4.23}$$

where the objective function can be rewritten as

$$\mathbb{E}[D_{\mathrm{KL}}(p(y|x)||p(y|z))] = \sum_{\hat{x}} p(\hat{x}|x) \sum_{z} p(z|\hat{x}) D_{\mathrm{KL}}(p(y|x)||p(y|z)). \tag{4.24}$$

Now we minimize (4.24) for each $x \in \mathcal{X}$ separately. Since x is fixed we choose $\hat{x}$ such that it minimizes the second sum in (4.24):

$$\hat{x}(x) = \arg\min_{\hat{x}} p(z|\hat{x}) D_{\mathrm{KL}}(p(y|x)||p(y|z)). \tag{4.25}$$

We have found for each x the right $\hat{x}$, which is equal to the quantizer function. Since the bit labels enters into the quantizer design via $p(z|\hat{x})$ the algorithm outputs the quantizer *and* the best label which maximizes $I(Y;Z)$. Note that the iterative procedure converges to a local optimum, which can be circumvented by applying the algorithm several times and then choosing the best solution.

The simulation results are shown in Figure 4.20. The plot is similar to Figure 4.18 except that we have added the performance for the channel-optimized quantizer (4.25), which is indicated as the blue curve. As we can see, the channel-optimized quantizer performs best in most SNR regimes. Hence, incorporating the BSC into the quantizer design entails further performance improvement.

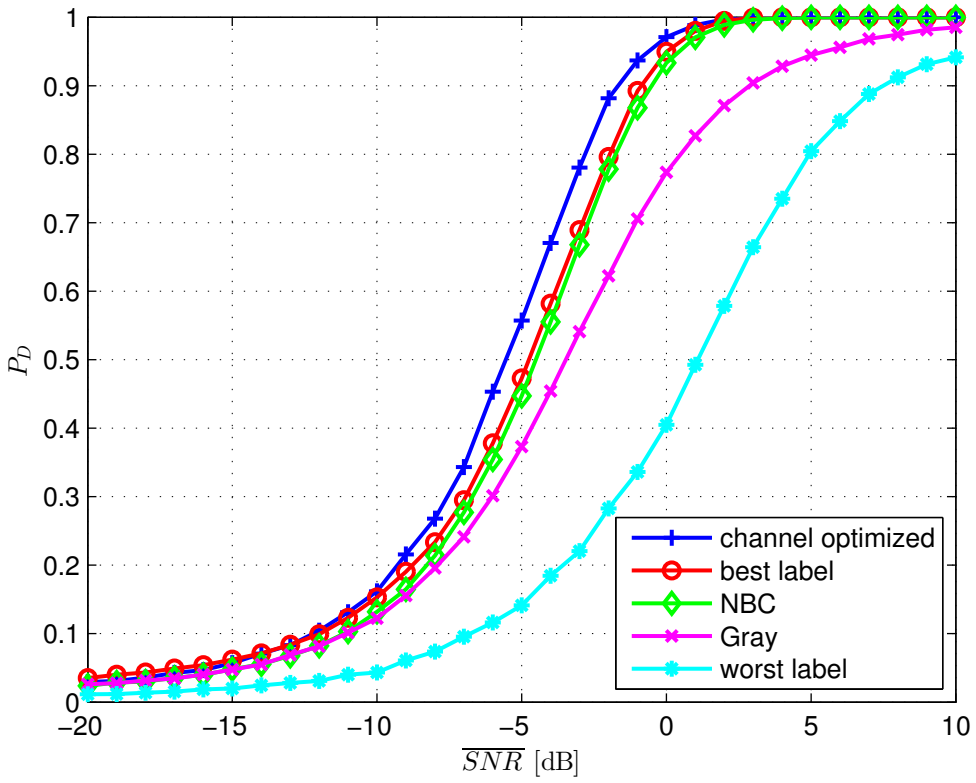


Figure 4.20: Performance of channel optimize quantizer.

4.6 Rate allocation

In the previous section, we have incorporated the BSC and the bit-labeling into our quantizer design which results in a further performance improvement. But so far we have considered the same quantizer resolution for every sensor S_i. Now we will drop this assumption. The main consideration is, does the performance improve if we allocate more bits to a sensor with better channel conditions than to an other sensor with worse channel.

3 bit allocation. We restrict our simulation setup from Figure 4.1 to three sensors and the noise variance of each sensor is different by 3 dB. We start our investigation with the assumption that three bits can be distributed to the quantizers of each sensor. Previously we have allocated one bit to each sensor. Now we try to find out if other

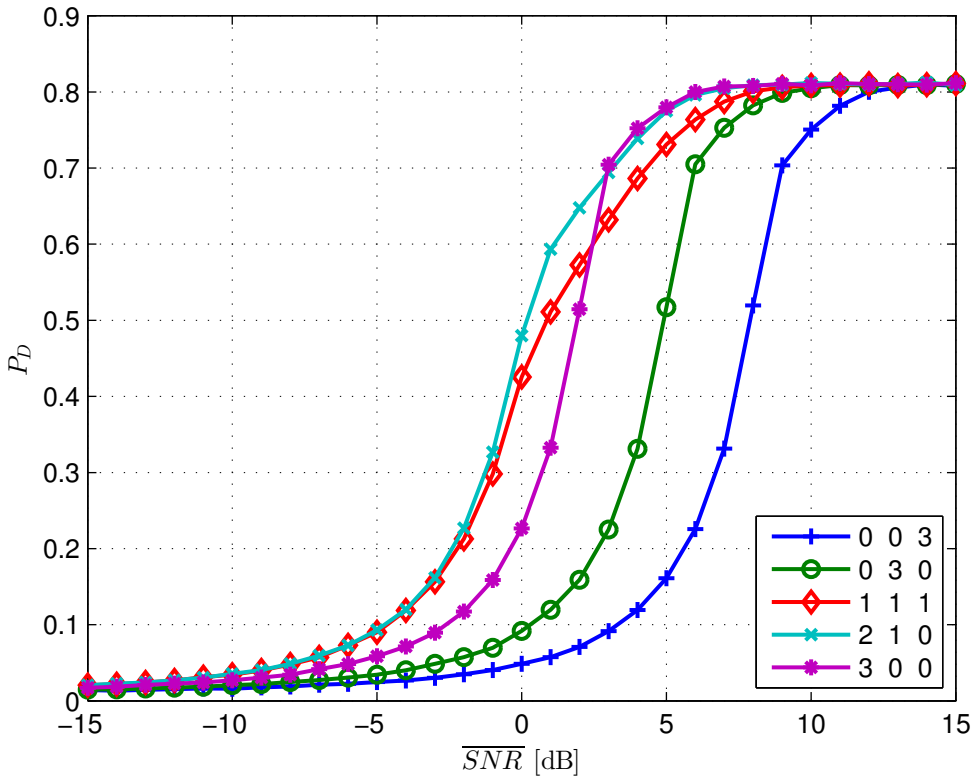


Figure 4.21: Different rate allocations.

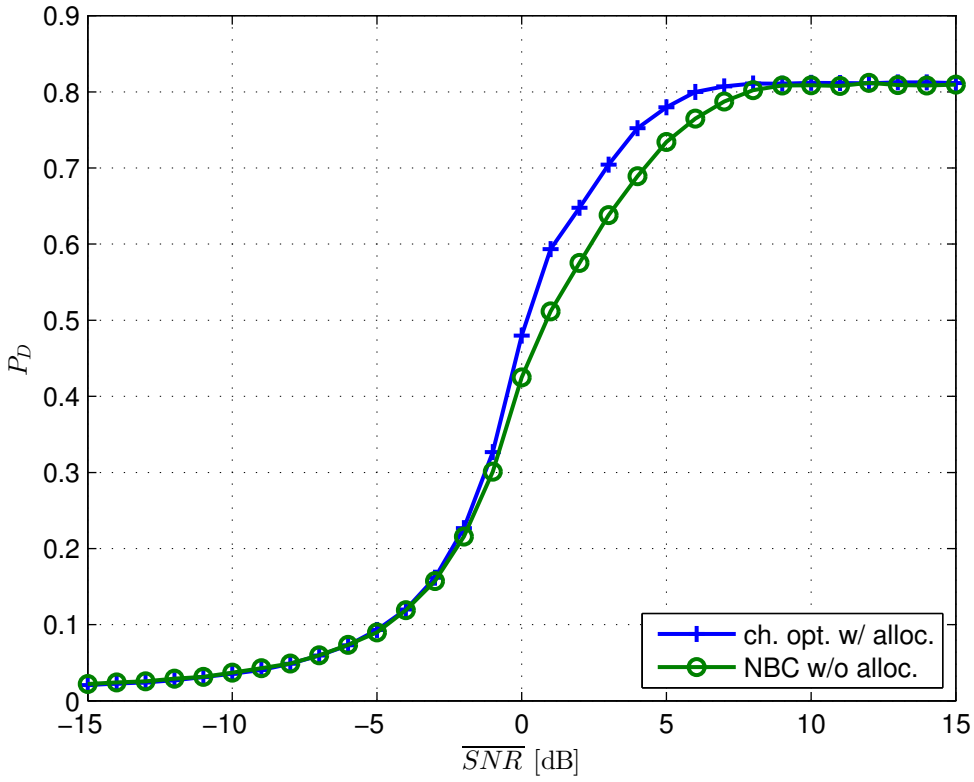


Figure 4.22: Performance comparison for 3 bits in total.

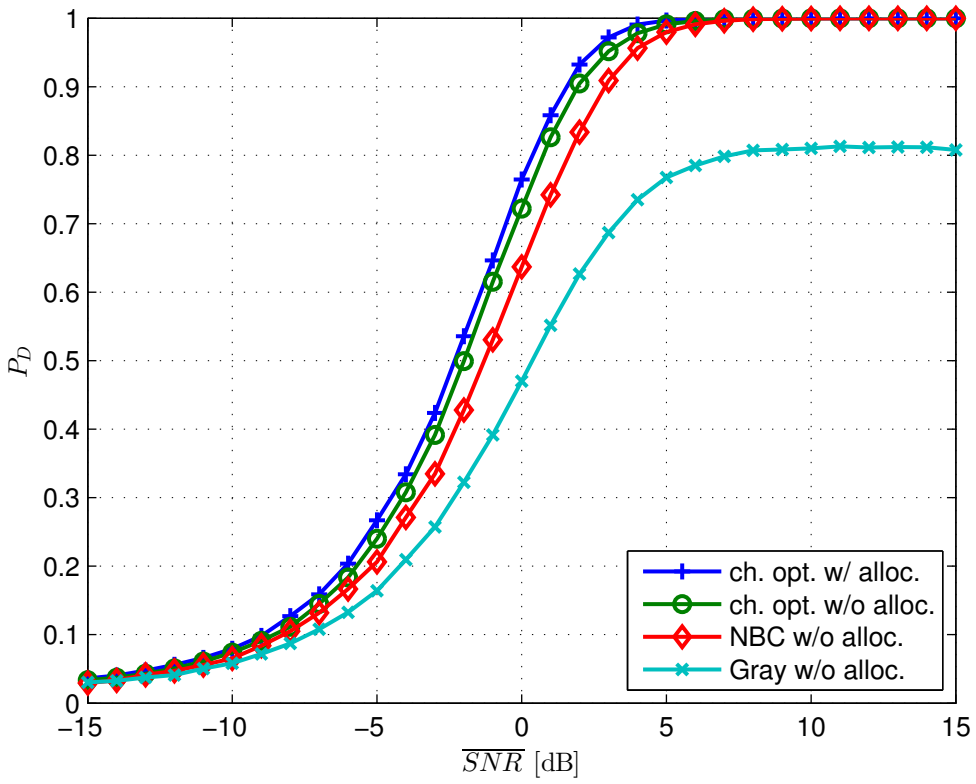


Figure 4.23: Performance comparison for 9 bits in total.

bit allocations might perform better. Therefore we try every bit allocation scenario and compare the results. The results are plotted in Figure 4.21. Every entry at the legend indicates the amount of bits per sensor, e.g., '0 0 3' means that zero bits are allocated to sensor S_1 and S_2 and three bits are allocated to sensor S_3. At low SNR regimes the sensor noise is large and therefore the measurement samples are unreliable. Hence, it is better to utilize more than one sensor to find the right decision. In these SNR ranges the uniform allocation (red curve) and the '2 1 0' allocation (cyan curve) perform similarly. But the higher the SNR gets, the more reliable the measurement samples are. Therefore it is better to choose a higher resolution for the sensor with the best channel condition. Hence, the '3 0 0' configuration (purple curve) outperforms the '2 1 0' configuration at some point. For very high SNR ranges all bit allocation scenarios perform similar and the detection probability P_D is mostly determined by the bit inversion probability P_b of the BSC.

Next we want to compare the achieved performance with the optimal rate alloca-

tion to the case with uniform allocation. The results are shown in Figure 4.22. The blue curve indicates the setup with rate allocation. Here, we have chosen the highest detection probability of Figure 4.21 for each SNR value. The green curve represents the IBM quantizer with a resolution of one bit per sensor. Note that a quantizer with one bit resolution represents the LLR with just two different values. Therefore there is no bit mapping necessary and the NBC mapping, the Gray mapping and the channel-optimized quantizer achieve all the same performance. As we can see in Figure 4.22 there is performance improvement up to 1.5 dB in some SNR ranges possible.

9 bit allocation. Now we want to repeat our simulations with nine bits in total to distribute among the sensors. The simulation results are visualized in Figure 4.23. The green curve represents the detection probability for the channel-optimized quantizer and performs better than the IBM quantizers with NBC and Gray mapping (red and cyan curve). The channel-optimized quantizer with rate allocation (blue curve) performs best. Here, we have introduced the side-constraint that the maximal resolution of a quantizer is restricted to 6 bits. This entails a huge reduction of the simulation time and furthermore according to Figure 4.5 there is only a minor performance improvement for resolutions higher than 5 bits.

4.7 NP detector for non-Gaussian noise statistics

In Section 4.2, we discovered that the IBM quantizer performs better than the Lloyd-Max and the MOE quantizer. So far all simulations were performed with Gaussian sensor noise. Now the question arises if this statement remains true for other sensor noise statistics. Therefore we consider in this section the Laplace and Cauchy distribution as sensor noise. We derive the LLRs, the pdf of the LLRs and evaluate the performance for these scenarios.

Laplace sensor noise. The Laplace distribution is also often called *double exponential distribution* because it consist of two exponential distribution which are spliced together. The Laplace distribution is often used in the field of speech recognition [20]. The pdf is given as:

$$f(x) = \frac{1}{2\lambda} \exp\left(-\frac{|x-\mu|}{\lambda}\right), \tag{4.26}$$

where μ defines the mean of the distribution and the variance is given as $\sigma^2 = 2\lambda^2$. This means the parameter λ is an indicator for the broadness of the distribution and therefore a measure for the introduced uncertainty. Now we want to derive the LLR for the Laplacian case. The conditional distributions for both hypothesis are given as:

$$\begin{aligned} p(x|\mathcal{H}_1) &= \tfrac{1}{2\lambda} \exp(-\tfrac{|x-1|}{\lambda}), \\ p(x|\mathcal{H}_0) &= \tfrac{1}{2\lambda} \exp(-\tfrac{|x+1|}{\lambda}). \end{aligned} \tag{4.27}$$

According (2.4) and (2.6) the LLR follows as:

$$\begin{aligned} l(x) &= \log \frac{p(x|\mathcal{H}_1)}{p(x|\mathcal{H}_0)} \\ &= \log \frac{\exp\left(-\frac{|x-1|}{\lambda}\right)}{\exp\left(-\frac{|x+1|}{\lambda}\right)} \\ &= -\frac{1}{\lambda}(|x-1| - |x+1|), \end{aligned} \tag{4.28}$$

where we have to distinguish between the cases $x < -1$, $-1 \leq x < 1$ and $x \geq 1$:

$$l(x) = \begin{cases} -\frac{2}{\lambda} & : \quad x < -1, \\ x\frac{2}{\lambda} & : -1 \leq x < 1, \\ \frac{2}{\lambda} & : \quad x \geq 1. \end{cases} \tag{4.29}$$

The mapping clips the LLR to the fixed values $-\frac{2}{\lambda}$ and $\frac{2}{\lambda}$ if the measurements are below -1 and above $+1$, respectively. The measurement is linearly transformed if the sample lies between the two values. We have derived the mapping from x to $l(x)$ and now we

are able to compute the conditional distributions of the LLR given both hypothesis. We start the derivation for the distribution of the measurement given the alternative hypothesis. If we consider the mapping (4.29), it is quite obvious that the probability that the LLR is equal to $\frac{2}{\lambda}$ is $\frac{1}{2}$. In the range between $-\frac{2}{\lambda}$ and $\frac{2}{\lambda}$ we perform the linear transformation and the distribution results in $\frac{1}{4}\exp\frac{l(x)-\frac{2}{\lambda}}{2}$. At last, the probability that the LLR is equal to $-\frac{2}{\lambda}$ can be found as $1-(\frac{1}{2}+\int_{-\frac{2}{\lambda}}^{\frac{2}{\lambda}}\frac{1}{4}\exp\frac{l-\frac{2}{\lambda}}{2}dl)$ because the total probability has to sum up to one. The derivation of the pdf of the measurement sample given the null hypothesis is quite similar and we get for the distributions:

$$p(l(x)|\mathcal{H}_1)=\begin{cases}\frac{1}{2}\exp(-\frac{2}{\lambda}) & : & & l(x) & = & -\frac{2}{\lambda},\\ \frac{1}{4}\exp\frac{l(x)-\frac{2}{\lambda}}{2} & : & -\frac{2}{\lambda} < & l(x) & < & \frac{2}{\lambda},\\ \frac{1}{2} & : & & l(x) & = & \frac{2}{\lambda}.\end{cases} \tag{4.30}$$

$$p(l(x)|\mathcal{H}_0)=\begin{cases}\frac{1}{2} & : & & l(x) & = & -\frac{2}{\lambda},\\ \frac{1}{4}\exp\frac{-l(x)+\frac{2}{\lambda}}{2} & : & -\frac{2}{\lambda} < & l(x) & < & \frac{2}{\lambda},\\ \frac{1}{2}\exp(-\frac{2}{\lambda}) & : & & l(x) & = & \frac{2}{\lambda}.\end{cases} \tag{4.31}$$

We have just derived the LLR mapping and the appropriate pdfs for the Laplace distributions now we are able to run our simulations.

Comparison of different quantizers. As mentioned above, we are interested in the performance of the three different quantizer strategies of Chapter 3. However, we run into problems with the MOE strategy. The quantizer tries to choose the quantizer boundaries so that every reproducer value occurs equally likely. The LLR distribution in the Laplacian case exhibits very high probability mass at the boundaries of the distribution. Therefore it is not possible to design a MOE quantizer with uniform distribution of the output variable. For that reason we omitted the MOE quantizer for the Laplacean case. For the Lloyd-Max and IBM quantizer we choose our general setting from Figure 4.1 except that we replace the Gaussian noise by the Laplacian noise and the previous LLR mapping by (4.29). The results are shown in Figure 4.24. The blue

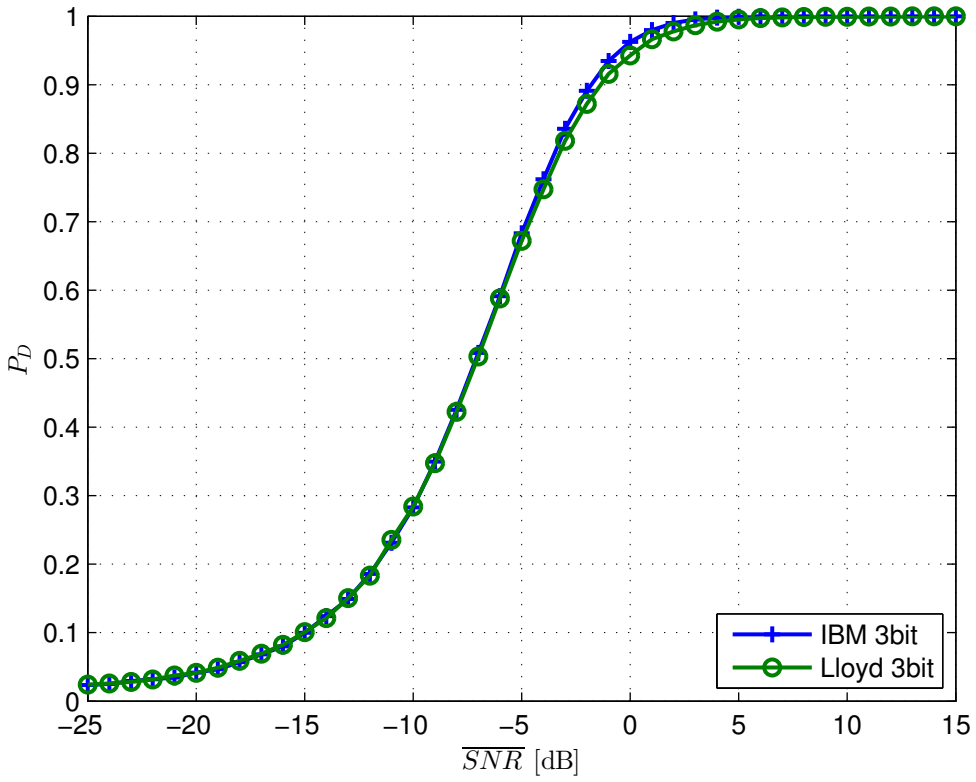


Figure 4.24: Detection probability for different quantizers with Laplace noise.

curve indicates the IBM quantizer, which performs slightly better as the Lloyd-Max quantizer (green curve). The Lloyd-Max quantizer achieves a better performance than in the Gaussian case (cf. Figure 4.8). This can be explained due to the fact that the conditional probability functions (4.30) and (4.30), which are needed for the quantizer design, are bounded. The restriction of the pdf affects positive for the Lloyd-Max principle and therefore the Llyod-Max quantizer achieves a better performance.

Cauchy sensor noise. The Cauchy distribution is often used in the field of physics and is also known as *Lorentz distribution*. The pdf is given as:

$$f(x) = \frac{1}{\pi} \frac{s}{(x + x_0)^2 + s^2}, \tag{4.32}$$

where the parameter s is often called *scale parameter* and x_0 the *location parameter* [21]. Here, x_0 specifies the location of the maximum of the pdf and s is the half-

width at the half-maximum of the distribution and hence, $2s$ is the full-width of the half-maximum. Therefore s can be interpreted as an indicator of the broadness of the distribution. The Cauchy distribution is one example of a *pathological* distribution, since mean, variance, and higher order moments do not exist. However, a derivation of the LLR is possible. For this reason we need again the conditional distributions of the data given the hypothesis, which are given as:

$$\begin{aligned} p(x|\mathcal{H}_1) &= \tfrac{1}{\pi}\tfrac{s}{(x+x_0)^2+s^2}, \\ p(x|\mathcal{H}_0) &= \tfrac{1}{\pi}\tfrac{s}{(x-x_0)^2+s^2}. \end{aligned} \tag{4.33}$$

The LLR follows as:

$$\begin{aligned} l(x) &= \log \frac{p(x|\mathcal{H}_1)}{p(x|\mathcal{H}_0)} \\ &= \log \frac{(x+1)^2+s^2}{(x-1)^2+s^2}, \end{aligned} \tag{4.34}$$

where the expression of the LLR cannot further simplified. Finding a compact analytical closed-form solution for $p(l(x)|\mathcal{H}_0)$ and $p(l(x)|\mathcal{H}_1)$ is not an easy task and was done by simulation numerically. But we can analyze the LLR mapping (4.34) in some more detail. More precisely, we are interested in the maximum and minimum of the mapping to run our simulations. For that reason we build the first derivative of (4.34) $\frac{\partial l(x)}{\partial x}$, which we denote as $l'(x)$. The expression can be found after some derivation as:

$$l'(x) = 4\frac{s^2-(x^2-1)}{(x^2-1)^2+s^2(x^2+1)+s^4}. \tag{4.35}$$

We set (4.35) to zero to find the position of the extrema:

$$l'(x) = 0: \qquad x^2-1-s^2 = 0 \quad \rightarrow \quad x = \pm\sqrt{1+s^2}. \tag{4.36}$$

If we plug in the solution (4.36) into the expression for the LLR (4.34), we obtain the extrema as:

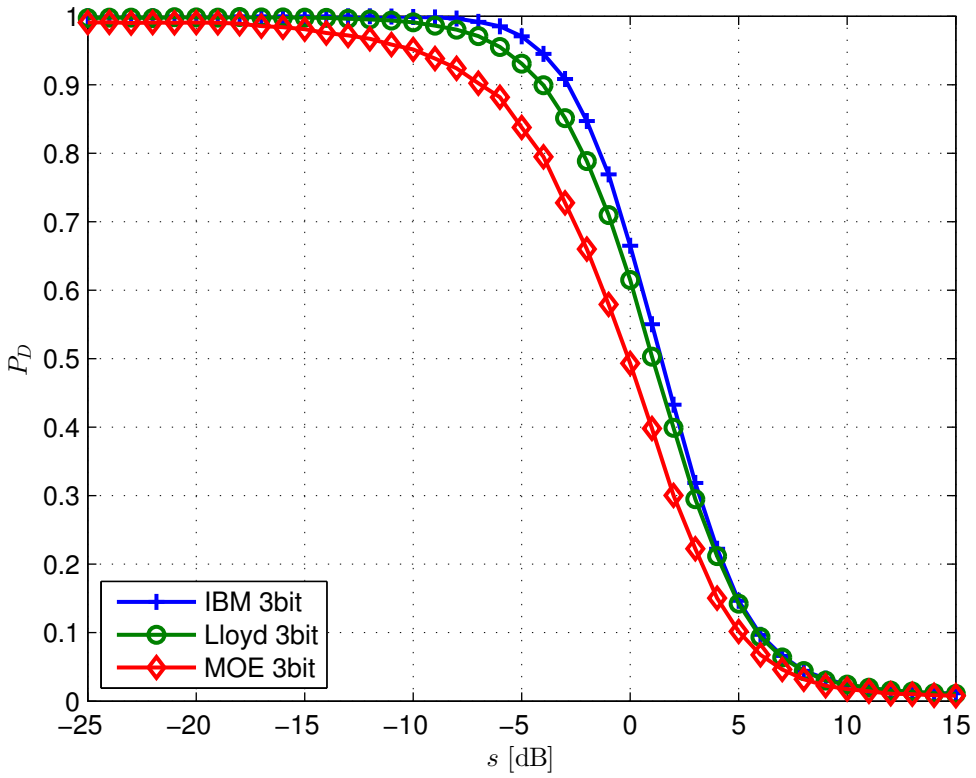


Figure 4.25: Detection probability for different quantizers with Cauchy noise.

$$l_{\text{ext1}}(x) = \log \frac{1+\sqrt{1+s^2}+s^2}{1-\sqrt{1+s^2}+s^2},$$
$$l_{\text{ext2}}(x) = \log \frac{1-\sqrt{1+s^2}+s^2}{1+\sqrt{1+s^2}+s^2}, \tag{4.37}$$

depending on the value of s which of them is the maximum and which the minimum. As (4.36) and (4.37) indicate the LLR of the Cauchy distribution is bounded between these two extrema. Therefore, also the conditional pdfs $p(l(x)|\mathcal{H}_1)$ and $p(l(x)|\mathcal{H}_0)$ are defined over a restricted region. We are now able to design quantizers according all three strategies from Chapter 3 and compare the results.

Comparison of different quantizers. The simulations were done with the same setup as before, except that the variance σ^2, equivalently the SNR, is replaced by s

because the variance is not defined at all. The results are plotted in Figure 4.25. As mentioned before, the parameter s is an indicator for the broadness of the distribution. Hence, the higher s the more unreliable is the taken measurement sample. This is contrary to the Gaussian case, where a high SNR indicates that the measurement sample is quite reliable. As for the other noise distributions the IBM quantizer (blue curve) achieves the highest performance of all quantizer strategies. The Lloyd-Max quantizer (green curve) and the MOE quantizer (red curve) perform worse.

In this subsection we have formulated the LLR computation for Laplace and Cauchy noise and applied it our distributed hypothesis testing scenario. The results of Figure 4.24 and Figure 4.25 showed that if the conditional pdf's $p(l(x)|\mathcal{H}_1)$ and $p(l(x)|\mathcal{H}_0)$ are bounded the Lloyd-Max quantizer achieves the same detection probability as the IBM quantizer. In the Gaussian noise case the IBM approach performs best. Therefore, we can summarize that the IBM quantizer achieves the best performance of all other not depending on the underlying noise statistics.

4.8 IBM quantizer for correlated sensor noise

So far we have always assumed that the sensor noise is statistically independent:

$$p(x_1, \ldots, x_N|\mathcal{H}_l) = p(\mathbf{x}|\mathcal{H}_l) = \prod_{i=1}^{N} p(x_i|\mathcal{H}_l). \tag{4.38}$$

This assumption was very convenient and we have argued that the condition of statistically independence is fulfilled if the sensors are far enough away from each other. In this section we drop this assumption and introduce correlation between the noise samples. This has two major consequences for our implementation:

- The test statistic (4.3) has to be generalized to correlated measurement samples. Especially the local LLR definition (2.4) and (2.6) has to be extended to a global definition because due to the introduced correlation the summation of the local LLRs is not optimum anymore. Hence, we calculate the LLR at the FC center and the sensors just reports the raw data sample to the FC.

- The quantizer design has to be extended to a global design strategy to incorporate the correlation between the sensors.

We note that our goal in this Section is to design local quantizers for correlated noise samples and not a vector quantizer. The difference is that a vector quantizer quantizes a higher dimensional continuous variable to a higher dimensional discrete variable. In contrast we aim to design local scalar quantizers which quantize a one-dimensional continuous variables to a one-dimensional discrete variables, although the single variables are correlated with each other. Both strategies incorporate the correlation between the variables and for the design we need to know the joint probability distribution.

We choose in this section as quantizer the IBM strategy since the single IBM can be extended to the multivariate information bottleneck method (MIBM) [22]. Similar to the original IBM it is possible to apply the MIBM for quantizer design to design single quantizers for correlated noise samples. Although the MIBM is valid for arbitrary high dimensional random variables we restrict in this section our setup to two dimensional variables. Hence, our setup from Figure 4.1 is restricted to two branches. Throughout this section we assume that the bit inversion probability of the BSC is zero or equivalently, we omit the BSC at all to keep the simulation scenario simple. Further we return from the Cauchy noise and the Laplace noise to our Gaussian noise assumption but extended to two dimensional Gaussian noise. The joint probability distribution is given as:

$$p(n_1, n_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)} \times \left(\frac{(n_1-\mu_1)^2}{\sigma_1^2} - 2\rho\frac{(n_1-\mu_1)(n_2-\mu_2)}{\sigma_1\sigma_2} + \frac{(n_2-\mu_2)^2}{\sigma_2^2}\right)\right), \tag{4.39}$$

where μ_1 and σ_1^2 is the mean and variance of n_1 and μ_2 and σ_2^2 is the mean and variance of n_2. The correlation between the two variables is introduced by the *correlation coefficient* which is given as:

$$\rho = \rho_{n_1,n_2} = \frac{C_{n_1,n_2}}{\sigma_1\sigma_2}. \tag{4.40}$$

Here, C_{n_1,n_2} denotes the *covariance* of n_1 and n_2 and is given as:

$$C_{n_1,n_2} = \mathbb{E}\{(n_1-\mu_1)(n_2-\mu_2)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(n_1-\mu_1)(n_2-\mu_2)p(n_1,n_2)dn_1dn_2. \quad (4.41)$$

The joint pdf (4.39) can also be denoted as $\mathcal{N}(\mu_1,\mu_2,\sigma_1^2,\sigma_2^2,\rho)$. Applying this notation the joint pdf of the measurement sample x given the hypotheses is given as:

$$\begin{aligned} x_1,x_2|\mathcal{H}_0 &\sim \mathcal{N}(-1,-1,\sigma_1^2,\sigma_2^2,\rho),\\ x_1,x_2|\mathcal{H}_1 &\sim \mathcal{N}(1,1,\sigma_1^2,\sigma_2^2,\rho). \end{aligned} \quad (4.42)$$

Before we start with the quantizer design we treat the case of unquantized measurement samples to get further insight into the behavior of correlation. Therefore we calculate the LLR for correlated noise samples and it is given as:

$$\begin{aligned} l(x_1,x_2) &= \log\frac{p(x_1,x_2|\mathcal{H}_1)}{p(x_1,\hat{x}_2|\mathcal{H}_0)}\\ &= \log\frac{\exp\left(-\frac{1}{2(1-\rho^2)}\left(\frac{(x_1-1)^2}{\sigma_1^2}-2\rho\frac{(x_1-1)(x_2-1)}{\sigma_1\sigma_2}+\frac{(x_2-1)^2}{\sigma_2^2}\right)\right)}{\exp\left(-\frac{1}{2(1-\rho^2)}\left(\frac{(x_1+1)^2}{\sigma_1^2}-2\rho\frac{(x_1+1)(x_2+1)}{\sigma_1\sigma_2}+\frac{(x_2+1)^2}{\sigma_2^2}\right)\right)}\\ &= x_1\frac{2}{1-\rho^2}\left(\frac{1}{\sigma_1^2}-\frac{\rho}{\sigma_1\sigma_2}\right)+x_2\frac{2}{1-\rho^2}\left(\frac{1}{\sigma_2^2}-\frac{\rho}{\sigma_1\sigma_2}\right). \end{aligned} \quad (4.43)$$

Both measurement samples are linearly transformed. For establishing the test statistic for the NP detector we need to know the conditionally distribution of the LLR given the hypothesis. Therefore we use the fact that a linear transformation of a Gaussian random variable is again Gaussian distributed. Being more precise, the conditional distribution can be find as:

$$l(x_1,x_2|\mathcal{H}_0) \sim \mathcal{N}(\mathbf{a\text{-}1},\mathbf{aC_xa}^{\mathrm{T}}),$$

$$l(x_1, x_2|\mathcal{H}_1) \sim \mathcal{N}(\mathbf{a}\,\mathbf{1}, \mathbf{a}\mathbf{C}_\mathbf{x}\mathbf{a}^\mathrm{T}), \tag{4.44}$$

where $\mathbf{a}$ is a vector consisting of the transformation factors of (4.43) and $\mathbf{C}_\mathbf{x}$ defines the covariance matrix of $\mathbf{x}$. The test statistic for the correlated case is the LLR from (4.43):

$$T_{\mathrm{sd,opt}} = l(x_1, x_2). \tag{4.45}$$

We also can find the decision function, which is equal to the uncorrelated case (4.3):

$$\phi_{\mathrm{sd}} = \begin{cases} T_{\mathrm{sd}} < \gamma_{\mathrm{sd}}, & -1 \\ T_{\mathrm{sd}} = \gamma_{\mathrm{sd}}, & \text{Decide}\,\mathcal{H}_1 \text{with prob}\,\eta_{\mathrm{sd}}\,, \\ T_{\mathrm{sd}} > \gamma_{\mathrm{sd}}, & 1 \end{cases} \tag{4.46}$$

where again γ_{sd} and η_{sd} are given according (4.4) and (4.5). We are ready to evaluate the performance for a distributed hypothesis testing scenario for correlated noise samples and without quantization.

Performance evalution for the unquantized case. As simulation setup we choose the prescribed scenario with two sensors and correlated noise samples. We omit the quantizers and evaluate the performance of the NP detector for different correlation coefficients. The results are visualized in Figure 4.26. The figure shows that the performance improves for lower values of ρ. This can be explained by the fact, that a negative value of ρ entails that the two measurement noise samples have opposite sign with high probability. Hence, this information can be use for noise cancellation and therefore the overall noise reduces or, equivalently, the uncertainty gets reduced. The blue curve indicates the case where $\rho = -0.9$ and performs best and the worst detection probability achieves the case where $\rho = 0.9$ (black curve). Here, the two samples exhibit a positive correlation and therefore the overall information, which can be extracted out of the measurement samples reduces. This entails a degradation in performance.

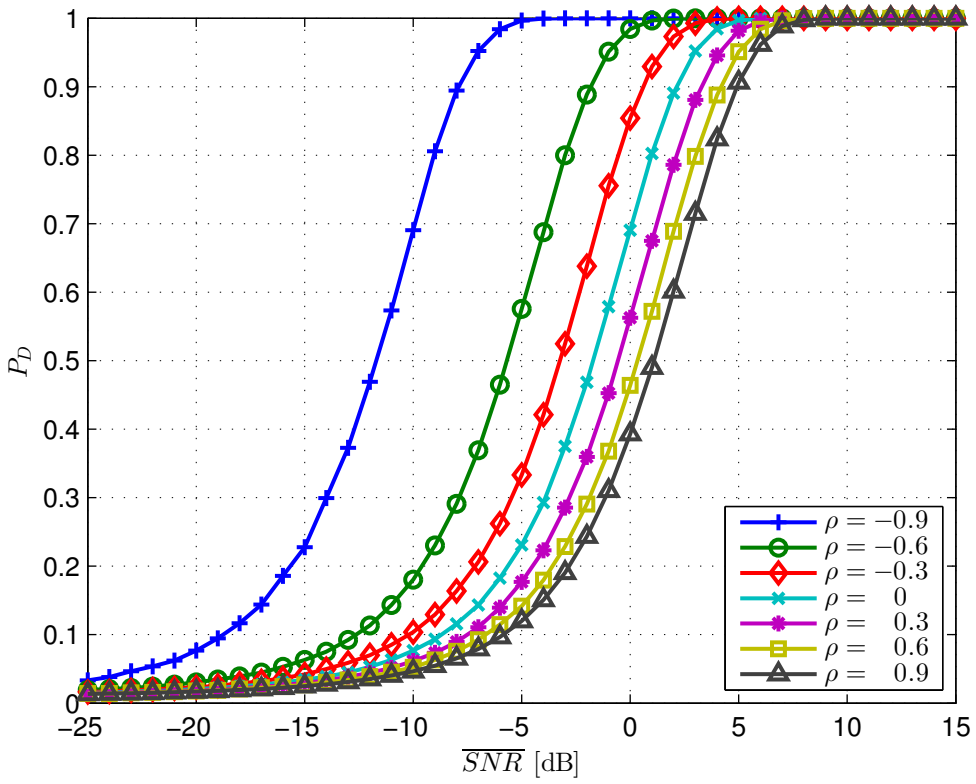


Figure 4.26: Detection probability for different correlation coefficients ρ for the unquantized case.

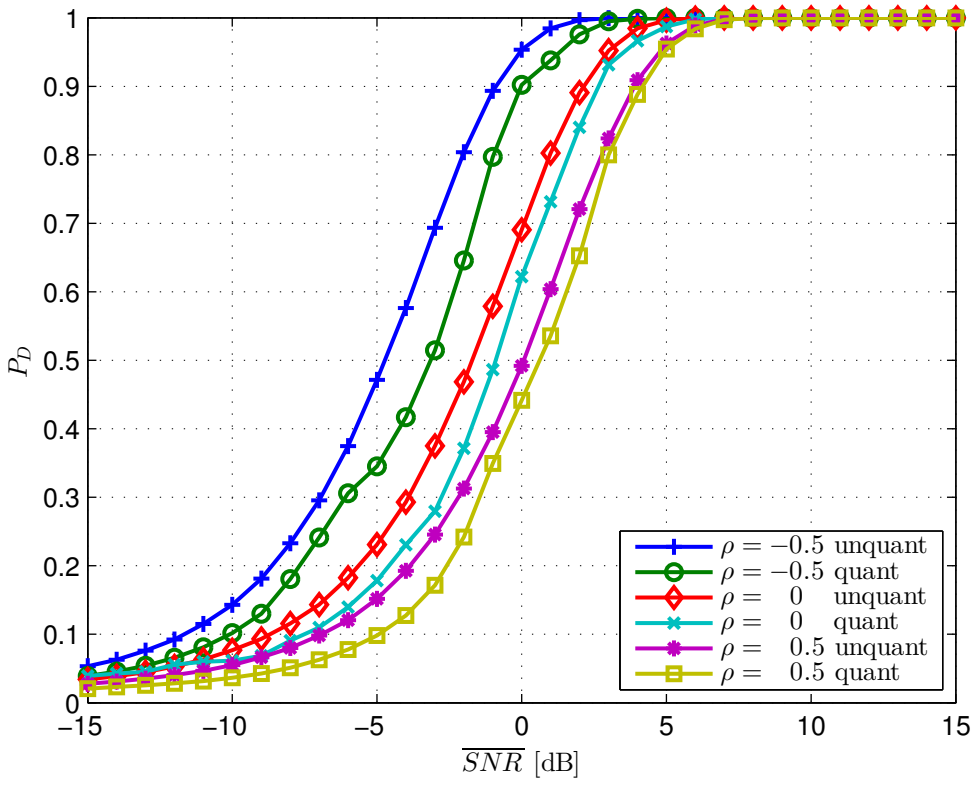


Figure 4.27: Detection probability for different correlation coefficients ρ for quantized and unquantized cases.

Performance evalution for the unquantized case. As last step we want to analyze the scenario for quantized measurement samples. Therefore we choose the same simulation setup as before and design a quantizer based on the MIBM principle where the correlation is taken into account at the quantizer design. We choose a resolution of two bits for each quantizer. Note that in this case the quantizers are designed for the statistics of the Gaussian noise. The actual LLR calculation is done at the FC and the LLR is defined as:

$$l(\hat{x}_1, \hat{x}_2) = \log \frac{p(\hat{x}_1, \hat{x}_2|\mathcal{H}_1)}{p(\hat{x}_1, \hat{x}_2|\mathcal{H}_0)}, \tag{4.47}$$

where the conditional distributions of (4.47) are the conditional distributions of the quantized variables given the hypothesis. The simulation results are presented in Figure 4.27. The figure shows the performance degradation due to the quantization step for different values of ρ. The gap between the quantized and the unquantized case could be decreased by increasing the resolution of the quantizers. On the other hand a higher resolution entails a higher simulation time. Note that the channel between the sensors S_i and the FC are assumed as error-free. A more realistic channel would degrade the performance for both cases.

5

Summary and Outlook

5.1 Summary

In this thesis, we have presented NP detectors and Bayesian detectors for a distributed detection scenario with multiple sensors. First, we assumed the senor noise as statistically independent Gaussian noise with zero-mean and different variance for every sensor. For that case we derived the LLR computation at each sensor and the global test statistic for the NP detector at the FC. We modeled the channel between the sensors and the FC as BSCs. In addition we introduced a method to incorporate the BSC into the overall test statistic, to take into account the detrimental effect of the BSC.

Particular attention we spent on the issue of quantizing the LLR. Therefore, we presented three different quantizer approaches. First, we introduced the Lloyd-Max quantizer, which tries to minimize the MSE of the output and the input of the quantizer. The other two quantizer strategies are based on information theoretic approaches. The MOE quantizer maximizes the entropy of the quantizer output, whereas the IBM quantizer maximizes the mutual information between quantizer output and the binary signal itself.

The last chapter was devoted to the performance evaluation and on topics to improve the overall performance. Indeed, the performance can be improved if we increase the number of sensors, the resolution of the quantizer, the number of measurement

samples, or we introduce a channel-coding scheme to counteract the detrimental effects of the BSC. We also compared the performance of all three quantizer strategies and found out that the IBM performs best for the Gaussian noise assumption. Next we derived the Bayesian detector for the distributed detection scenario and repeated some of the major simulations. Then we spent our attention on the HD case. Therefore we introduced the K-out-of-N rule and evaluated its performance for the NP and Bayesian case. In addition we compared the HD to the IBM quantizer with a resolution of one bit and stated that even in this case it is beneficial to use SD instead of HD. Next we introduced the important topic of bit labeling and presented that the overall performance of the whole detector is dependent on the choice of the labeling. Moreover we introduced a possibility to incorporate the bit labeling and the BSC into the IBM quantizer design to increase the performance. As next step we mentioned the possibility of variable rate allocation. The main idea was to allocate sensor with higher SNR with a higher quantizer resolution and sensors with lower SNR values with a lower quantizer resolution. With that approach the overall performance could be further improved. Then we specified the LLR calculation for the Laplace and Cauchy noise statistics and evaluated the performance. We concluded that for LLR statistics with bounded pdf, which is the case for Laplace and Cauchy noise, the Lloyd-Max quantizer performs better than in the Gaussian case, but still the IBM quantizer achieves the best performance. Next, we generalized our simulation setup to the case of correlated noise samples. Therefore, we extended the IBM principle for a global quantizer design which outputs local quantizers which takes the correlation into account. In addition we extended the test statistic to correlated noise samples and evaluated the performance. We compared the results to the unquantized case and concluded that the performance degrades due to quantization step for the correlated case as well. As last step we stated that the performance can be improved if the noise statistic exhibits a negative correlation coefficient due to noise averaging effects.

5.2 Outlook

In this section we propose several ideas for future research:

- The simulation scenario for correlated noise samples from Section 4.8 could be extended to higher-dimensional random variables. So far we only considered two-dimensional Gaussian noise samples with correlation coefficient ρ. This approach could be generalized to the N-dimensional Gaussian case with covariance matrix $C_{\mathbf{n}}$. However, note that this would increase the simulation runtime.

- In Section 4.8 we assumed the links between sensor S_i and FC as error-free. It would be beneficial to run simulations with more realistic channels. Therefore we could model the channels as BSCs like in the uncorrelated case and generalize the procedure of LLR correction from Section 4.1 to correlated scenarios.

- In Section 4.6, we showed that the performance improves if we apply a appropriate rate allocation, where the best allocation setup was found by an exhaustive search over all possible patterns. For scenarios with a higher number of sensors this simple search algorithm will not be feasible anymore. Therefore the introduction of some systematic allocation strategy like the *reverse water-filling* algorithm [11] would be preferable.

- We already introduced the repetition code as simple channel coding scheme in Section 4.2 to counteract the introduced bit errors by the BSC. This strategy could be extended to more sophisticated schemes to further improve the overall performance.

Bibliography

[1] R. Viswanathan and P. Varshney, "Distributed detection with multiple sensors: Part i — Fundamentals," *Proc. IEEE*, vol. 85, pp. 54–63, Jan. 1997.

[2] J. Lunden, V. Koivunen, A. Huttunen, and H. Poor, "Spectrum sensing in cognitive radios based on multiple cyclic frequencies," in *Proc. 2nd Int. Conf. Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, pp. 37–43, Aug. 2007.

[3] S. M. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*. Upper Saddle River (NJ): Prentice Hall, 1998.

[4] S. Chaudhari, J. Lunden, V. Koivunen, and H. Poor, "Cooperative sensing with imperfect reporting channels: Hard decisions or soft decisions?," *IEEE Trans. Signal Process.*, vol. 60, pp. 18–28, Jan. 2012.

[5] J. N. Tsitsiklis, "Decentralized detection," *Advances in Statistical Signal Processing*, vol. 2, no. 1, pp. 297–344, 1993.

[6] P. K. Varshney, *Distributed Detection and Data Fusion*. New York: Springer, 1997.

[7] R. Viswanathan and V. Aalo, "On counting rules in distributed detection," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 772–775, May 1989.

[8] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, pp. 129–137, March 1982.

[9] J. Max, "Quantization for minimum distortion," *IEEE Trans. Inf. Theory*, vol. 6, no. 2, pp. 7–12, 1960.

[10] D. Messerschmitt, "Quantizing for maximum output entropy (corresp.)," *IEEE Trans. Inf. Theory*, vol. 17, pp. 612–612, Sept. 1971.

[11] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York (NY): Wiley, 2nd ed., Sept. 2006.

[12] R. Wood, "On optimum quantization," *IEEE Trans. Inf. Theory*, vol. 15, pp. 248–252, March 1969.

[13] N. Tishby, F. Pereira, and W. Bialek, "The information bottleneck method," in *Proc. 37th Annu. Allerton Conf. Commun., Control, Comput.*, pp. 368–377, Sept. 1999.

[14] R. E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Inf. Theory*, vol. 18, pp. 460–473, 1972.

[15] G. Zeitler, R. Kötter, G. Bauch, and J. Widmer, "Design of network coding functions in multihop relay networks," in *Proc. 5th Int. Symp. Turbo Codes and Related Topics*, pp. 249–254, Sept. 2008.

[16] S. Lin and D. J. Costello, *Error Control Coding*. Upper Saddle River (NJ): Prentice Hall, 2nd ed., 2004.

[17] L. L. Scharf, *Statistical Signal Processing*. Reading (MA): Addison Wesley, 1991.

[18] E. N. Gilbert, "Gray codes and paths on the n-cube," *The Bell System Technical Journal*, vol. 37, pp. 815–826, May 1958.

[19] A. Winkelbauer, G. Matz, and A. Burg, "Channel-optimized vector quantization with mutual information as fidelity criterion," in *Proc. 47th Asilomar Conf. Signals, Systems, Computers*, pp. 851–855, Nov. 2013.

[20] S. Kotz, T. J. Kozubowski, and K. Podgorski, *The Laplace distribution and generalizations - A revisit with applications to Communications, Economics, Engineering, and Finance*. Birkhäuser, 2001.

[21] N. L. Johnson, S. Kotz, and N. Balakrishnan, *Continuous Univariate Distributions*. Wiley Interscience, 1994.

[22] N. Slonim, N. Friedman, and N. Tishby, "Multivariate information bottleneck," *Neural Computation*, vol. 18, pp. 1739–1789, Aug. 2006.

Notation

Throughout this thesis, vectors are denoted by boldface letters and random variables are denoted by uppercase letters. Unless noted otherwise, the meaning of the following symbols is as stated below.

Symbol	*Meaning*
$\mathcal{B}$	Binomial distribution, $\mathcal{B}(k,n,p) = \sum_{i=0}^{k} \binom{n}{i} p^i (1-p)^{n-i}$
$\mathbf{c}$	Bit sequence
d	Number of bits
$d_{j,k}$	Hamming distance between bit sequences $\mathbf{c}_j$ and $\mathbf{c}_k$
D	Number of different quantizer outputs
D_{KL}	Kullback-Leibler divergence
D_Q	Distortion measure for quantizer design
C_{ij}	Cost assigned for $\mathcal{H}_i$ although $\mathcal{H}_j$ is in force
d^2	Deflection coefficient
$\mathbb{E}\{X\}$	Expectation of the random variable X
f	Weight of detection probability for local NP detector
g	Decision boundary
$H(X)$	Entropy of the random variable X
h	Weight of false alarm probability for local NP detector

$\mathcal{H}_i$	Hypothesis, $\mathcal{H}_i : \theta \in \Theta_i$
$I(X;Y)$	Mutual information of the random variables X and Y
$L(\cdot)$	Likelihood ratio, $\frac{p(\cdot\|\mathcal{H}_1)}{p(\cdot\|\mathcal{H}_0)}$
$l(\cdot)$	Log-likelihood ratio, $l(\cdot) = \log L(\cdot)$
$\hat{l}(\cdot)$	Quantized Log-likelihood ratio
$\mathcal{N}(\mu, \sigma^2)$	Gaussian distribution with mean μ and variance σ^2
n	Gaussian noise sample
$P\{\mathcal{E}\}$	Probability of the event $\mathcal{E}$
P_A	Probability for a correct accepting, $P_\mathrm{A} = P\{x \in \mathcal{X}_0\|\mathcal{H}_0\}$
P_b	Bit inversion probability of BSC
P_D	Probability for a correct detection, $P_\mathrm{D} = P\{x \in \mathcal{X}_1\|\mathcal{H}_1\}$
P_FA	Probability for a false alarm, $P_\mathrm{FA} = P\{x \in \mathcal{X}_1\|\mathcal{H}_0\}$
P_M	Probability for a miss $P_\mathrm{M} = P\{x \in \mathcal{X}_0\|\mathcal{H}_1\}$
$R(\cdot)$	Posterior probability ratio, $R(\cdot) = \frac{P\{\mathcal{H}_1\|\cdot\}}{P\{\mathcal{H}_0\|\cdot\}}$
R_B	Baysian risk
Q	Quantizer
R	Rate in rate-distortion theory
S	Sensor
t	error-correction capability
T	Test statistic
$\mathcal{U}$	Set of all possible outcomes of the sensor, $\mathcal{U} = \{1, \ldots, D\}$
u	Compressed information at the output of the sensor
$\mathrm{var}(\cdot)$	Variance
$\mathcal{X}$	Observation space
x	Observation, $x \in \mathcal{X}$
$\hat{\mathcal{X}}$	Quantized observation space
$\hat{x}$	Quantized observation
y	Relevant information in IBM
z	Output of BSC

α	Size of LRT
β	Lagrange multiplier
γ	Threshold of LRT
γ_{FC}	Fusion rule
δ	Delta function
η	Probability for deciding for $\mathcal{H}_1$, when the test statistic is equal to the γ
$\phi(\cdot)$	Decision function
$\psi()$	Normalization function in IBM
Θ	Parameter space of hypotheses
θ_i	Element of parameter space, $\theta_i \in \Theta$

List of Abbreviations

BA	**B**lahut **A**rimoto
BER	**b**it **e**rror **r**ate
BFSK	**b**inary **p**hase **s**hift **k**eying
BSC	**b**inary **s**ymmetric **c**hannel
cdf	**c**umulative **d**istribution **f**unction
DSN	**d**istributed **s**ensor **n**etwork
FC	**f**usion **c**enter
HD	**h**ard **d**ecision
IBM	**i**nformation **b**ottleneck **m**ethod
LLR	**l**og-**l**ikelihood **r**atio
LRT	**l**ikelihood **r**atio **t**est
M-FSK	**M**-ary **f**requency **s**hift **k**eying
MIBM	**m**ultivariate **i**nformation **b**ottleneck **m**ethod
MOE	**m**aximum **o**utput **e**ntropy
MSE	**m**ean-**s**quare **e**rror
NBC	**n**atural **b**inary **c**oding
NP	**N**eamon-**P**earson
pdf	**p**robability **d**ensity **f**unction
pmf	**p**robability **m**ass **f**unction
ROC	**r**eceiver **o**perating **c**haracteristic
SD	**s**oft **d**ecision

SNR	**s**ignal-to-**n**oise **r**atio