Universitat
Autònoma
de Barcelona

etse

COOPERATIVE STRATEGIES FOR IMPERFECT CSI SCENARIOS BASED ON DISTRIBUTED ALAMOUTI

Memòria del Projecte Fi de Carrera
d' Enginyeria Tècnica de
Telecomunicació. Especialitat de
Sistemes Electrònics.
realitzat per
Marc Barceló Lladó
i dirigit per
José López Vicario
Bellaterra, 18 de Septembre de 2008

Abstract

Cooperative communications are gaining much interest in modern communications because they allow to improve the information transmission between a source and a destination using various intermediate terminals. This project is a complete study of cooperative systems, analyzing its performance and comparing the use of a single terminal with the use of the Alamouti code, which uses two terminals. First, there is an introduction to cooperative systems and to information theory. Then we have studied a cooperative system using the information theory, in terms of outage probability, and subsequently we have adapted it to a real cooperative system using a QPSK modulation, studying its packet error probability. Finally several protocols are proposed to improve the performance of the studied cooperative system.

Contents

Acronyms

ARQ	Automatic Repeat Request
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
CSI	Channel State Information
FDD	Frequency Division Duplexing
GPS	Global Positioning System
MIMO	Multiple-Input Multiple-Output
ML	Maximum Likelihood
MRC	Maximum Ratio Combining
PER	Packet Error Rate
SER	Symbol Error Rate
SISO	Single-Input Single-Output
SNR	Signal-to-Noise Ratio
STBC	Space-Time Block Code
TDD	Time Division Duplexing

Chapter 1

Introduction

1.1 Motivation

The wireless communications have to satisfy the increasing demand for high bandwidth and high data speed services. In order to continue the improvement of these two factors, while maintaining the reliability of communications, it is necessary to incorporate enhancements to the traditional communications scheme.

A possible option is to use multi-antenna systems like MIMO (Multiple-Input Multiple-Output), which provides capacity improvements compared to single-antenna configuration by adding multiple antennas to the source, to the destination, or both. On the other hand, the terminals are increasingly smaller and lighter, so the incorporation of multiple antennas in a single device can be non practical.

To avoid an increase of the weight and the size of the terminals, cooperative system techniques can be used. They allow to emulate the behavior of a MIMO system using only one antenna for each terminal. This is achieved using various terminals located between the transmitter and the receiver working as a virtual antenna array.

With this new communications scheme, important improvements in the system performance are obtained, allowing to increase the bandwidth and the data rate as the new communications systems require.

1.2 Objectives

We have two important objectives to develop along this work:

- The first main objective of this work is to study a cooperative system from an information theory point of view. We want to compare the system performance in terms of outage probability when a best relay selection method is applied and when the Alamouti code is used. Also, we want to study the importance of the feedback errors.

- The second main objective is to implement a realistic cooperative system using the results obtained previously. Then we want to analyze the system performance in terms of *Packet Error Rate* (PER). Using this results we want to propose several protocols aimed at optimizing the choice of the transmission option.

1.3 Organization of the Document

In this work we study the cooperative communications, commenting the main advantages over the traditional communications system, comparing its different alternatives, and proposing various useful protocols for a real cooperative system.

Chapter 2 is a theoretical introduction of cooperative communications, the principal relaying algorithms and the Alamouti scheme. Here we review the state of the art on cooperative communications.

In *chapter 3* we elaborate a reference cooperative model defining the channel model, the power allocation and the duplex mode used along the work. In this chapter we want to establish the base of the cooperative communications system. It is analyzed in *chapter 5* and in *chapter 6*.

Chapter 4 is a summary of the principal theoretical concepts of the information theory, as the channel capacity, the information channel and the outage capacity.

In *chapter 5*, the first of the two main objectives of this work is developed. We present two cooperative transmission methods, the Best Relay option and the Alamouti option. To evaluate the system performance we study the outage probability of the cooperative model using the information theory concepts learned in *chapter 4*. First, different geometries of the relay position are studied, and then, we situate the relays randomly. In this chapter we introduce the consequences of having a certain feedback error probability.

In *chapter 6* we develop the second main objective. We start introducing the real cooperative model. Then we introduce three protocols using different criteria as the feedback error probability, the number of available relays and the average SNR. Then we change the power allocation explained in *chapter 3* to propose a new protocol. Finally we change the channel model introducing a correlation parameter, to improve the previous protocols, by using a new relay selection strategy.

In *chapter 7* we conclude this work commenting the results of the cooperative system performance obtained in *chapters 5* and *6*, and analyzing the improvements on the system performance due to the proposed protocols.

Chapter 2

Introduction to Cooperative Communications

2.1 Introduction

In wireless communications systems, the physics of electromagnetic waves lead to multipath propagation of wireless signals and, in turn, variations in received signal strength. Combined with transceiver motion these effects produce wireless channel variations, generally called fading, in space, frequency, and time. Diversity techniques for mitigating multipath fading are important to improve the performance of wireless communications systems and networks.

The main motivation of cooperative communications systems is to improve the reliability of communications in terms of outage probability and/or *Packet Error Rate* (PER) for a given transmission rate.

It has been physical layer researchers who have championed the use of cooperative diversity in wireless networks. They argue that nodes equipped with a single antenna, through physical layer coding and signal processing, could achieve similar diversity and coding gains to the co-located multi-antenna systems, while leveraging the distributed hardware and battery resources that are already available. It is explained in [Tar99].

The advantages of *Multiple-Input Multiple-Output* (MIMO) systems have been widely demonstrated, to extend that certain transmit diversity methods have been incorporated into wireless standards. Although transmit diversity is clearly advantageous on a cellular base station, it may not be practical for other scenarios. Specifically, due to size, cost, or hardware limitations, a wireless agent may not be able to support multiple antennas. Examples include the size in most handsets or both size and power in nodes in a wireless sensor network.

The cooperative communications allow single-antenna mobiles to get some benefits of MIMO systems. The basic idea is that single-antenna mobiles in a multi-user scenario can share their antennas in a manner that creates a virtual MIMO system.

Transmitting independent copies of the signal generates diversity which can effectively combat deleterious effects of fading. In particular, spatial diversity is generated by transmitting signals from different locations, thus allowing independently versions of the signal at the receiver. Cooperative communications generates this diversity in a new way as it is commented in *section 2.3*.

2.2 Spatial Diversity

Increasing the quality or reducing the effective error rate in a multipath fading channel is extremely difficult. In *Additive White Gaussian Noise* (AWGN), using typical modulation and coding schemes, reducing the effective *Bit Error Rate* (BER) from 10^{-2} to 10^{-3} may require only 1 or 2 dB higher *Signal-to-Noise Ratio* (SNR). Achieving the same in a multipath fading environment, however may require up to 10 dB improvement in SNR [Ala98].

It is crucial to effectively combat or reduce the effect at both the remote units and the base stations, without additional power or any sacrifice in bandwidth.

Theoretically, the most effective technique to mitigate multipath fading in a wireless channel is transmitter power control. If channel conditions experienced by the receiver on one side of the link are known at the transmitter on the other side, the transmitter can adjust the signal to overcome the effect of the channel at the receiver. There are some problems with this approach. The major problem is the required dynamic range. For the transmitter to overcome a certain level of fading, it must increase its power by the same level, which in most cases is not practical because of radiation power limitations and the size and cost of the amplifiers.

In most scattering environments, the spatial diversity, is a practical, effective and, hence, a widely applied technique for reducing the effect of multipath fading. It is used to transmit the signal via several independent diversity branches to get independent signal replicas.

2.3 Cooperative Communications

A Cooperative communications system uses several terminals situated between the source (terminal which wants to send the information) and the destination (terminal which has to receive the information) to improve their wireless communication.

The classical relay channel model is composed of three terminals: a source that transmits information, a destination that receives information and a relay that both receives and transmits information. Some level of synchronization between the three terminals is required.

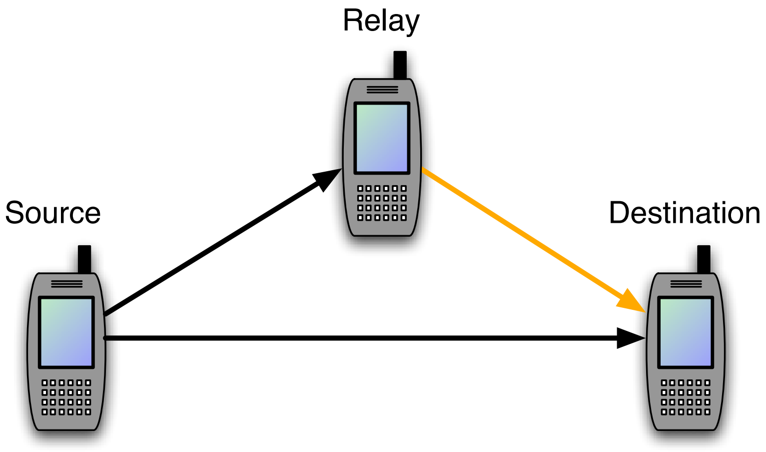

Figure 2.1. Three nodes model

The transmission occurs in two time slots. In *figure 2.1* the first time slot is represented by black arrows and the second time slot is represented by an orange arrow.

In cooperative wireless communication, the wireless agents (users), may increase their effective quality of service (measured at the physical layer by bit error rates, block error rates, or outage probability) via cooperation.

Cooperative leads to interesting tradeoffs in code rates and transmit power. In the case of power, one may argue on one hand that more power is needed because each user is transmitting for both users. On the other hand, the baseline transmit power for both users will be reduced because of diversity. Then, a reduction of transmit power can be obtained.

Similar questions arise for the rate of the system. In cooperative communications each user transmits both his own bits to another user who has to receive the information. One might think this causes loss of rate in the system. However, the spectral efficiency of each user improves because, due to cooperation diversity the channel code rates can be increased.

Among other potential benefits, cooperative communications leverages the spatial diversity available when multiple transmissions experience independent fading and/or shadowing. For example, if the source signal experiences a deep fade at the destination, there remains a significant chance that it can be effectively communicated to the destination via one of the relays.

Since cooperative communications is a network problem, protocol layering issues and cross-layer architectures naturally arise. At the physical layer, encoding signal processing algorithms are required at the terminals. Synchronization of signals in time and frequency must be addressed by protocols in the link layer and medium-access in coordination with the physical layer. Finally collecting sets of radios into cooperative groups is inherently a cross-layer issue that can involve the physical, medium-access control, link, and even network layers.

As we will further elaborate, the right combination of architecture (what logical components are identified and how they can interact), and algorithms (specific signal encoding, processing, and decoding techniques) can depend upon the application context, hardware available, and complexity of the system.

2.4 State of the Art on Cooperative Communications

The basic ideas behind cooperative communications can be traced back to the work of Cover and El Gamal on the information theory properties of the relay channel [Cov79]. This work analyzed the capacity of the three-node network consisting of a source, a destination, and a relay. It was assumed that all nodes operate in the same band, so the system can be decomposed into a broadcast channel from the view point of the source and a multiple access channel from the viewpoint of the destination.

Various extensions to the case of multiple relays have appeared in the work of Schein and Gallager [Sch00] and [Sch01], Gupta and Kumar [Gup01], Gastpar [Gas02] and Reznik [Rez02].

For channels with multiple information sources, Krame and Wijngaarden [Kra00] consider a mutiple-acces channel in which the multiple sources communicate to a single destination and a share single relay.

Work by King [Kin78], Carleial [Car82] examines multiple-access channels with generalized feedback. Here the generalized feedback allows the sources to essentially act as relays for one another. The construction in [Car82] can be viewed as two-terminal generalizations of the cooperation scheme in [Cov79].

Sendonaris introduces multipath fading into the model of [Car82], calling their approaches for this system model “user cooperation diversity” [Sen98].

Some recently works as [Jin06] and [Scu05], have described the performance of a cooperative system when a *Space-Time Block Code* (STBC) is used. STBC are explained in *section 2.6*.

A OFDM-based cooperative communications study has been done in some works as [Lin05].

2.5 Relaying Algorithms

In cooperative communications there are several techniques to establish a cooperation between the source and the destination. Since a growing number of relaying algorithms are appearing in the literature, we summarize the most important of these.

2.5.1 Amplify-and-Forward.

In this algorithm, relays simply amplify what they receive subject to their power constraint. Amplifying corresponds to a linear transformation at the relay.

Consider the case of a single relay. This simple algorithm divides transmissions into two blocks of equal duration, one block for the source transmission and one block for the relay transmission. The source transmits $X_s[k]$ for $k=1,2,...,n$. The relay processes its corresponding received signal $Y_r[k]$ for $k=1,2,...,n$, and relays the information by transmitting:

$$X_r[k] = \beta_r Y_r[k-n]$$

for $k=n+1, n+2,...2n$, where β_r is the relaying gain.

To remain within its power constraint, an amplifying relay must use gain:

$$\beta_r \leq \sqrt{\frac{P_s}{\left|A_{r,s}\right|^2 P_r + N_0}}$$

where the gain is allowed to depend upon the fading coefficient A_{rs} between the source s and the relay r. P_s is transmission power of the source and P_r is the transmission power of the relay. The destination processes its received signal $Y_d[k]$ for $k=1,2,...,2n$ by some form of diversity combining of the two sub-blocks of length n.

When multiple relays are active, they can each relay in their own block of channel uses so that their transmissions do not interfere at the destination, or they can relay simultaneously so that their transmissions interfere at the destination. The former approach offers better diversity benefits, but decrease bandwidth efficiency.

This method was proposed and analyzed by Laneman in [Lan01]. It has been shown that for the two-user case, this method achieves diversity order of two, which is the best possible outcome at high SNR.

2.5.2 Decode-and-Forward.

For decode-and-forward, relays apply some form of detection and/or decoding algorithms to their received signals and re-encode the transformation into their transmit signals, This process often corresponds to a nonlinear transformation of the received signals.

Again we consider the simplest algorithm, the case of a single relay. This algorithm divides transmissions into two blocks of equal duration, one block for the source transmission and one block for the relay transmission. The source transmits $X_s[k]$ for $k=1,2,...,n$. The relay forms an estimate $X'_s[k]$ by decoding its corresponding received signal $Y_s[k]$ for $k=1,2,...,n$, and relays an re-encoded version of $X'_s[k]$. For example the relay can implement repetition coding by transmitting the signal

$$X_r[k] = \sqrt{\frac{P_r}{P_s}} X'_s[k-n]$$

for $k=n+1,n+2,...,2n$.

P_s is transmission power of the source and P_r is the transmission power of the relay. The destination processes its received signal $Y_d[k]$ for $k=1,2,...,2n$ by some form of diversity combining of the two sub-blocks. The relay can encode the source message using a codeword that is generally correlated, by not necessarily identical to, the source codeword. The decode and forward communication is represented in *figure 2.2*.

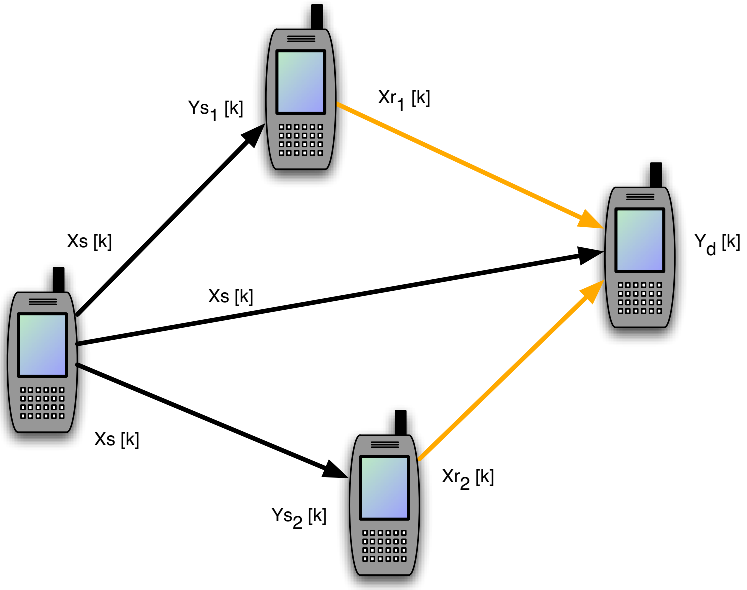

Figure 2.2. Decode and Forward

When multiple relays are involved, they can all employ repetition coding or a more general space-time code to transmit information jointly with the source to the destination.

A further improvement of decode and forward is dynamic decode and forward. In this protocol, the relay starts by receiving from the source and does not begin transmitting until it is sure it has correctly received the source transmission.

Figure 2.3 compares the decode and forward algorithm and the amplify and forward algorithm, showing the transmitted and the received signals by the terminals.

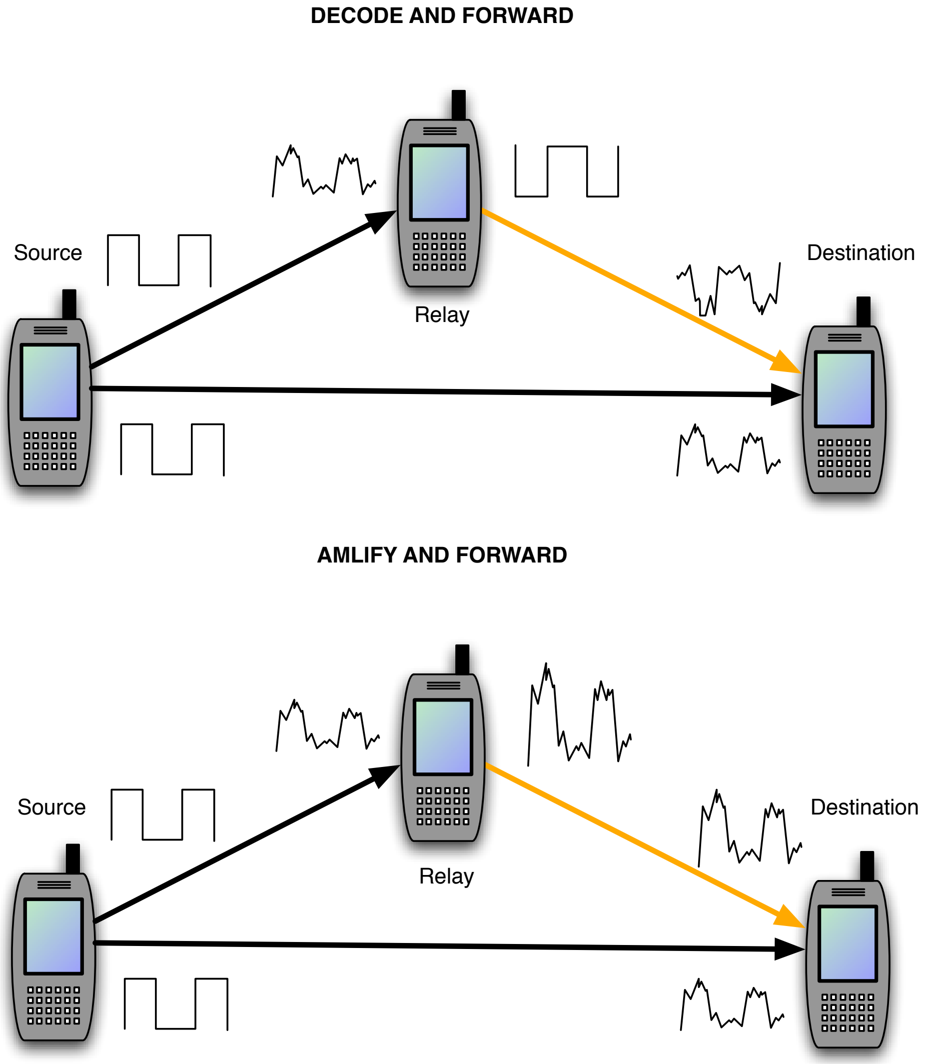

Figure 2.3. Comparison between Decode and Forward and Amplify and Forward

2.5.3 Selection and Dynamic Relaying

Fixed decode and forward is limited by direct transmission between the source and relay. However, since the fading coefficients are usually known to the receivers, $A_{r,s}$ (fading coefficient of the relay-to-source link) can be measured to high accuracy by the cooperating terminals and they can adapt their transmission format according to the realized value of $A_{r,s}$.

We can combine the two algorithms commented before to achieve diversity gain. With this algorithm, if the measured $|A_{r,s}|^2$ falls below a certain threshold, the source simply continues its transmission directly to the destination, but if the measured $|A_{r,s}|^2$ lies above the threshold, the relay forwards what it received from the source, using amplify-and-forward or decode-and-forward.

This algorithm offers diversity because the information is lost when two of the fading coefficients are small. Specifically, if $|A_{r,s}|^2$ is small, then $|A_{d,s}|^2$ must also be small to lost the information when the relay employs amplify-and-forward or decode-and-forward.

2.5.4 Incremental Relaying

Fixed and selection relaying can make inefficient use of the degrees of freedom of the channel, especially for high rates, because the relays repeat all the time. In an incremental relaying protocol a single bit indicates the success or failure of the direct transmission exploiting limited feedback from the destination terminal.

These incremental relaying protocols can be viewed as extensions of incremental redundancy, or hybrid *Automatic-Repeat-Request* (ARQ), to the relay context. In ARQ the source retransmits if the destination provides a negative acknowledgment via A; in incremental relaying, the relay retransmits in an attempt to exploit spatial diversity.

2.6 Space Time Block Coding

Cooperative communications use several techniques to improve their system reliability. One of the most used technique is the *Space Time Block Coding* (STBC). It is used to transmit multiple copies of a data stream across a number of antennas and to exploit the various received versions of the data to improve the reliability of data-transfer. STBC combines all the copies of the received signal in an optimal way to extract as much information from each as possible.

In a STBC, the data stream to be transmitted is encoded in blocks, which are distributed among spaced antennas and across time. The process of receiving diverse copies of the data is known as diversity reception.

An STBC is usually represented by a matrix as in *figure 2.4*. Each row represents a time slot and each column represents one transmit antenna.

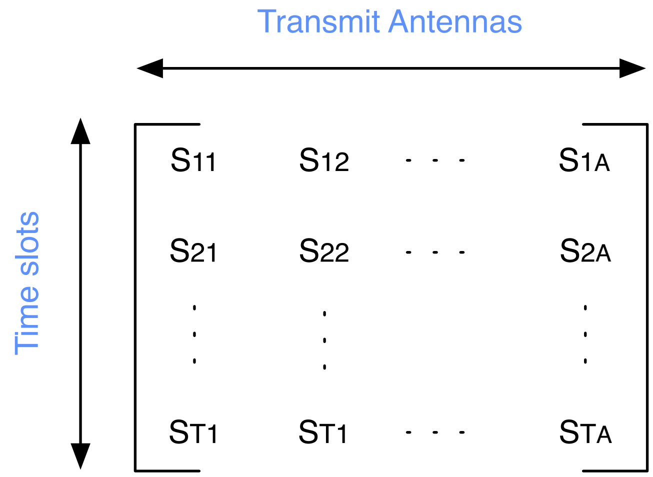

Figure 2.4. STBC transmission matrix

Where *A* is the number of transmit antennas and *T* is the number of time slots. Here, $S_{i,j}$ is the modulated symbol to be transmitted in time slot *i* from antenna *j*.

An interesting STBC is the Alamouti code. It is a simple transmit diversity scheme which improves the signal quality at the receiver on one side of the link by simple processing across two transmit antennas on the opposite side. The obtained diversity order is equal to applying *Maximum Ratio Combining* (MRC) with two antennas at the receiver.

The important thing about this scheme lies in that it is not necessary a bandwidth expansion, as redundancy is applied in space across multiple antennas, not in time or frequency.

2.6.1 Alamouti Scheme Description

Consider a source creating a complex symbol *s(n)* every time slot. Each pair of consecutive symbols s_1 and s_2, are processed in block by an Alamouti encoder. It transmits the first symbol by the first antenna and the second symbol by the second antenna. Subsequently the first antenna transmits s_2^* (where * denotes simple conjugated) symbol and the second antenna transmit $-s_1^*$.

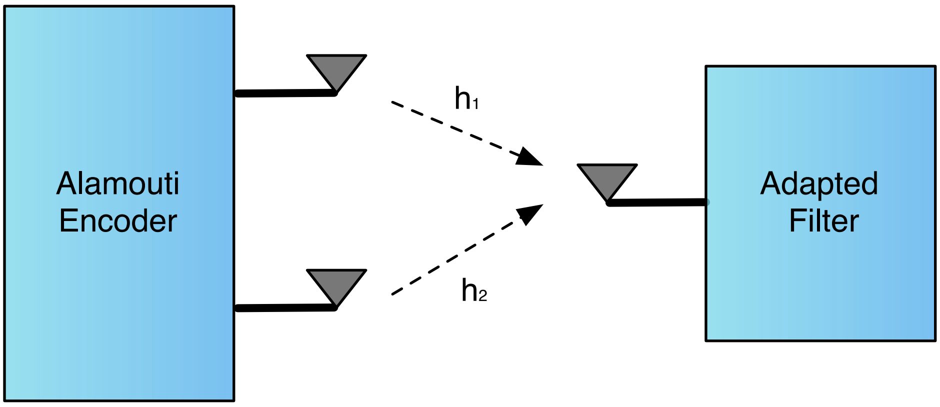

Figure 2.5. Alamouti system

The parameters h_1 *and* h_2 used in *figure 2.5* are the channel response between the transmission antenna and the receiver.

It is possible to represent the symbol block that describes the Alamouti code in a matrix as

$$\mathbf{S} = \begin{bmatrix} s_1 & s_2 \\ s_2^{\ *} & -s_1^{\ *} \end{bmatrix}$$

It is important to observe that the source is sending two symbols in two time slots. It is the same speed of a *Single-Input Single-Output* (SISO) system, where only the direct path is considered.

The receiver observes two signals coming from the source. Considering a Rayleigh channel, it is possible to write the received signals as

$$\mathbf{y} = \begin{bmatrix} r_1 \\ r_2^{*} \end{bmatrix} = \begin{bmatrix} h_1 & h_2 \\ -h_2^{*} & h_1^{*} \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2^{*} \end{bmatrix} = \mathbf{H_{eq}}s + \mathbf{v}$$

where n_1 and n_2 are White Gaussian Noise of the two received signals respectively.

From the above expression one can extract the main particularity of Alamouti: The resulting equivalent channel matrix H_{eq} is orthogonal. Then transmitted symbols can be easily decoupled at the receiver by using simple match filtering as

$$\widehat{s} = \mathbf{H_{eq}}^{H}\mathbf{y} = h^2 + \mathbf{H_{eq}}^{H}\mathbf{v}$$

Due to the orthogonality of H_{eq}, the components of the noise vector are independent and identically distributed and, hence independent optimum *Maximum Likelihood* (ML) detection can be carried out for each transmitted symbol.

The received SNR is the same for the two transmitted symbols and can be written as:

$$\gamma = \frac{\overline{\gamma}}{2}\left(|h_1|^2 + |h_2|^2\right)$$

where $\overline{\gamma}$ is the total transmitted power at the transmitter. One can observe that the diversity order is two if the channel fades are uncorrelated.

Chapter 3

Reference Model

The work has two well differentiated parts. The first one is the study of a cooperative system from an information theory point of view, and the second one deals with a more realistic model than the previous one. That is why two different models, although closely related to each other, will be used along this work. This section explains the basis of these two models.

Once we have introduced the cooperative systems, we are able to model a complete cooperative system. This is our first step to create a real cooperative system.

Our cooperative system uses the decode and forward algorithm, because its analysis is less complex than the amplify and forward algorithm, and it provides a similar performance.

In this work we deal with a system composed of a source, a destination and N relays. The number of relays in a real scenario is not constant and different values have been analyzed.

The communication shall be done using two time slots. In the first time slot, the source transmits the information in broadcast mode. In the second time slot, the relays which have received the information from the source, transmit this information to the destination. It is shown in *figure 3.1*.

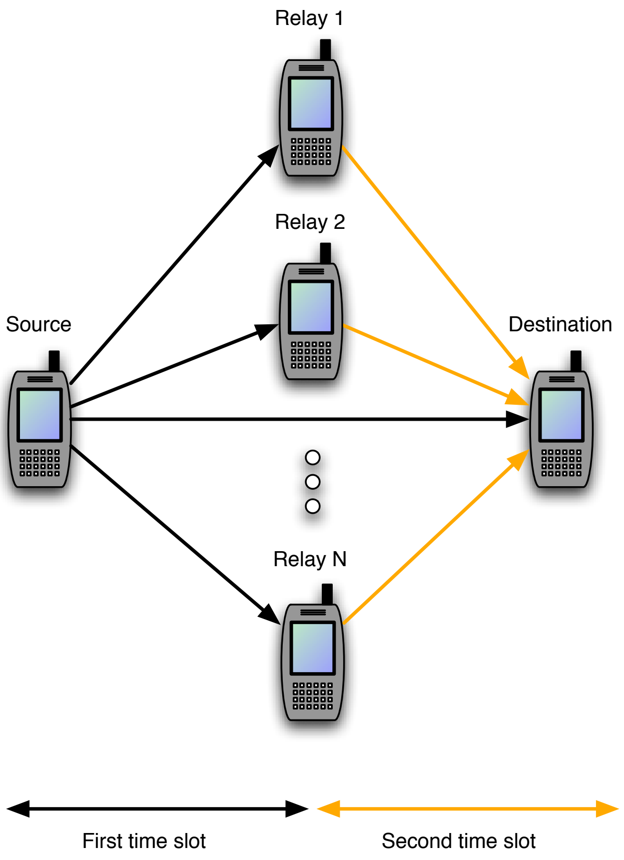

Figure 3.1. General Model

It is not practical to use all the available relays to transmit the information to the destination, because some of these relays could be unusable. It is better to allocate the available power among the best relays. The selection of these relays will be studied along this work, proposing different strategies and methods.

3.1 Duplex Mode

One cost of employing relays in practical systems is that current radios cannot transmit and receive simultaneously in the same frequency band, i.e. they must operate in half-duplex mode.

There are two important duplex modes that can be used in a cooperative scenario. The first one is TDD (Time Division Duplexing), and the other one is FDD (Frequency Division Duplexing).

TDD allows to separate the relay and the destination signals using the same frequency as it is shown in *figure 3.2*. Using TDD, it can be assumed that relay-to-destination link is equal to destination-to-relay link.

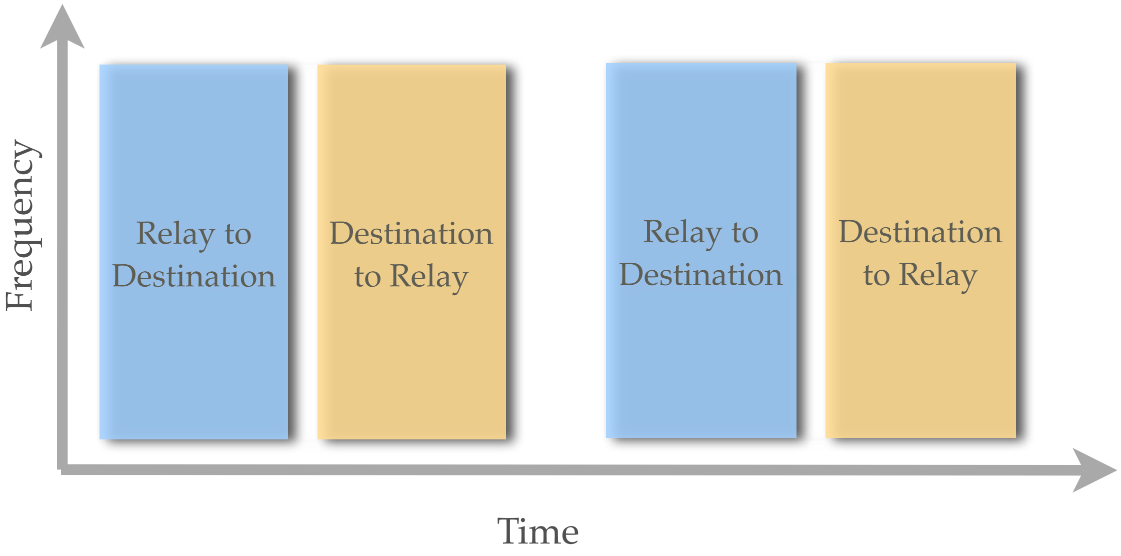

Figure 3.2. Time Division Duplexing

Using FDD, the relay and the destination can transmit at the same time at different frequencies. It is shown in *figure 3.3*. Using this duplex mode, the terminals must have two antennas or a duplexer. In this case the two links are considered equals only if:

$$|f_{rd} - f_{dr}| > B_c$$

where f_{rd} is the frequency used in the relay-to-destination link, f_{dr} is the frequency used in the destination-to-relay link, and B_c is the coherence bandwidth.

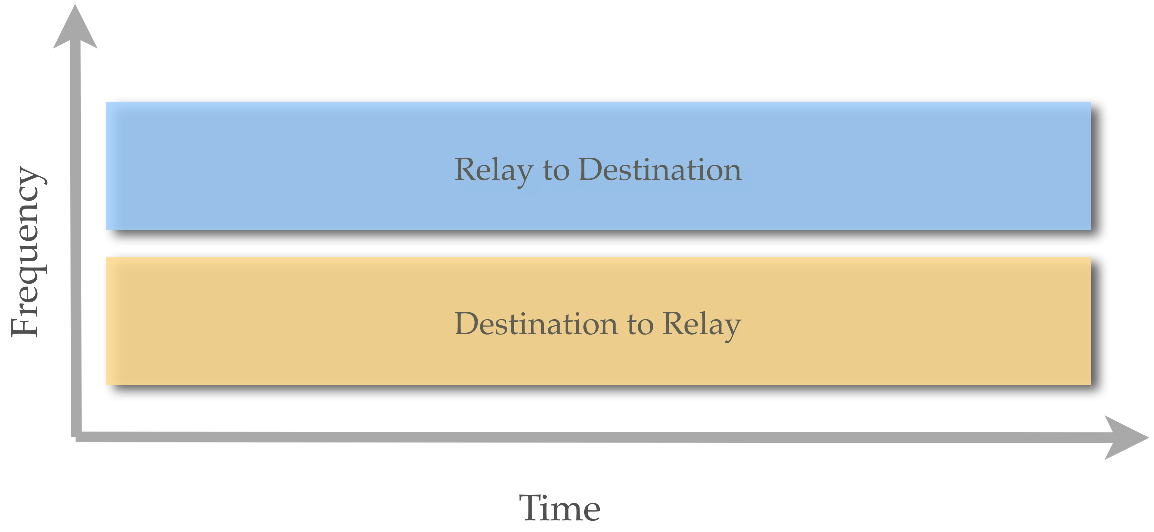

Figure 3.3. Frequency Division Duplexing

In this work a TDD duplex mode is used. We have chosen it because, as we have commented before, it needs to use terminals with two antennas or a built-in duplexer, and it is not practical in scenarios where the terminals should be small and light.

3.2 Channel Model

One of the most important parts of a communications model is channel modeling. In this work we define an arbitrary link *A-B* between two nodes *A* and *B*. Node *A* can be the source or a relay, while node *B* can correspond to a relay or the destination. With this model in mind, the received signal in the link *A-B* can be written as

$$r_B = h_{A,B} \cdot x_A + n_B$$

where x_A is the transmitted symbol with power P_A, n_B is AWGN noise with zero mean and variance σ_n^2, $h_{A,B}$ is the channel response between nodes *A* and *B,* modeled as $h_{A,B} \sim CN(0,1)$ (Rayleigh fading). We assume a block fading channel where the channel response remains constant during one time-slot and that different channels (for changing *A* or *B*) are independent and identically distributed.

The channel estimation is performed using a pilot signal from the relay to the destination, before the information transmission.

3.3 Power Allocation

In a cooperative scenario there are multiple relays, so power allocation is a key aspect of cooperative systems because it decides how many relays must transmit, which relays must transmit and how it distributes the available power. Power allocation has been decided so that it can be a fair comparison between a cooperative transmission and a non cooperative transmission. The source transmits with power P_s and it is divided into equal shares among the source-to-relay links and the relay-to-destination links.

The transmission from the source to the relays uses Ps/2 because it is transmitting in broadcast to all the relays, in contrast, the transmission power from the relays to the destination must be shared among the used relays. At first, this power will be uniformly divided among the relays.

Chapter 4

An Overview of Information Theory

4.1 Introduction

Although problems of relaying and cooperation have been examined in the information theory community in the last years, the fundamental performance limits, in terms of the Shannon capacity are not known in general. Nevertheless, useful bounds on capacity have been obtained for various approaches. When it is applied to wireless channel models in particular, relaying and cooperation can be shown to offer significant performance enhancements in terms of various performance metrics, including: increased capacity, improved reliability in terms of diversity, diversity-multiplexing tradeoff and bit or symbol error probabilities. These modern perspectives on and applications of relaying and cooperation have generated considerable research activity on relaying and cooperation within the communications, signal processing, and networking communities, and renewed interest within the information theory community.

4.2 Historical Background

The beginnings of coding and information theory go far back in time. Many fundamental ideas were understood long before 1948, when the two theories were first established on a firm basis in “A Mathematical Theory of Communication” written by Claude Shannon. Soon additional papers on information theory appeared in the journals and courses were taught on the subject in various universities.

As in most fields that suddenly open up, people though that this new field doesn’t have limitations. As a result of over-expectations of what information theory could do gradually set in a disenchantment. Now, a more just evaluation can be made, somewhere between the enthusiasm of the first days and the sad disappointment that slowly followed.

Information Theory sets bounds on what can be done but does little to aid in the design of a particular system. In general, information theory has ideas that are widely applicable to situations remote from its original inspiration. The applicability of the ideas is not exact, but the ideas are still useful.

At about the same time as information theory was created, and in about the same place, coding theory was also created. The basic paper, however was delayed until 1950. In the case of coding theory, the mathematical background was at first less elaborate than information theory, and for a long time received less attention from the theorists. With the passing of time, however, various mathematical tools such as group theory, the theory of finite fields have been applied to coding theory. Thus coding theory has now become an active part of mathematical research.

Both coding and information theory, give a central role to errors and are therefore of special interest, since in real-life errors are everywhere. Logically speaking, coding theory leads to information theory, and information theory provides the bounds on what can be done by suitable encoding of the information.

4.3 Model of the Communications System

The conventional communications system is modeled, from an information theory point of view by:

- An information source.
- An encoding of this source.
- A channel trough, which the signal is sent.
- A noise (error) source that is added to the signal in the channel.
- A decoding and hopefully a recovery of the signal information from the contaminated received signal.
- A sink for the information.

These elements are represented using a block diagram in *figure 4.1*.

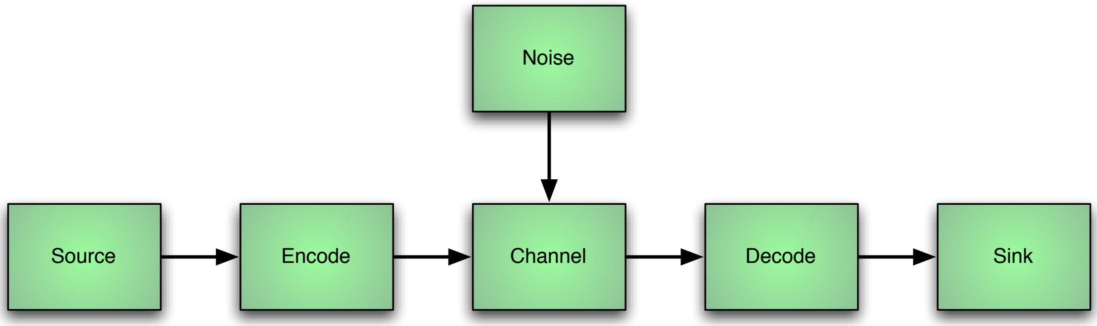

Figure 4.1. Model of the communications system

4.4 Information Source

We assume a sequence of symbols in a source alphabet $s_1, s_2, ..., s_q$ *so* its important to notice that information theory does not handle the meaning of the information, it treats only the amount of information.

Although information theory has a part devoted to analog (continuous) signals, we shall concentrate on digital signals for the simplicity of the theory and because analog signals are of decreasing importance in our technical society.

4.5 Encoding a Source Alphabet

It is conventional to represent information as being in one of two possible states, on or off. At present devices with two states, called binary devices, are much reliable than a multi-state devices. As a result, binary systems dominate all others. It is customary to use the symbols “0” and “1” as the names of two states.

For a system having *k* binary digits (bits) the total number of distinct states is, by elementary combinational theory 2^k. In general if we have *k* different devices, the total numbers of states is the product $n_1 \cdot n_2 \cdot \ldots \cdot n_k$.

In information theory we need to think of a source as a random, or stochastic, source of information, and ask how we may encode, transmit, and recover the original information so the designer must view all possible messages could be sent.

4.6 Basic Concepts about Information Theory

Coding theory answers the questions of how to design codes for white noise and how to compress the message when the probabilities of the message are known. We now need a general method for describing the structure of the source, so we need the concept of entropy. For us, entropy is simply a function of a probability distribution p_i.

Information theory combines noise protection and efficiency of channel use into a single theory. However, the simple model of a channel noise is sometimes unrealistic, and we will occasionally treat more general patterns errors. This leads to the important concept of channel capacity, which is explained in *section 4.6.5*.

In this section we introduce the basic concepts of information theory. For further details, the interested reader is referred to [Rom92].

4.6.1 Information

Suppose that we have the source alphabet of q symbols $s_1,s_2,...,s_q$ each with its probability $p(s_1)=p_1$, $p(s_2)=p_2$, ... , $p(s_q)=p_q$. When we receive one of those symbols, if a symbol with a low probability comes we would feel surprised, we would get more information than when a symbol with a higher probability came. The amount of information is defined as

$$I(s_i) = \log_2 \frac{1}{p_i}$$

The information of two different symbols is the sum of the information from each separately. The probabilities of two independent choices are multiplied together to get the probability of the compound event. As result we have

$$I(s_1) + I(s_2) = \log_2 \frac{1}{p_1 p_2}$$

4.6.2 Entropy

If p_1 is the probability of getting information $I(s_1)$, then on the average we get for each symbol s_1

$$p_i I(s_i) = p_i \log_2 \frac{1}{p_i}$$

From this it follows that on the average, over the whole alphabet of symbols s_i, we will get the entropy of the signal S having q symbols s_i and probabilities p_i.

$$H_r(S) = \sum_{i=1}^{q} p_i \log_2 \frac{1}{p_i}$$

The distribution consisting of just two events is very common. If p is the entropy of the first symbol, then the entropy function is

$$H_2(p) = p \log_2\left(\frac{1}{p}\right) + (1-p)\log_2\left(\frac{1}{1-p}\right)$$

It is possible to prove the relationship between the average code length L and the entropy $H(S)$ but it is not the objective of this work so we just say the entropy supplies a lower bound on the average code length L for any uniquely decodable system.

4.6.3 The Information Channel

An information channel is a statistical model of the medium through which the signal passes. In practice there are physical limitations on the fidelity with which the transmission can occur.

A channel is described by a set of conditional probabilities $P(b_j|a_i)$, which are the probabilities that an input a_i from an alphabet of q letters will appear as some b_j from an alphabet of s letters. It is represented in *figure 4.2.*

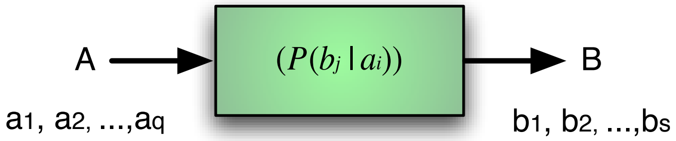

Figure 4.2. Information Channel

The probabilities $P_{i,j}$ characterize the channel completely. We are supposing that the channel is stationary, meaning that the probability does not change with time and that the errors are independent of each other.

4.6.4 Mutual Information

Consider again the previous transmission system. Prior to reception the probability of the input symbol a_i was $p(a_i)$. This is a prior probability of a_i. After reception of b_j, the probability that the input symbol was a_i becomes $P(b_j|a_i)$, the conditional probability that we sent a_i given that we received b_j. This is a posterior probability of a_i. The change in the probability measures how much the receiver learned from the reception of the b_j.

The difference between the information uncertainty before (a priori probabilities) and after reception of a b_j (a posterior probability) measures the gain in information due to the reception of the b_j. This information is called the mutual information and is naturally defined as

$$I(a_i;b_j) = \log_2\left[\frac{P(a_i \mid b_j)}{p(a_i)}\right]$$

If the two probabilities $p(a_i)$ and $P(b_j/a_i)$ are the same, then we have gained no information and the mutual information is zero. No information has been transmitted. It is only when we have learned something new about the probabilities of the a_i from the received b_j that the mutual information can be positive

Because of the inevitable noise the behavior of a channel can be understood only on the average. Averaging the mutual information over the alphabets

$$I(A;b_j) = \sum_i P(a_i \mid b_j)\log_2\left[\frac{P(a_i \mid b_j)}{p(b_j)}\right]$$

$$I(a_i;B) = \sum_j P(b_j \mid a_i)\log_2\left[\frac{P(b_j \mid a_i)}{p(b_j)}\right]$$

So we can define *I(A;B)* which is symmetric in the two alphabets, provides a measure of the information gain of the whole system and does not depend on the individual input and output symbols but only on their frequencies. It is called the system mutual information.

The system mutual information has the properties:

- $I(A;B) \geq 0$
- $I(A;B) = 0$ if and only if *A* and *B* are independent
- $I(A;B) = I(B;A)$ from symmetry

Using algebraic manipulations we can link the system mutual information with entropy

$$I(A;B) = H(A) - H(A \mid B)$$

In a channel with a high error probability, the system mutual information is near to zero, there is not relation between them, and in an ideal channel is maximum.

4.6.5 Channel Capacity

Given the conditional probabilities $P(b_j/a_i)= P_{i,j}$ which define a channel, it is necessary to know the maximum amount of information we can send through the channel.

The channel capacity is defined as the maximum over system mutual information over all possible assignments of the *p(a)*.

$$C = \max_{p(a)} I(A;B)$$

4.6.6 Shannon’s Theorem

Once the signal passes through the channel encoder, it has more bits than before because channel encoder adds redundancy bits to protect the code, we can define the code rate as

$$R_k = \frac{l_k}{n}$$

where n is the number of channel uses and l_k is the number of bits of information sent. Is useful to compare it with the channel capacity to know if it is possible to transmit the information trough the channel.

The code rate is not a speed measure but it leads to know how many redundancy is added by the channel encoder. To know about the speed of the transmission we can define the bit rate as

$$R_b = \frac{R_k}{Tc}$$

where Tc is the time used to transmit one channel use. We can know the critical bit rate replacing R_k value for C value. We can summarize Shannon's theorem with the following premises:

•A given communications system has a maximum rate of information C known as the channel capacity.

•If the information rate R is less than C, then one can approach arbitrarily small error probabilities by using intelligent coding techniques.

•To get lower error probabilities, the encoder has to work on longer blocks of signal data. This entails longer delays and higher computational requirements.

4.6.7 Gaussian Channel Capacity

The Gaussian channel model adds AWGN noise to the input transmission as shows *figure 4.3*.

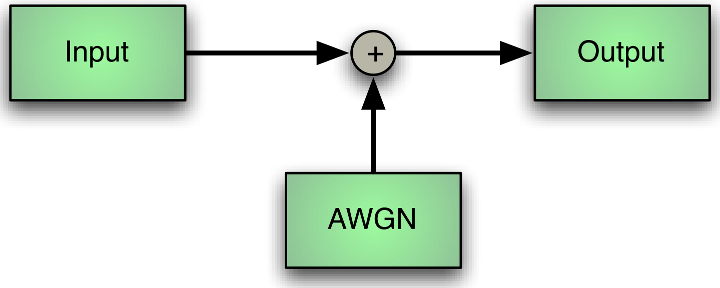

Figure 4.3. Gaussian Channel model

Having a limited power P_n and a bandwidth transmission B, we can define the noise power as

$$P_n = N_0 B$$

where N_0 is the noise spectral density. Linking it with the signal power we can define the SNR (Signal to Noise Ratio)

$$SNR = \frac{P_s}{N_0 B}$$

The Shannon-Hartley theorem states that the channel capacity is given by

$$C = B\log_2(1+SNR)$$

measured in bits per second and

$$C = \log_2(1+SNR)$$

measured in bits per second/Hz or bits per channel use.

This expression makes intuitive sense:

- As the bandwidth of the channel increases, it is possible to make faster changes in the information signal, thereby increasing the information rate.
- As SNR increases, one can increase the information rate while still preventing errors due to noise.
- For no noise, SNR→∞ and an infinite information rate is possible irrespective of bandwidth.

4.6.8 Outage Capacity

It is defined as the maximum data rate that can be maintained in all non-outage channel states times the probability of non-outage. In other words, outage capacity $C_{out,q}$ is defined as the rate supported the (100-q) % of the channel realizations, that is:

$$\text{Prob }(\log_2(1+SNR_i) < C_{out,q}) = q\%$$

Chapter 5

Analysis of the Cooperative System from an Information Theory Viewpoint

It is the first one of the two main chapters. Once discussed the principal concepts of the information theory and designed the reference model, the cooperative system performance is analyzed in terms of outage probability.

In this chapter the use of distributed Alamouti is studied and compared with a cooperative system where only the best relay is selected. Finally a system with a certain feedback error probability is compared with a system where an ideal feedback link is assumed.

5.1 System Parameters

The first step in designing a cooperative model is to define the outage probability and the principal system parameters as spectral efficiency or the number of available relays.

In this chapter, as we have commented before we analyze the outage probability of the system to study the system performance. The outage probability is the probability of having an unusable channel.

The outage probability definition is obtained directly from the information theory. Using the Gaussian channel capacity, we apply the following criterion to consider a useful channel.

$$C \geq 2R$$

where *R* is the spectral efficiency, explained in *section 5.1.1,* and *C* is the Gaussian channel capacity. We consider *2R* for a fair comparison with a non cooperative system. In other words, the cooperative systems use two time slots to transmit the information and by considering 2R, both the cooperative and non cooperative scheme will transmit the same amount of information.

5.1.1 Spectral Efficiency

The transmission schemes are further parametrized by the rate *r* (bits per second), or by the spectral efficiency (bits per channel use) defined in the continuous-time as:

$$R = \frac{2r}{W}$$

where *W* is the transmission bandwidth.

To evaluate the impact of the spectral efficiency over the outage probability it is necessary to simulate different R values. Now we consider a noise power and a signal power equal to one watt (0 dBW).

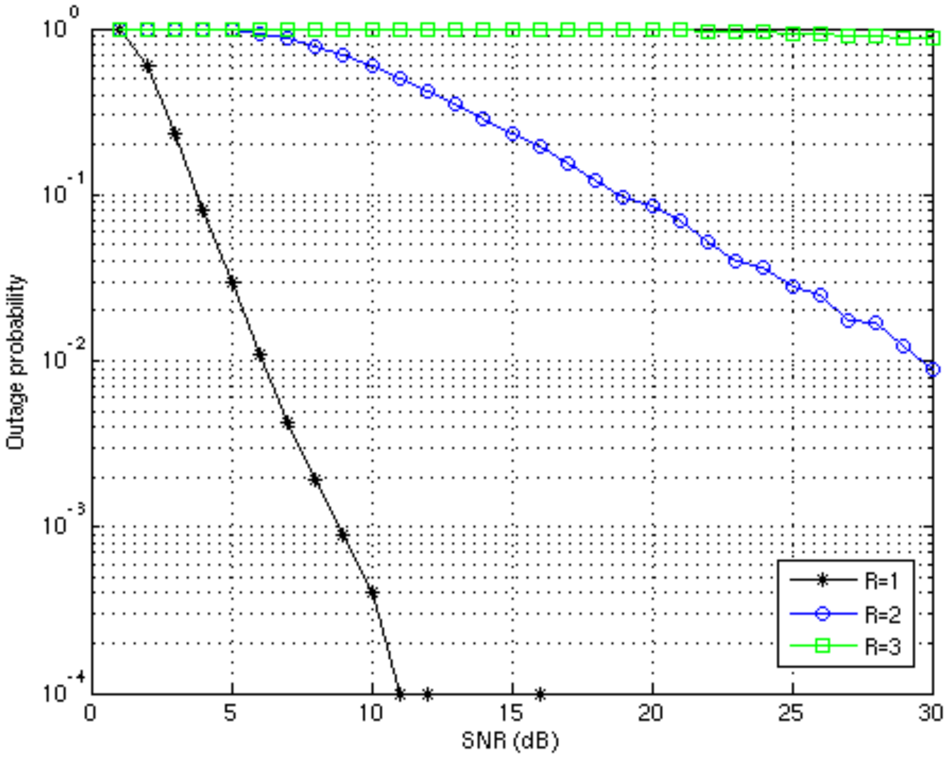

Figure 5.1. Outage probability using different values of spectral efficiency.

In *figure 5.1*, can be observed quite different results when the outage condition changes. R depends on many factors such as the used application, the channel...so we have decided to study a cooperative scenario using two bits per channel use.

5.1.2 Number of Relays

The main difference between a cooperative scheme and a traditional scheme is, as we have previously commented, the existence of relays between the source and the destination.

To show how the number of relays affects to the system performance, it is necessary to analyze the outage probability for different number of available relays.

Now we are considering the simplest algorithm. The source transmits the information to all relays but only some of them will be able to receive the information correctly. The set of operational relays is what we call decoding set. When they receive the information, the best relay transmits it to the destination. The destination chooses the best relay using a previous pilot signal. When it does not have any relays to choose, the system is in outage.

At the moment, we do not have in mind the distance between the relays, the source and the destination.

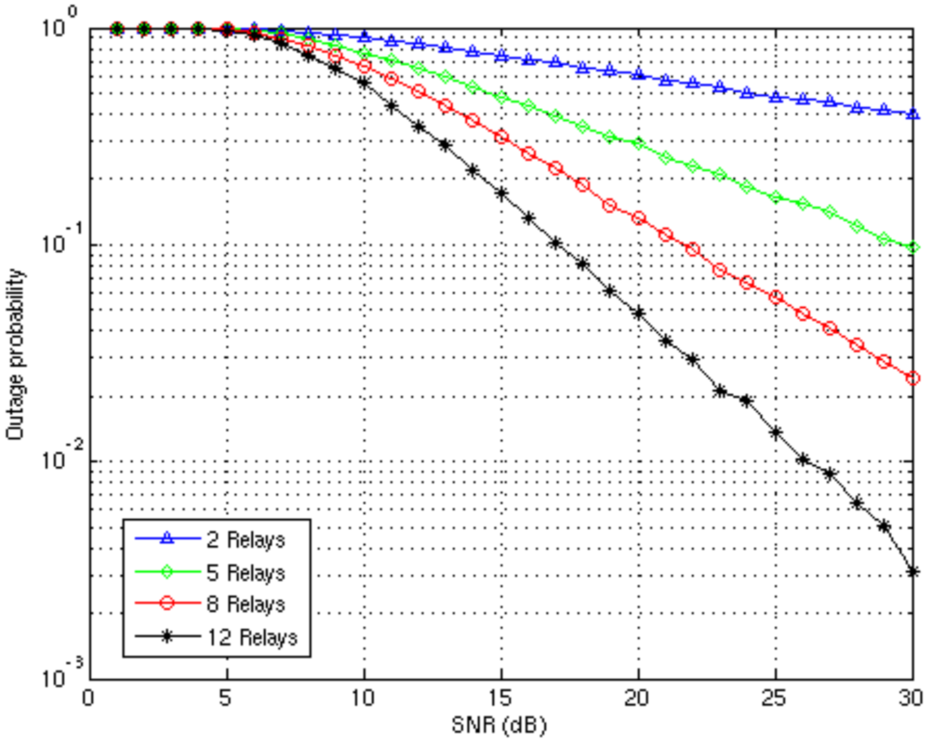

Figure 5.2. Outage probability using different number of relays

Figure 5.2 shows the importance of the number of relays. In the literature of cooperative communications different values are used. The number of relays depends on the scenario, for this reason, it is important to use a realistic number to evaluate a cooperative scheme in front of a traditional scheme, and we consider that 10 relays is a realistic number of relays. Then we adopt this value in this work, but the conclusion obtained could be applied to other scenarios.

5.2 Path Loss Modeling

In *section 5.1* the position of relays is not taken into account but in a real cooperative scenario, relays are positioned between the source and the destination, being each one placed in a different position. To take into account these different positions, in this work the channel is modeled as $h_A \sim CN(0, \sigma^2_{A,B})$(Rayleigh fading), being $\sigma^2_{A,B}$ the channel strength depending on the simplified path loss model [Gol05],

$$\sigma^2{}_{A,B} = \left(\frac{\lambda_c}{4\pi d_o}\right)\left(\frac{d_{A,B}}{d_o}\right)^{-\mu}$$

with λ_c standing for the carrier wave length, d_o is a reference distance, being d_o=100 in this work as in [Zha05], $d_{A,B}$ is the length of the link, and μ is the path loss coefficient being μ=3 in this work as in [Zha05].

We assume a block-fading channel where the channel response remains constant during one time slot and the channels between the relays and the destination are independently distributed.

The total transmission power P_s is evenly distributed among the source and the selected relays, and it is defined as

$$Ps = \frac{SNR \cdot Pn}{\sigma^2{}_{A,B}}$$

where *Pn* is the noise power and it is defined as 1 W (0 dBW) along this work. The instantaneous SNR for the Alamouti case is denoted by

$$SNR_i = \frac{Ps}{4Pn}\left(|h_1|^2 + |h_2|^2\right)$$

and for the Best Relay case

$$SNR_i = \frac{Ps}{2Pn}|h_1|^2$$

where h_1 and h_2 are the two channel response of the two selected relays.

5.3 Transmission Options

In a cooperative scenario there are multiple relays in order to improve the information flow between the source and the destination. These relays are not always available because the environment is changing continuously, so it is necessary to find a way to manage the information that arrives to the destination from them. Two strategies are considered in this work:

- Best Relay: The destination receives the pilot signal sent by the relays, and estimates which is the best relay, discarding the other relays.
- Alamouti: The destination chooses the two best relays. The information is sent by the two relays using the Alamouti code.

5.4 System Performance

To study the performance of the Best Relay and the Alamouti option, their outage probability is analyzed. As we have commented in *section 5.1*, a communications system is in outage when the SNR of the receiver does not exceed a certain threshold. To define this threshold the channel capacity theorem is used. It must satisfy the following expression in terms of SNR

$$\log_2(1+SNR_i) \geq 2R$$

where R is the spectral efficiency and SNR_i is the instantaneous *SNR.* We consider *2R*, as we have commented before, because a cooperative system uses two time slots to transmit the information, and it has to transmit the same information that a non cooperative system using the same time. In *section 5.1.1*, R has been defined as 2 bits per channel use.

From the above equation, the instantaneous SNR must satisfy:

$$SNR_i \geq 2^{2R} - 1$$

The average SNR is defined as

$$SNR = \frac{Ps}{Pn}$$

where *Ps* is the total transmission power and *Pn* is the noise power. The instantaneous SNR can be approximated as follows

$$SNR_i = \frac{\mathrm{E}\left[|s_r|^2\right]}{\mathrm{E}\left[|n|^2\right]}$$

where *n* represents AWGN noise and s_r is the received symbol. It can be expressed as the combination of the sent symbol *x* and the channel response *h*:

$$s_r = h \cdot x$$

By using the above equation, we can rewrite the instantaneous SNR expression as

$$SNR_i = \frac{\mathrm{E}\left[|h \cdot x|^2\right]}{\mathrm{E}\left[|n|^2\right]} = \frac{|h|^2 \cdot \mathrm{E}\left[|x|^2\right]}{\mathrm{E}\left[|n|^2\right]} = \frac{|h|^2 \cdot Ps}{P_n}$$

and it is equivalent to

$$SNR_i = |h|^2 \cdot SNR$$

In order to take distances into account, we consider a scenario where the relays are situated along the axis of symmetry between the source and the destination. Notice that *d* is the distance between the source and the destination, and the distance between the first relay and the N relay.

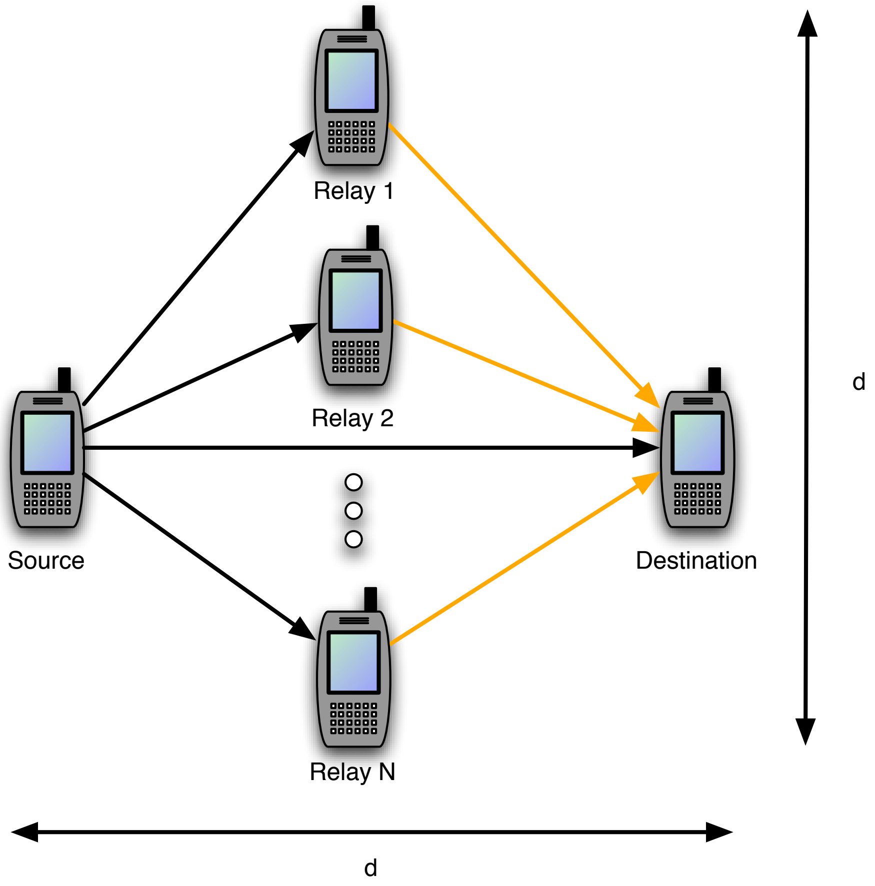

Figure 5.3. General model

In *figure 5.3* the black arrows represent the transmission of the first time slot and the orange arrows represent the transmission of the second time slot.

We can compare the performance of this cooperative model with the performance of a system where a cooperative strategy is not used (Direct transmission).

To calculate the SNR of the Alamouti and the Best Relay options, the SNR expressions of *section 5.2* has been used.

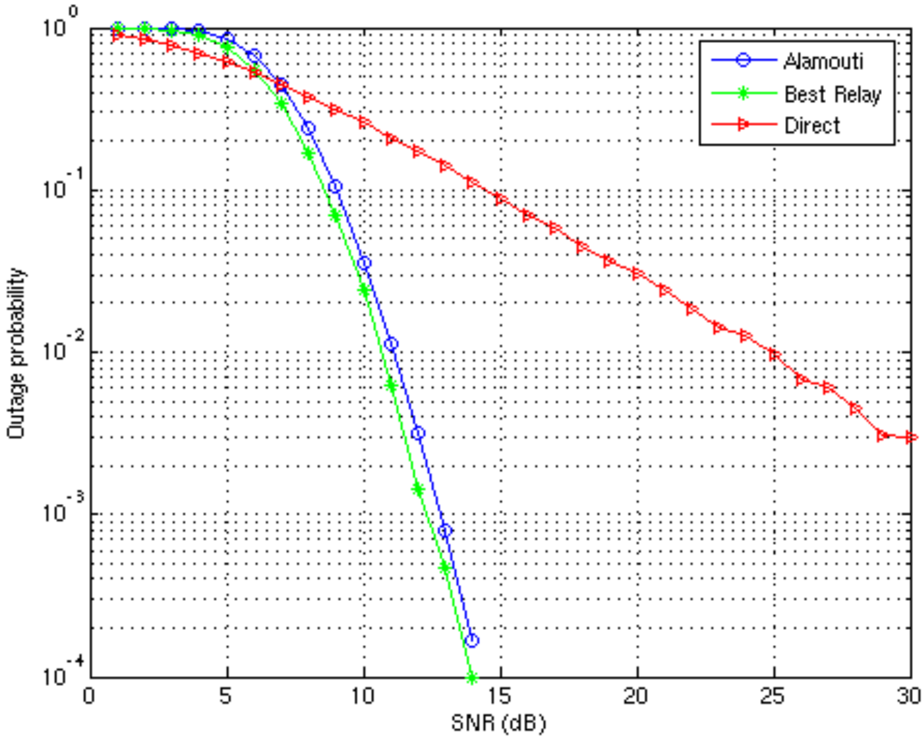

Figure 5.4. Outage probability when distances are taken into account

Figure 5.4 shows a better performance of a cooperative system (Best Relay and Alamouti) in front of a non cooperative system. The improvement is about 5 dB with a target outage probability of 10^{-1} and about 15 dB with a target of 10^{-2}. The difference increases with the SNR.

With this model we get always a better performance choosing the Best Relay method. It is because the destination always chooses the correct relay, and it never gets wrong. The case where it is possible to choose a wrong relay is studied in *section 5.8.*

In *figure 5.4,* it can clearly be observed two different slopes for the cooperative and the non cooperative transmission. To explain it, in *section 5.5* we have done a slope analysis of the curves.

5.5 Slope Analysis

The slope analysis has been done using the outage probability expression derived in [Vic08]:

$$P_{out}(y) = \left(1 - e^{-\frac{2y}{\bar{\gamma}}}\right)^{K}$$

which can be expressed as follows for the asymptotic high SNR regime ($\bar{\gamma} \to \infty$):

$$P_{out}(y) = \left(\frac{2y}{\bar{\gamma}}\right)^{K} + o\left(\left(\frac{1}{\bar{\gamma}}\right)^{K}\right)$$

The diversity order is defined as

$$d = \lim_{\bar{\gamma} \to \infty} \frac{-\log(P_{out})}{\log(\bar{\gamma})}$$

and using the outage probability expression defined before

$$\log(P_{out}) = \left(k \cdot \log\left(\frac{2y}{\bar{\gamma}}\right)\right) = -k \log\left(\bar{\gamma}\right) + k \log(2y)$$

so the diversity order is

$$d = \lim_{\bar{\gamma} \longrightarrow \infty} \frac{+k \log\left(\bar{\gamma}\right) - k \log(2y)}{\log(\bar{\gamma})} = k$$

where k is the diversity gain which defines the slope. From the above equation we can observe that if the SNR increases the outage probability curve goes faster to zero.

5.6 Relay Selection Criteria

To use the Alamouti STBC, the destination must choose the two best relays. To this end we have two possible criteria, to choose the two relays with the best instantaneous SNR or to choose the two nearer relays.

To choose the two nearer relays the destination needs to know the position of the active relays every moment. This estimation is outside the objective of this work, although a possible option would be to incorporate *Global Positioning System* (GPS) receivers in the relays, and send the GPS position to the destination together with the pilot signal.

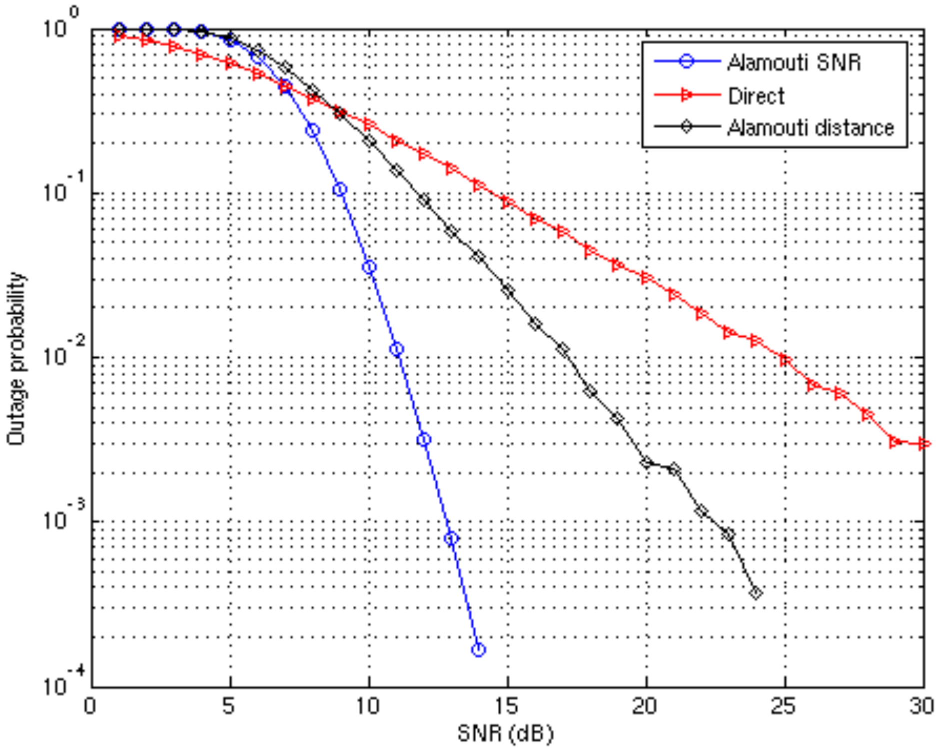

Figure 5.5. Outage probability using two different relay selection methods

Figure 5.5 shows a better performance if the two relays are selected by the SNR criterion. This is because the destination does not have always a better signal quality when the relay is closer. Because of this we will continue using the SNR criteria.

5.7 Relays Position

Previously it has been assumed a vertical position of the relays but different geometries in the scenario can be applied. It is interesting to compare the outage probability of a cooperative system when the position of the relays is different to the vertical distribution, as rectangular distribution (*figure 5.6*) or horizontal distribution (*figure 5.7*) to see how much can change the performance due to the position of the relays. Only the Alamouti option is considered now.

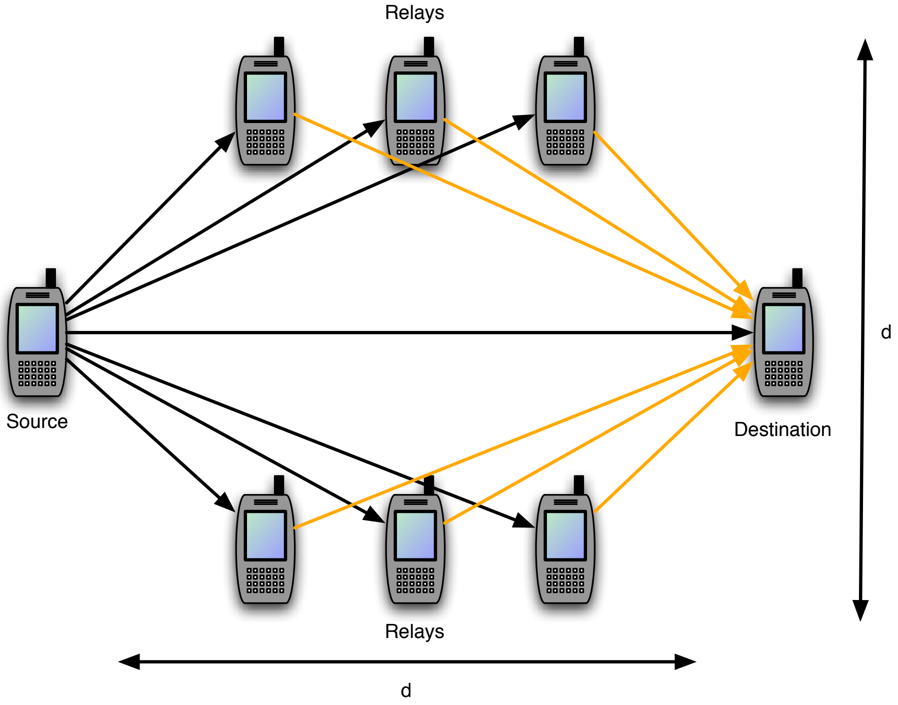

Figure 5.6. Rectangular geometry

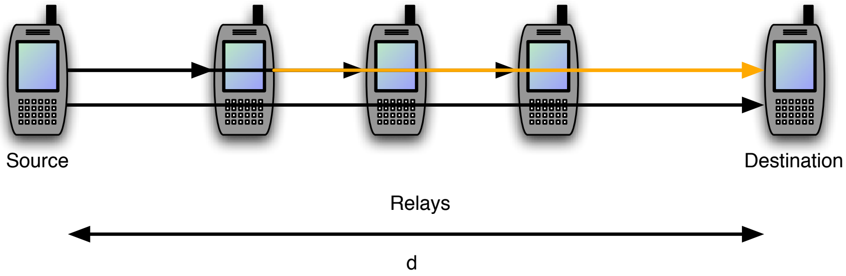

Figure 5.7. Linear geometry

With this study we try to get a different performance curves to estimate which is the average performance of the system.

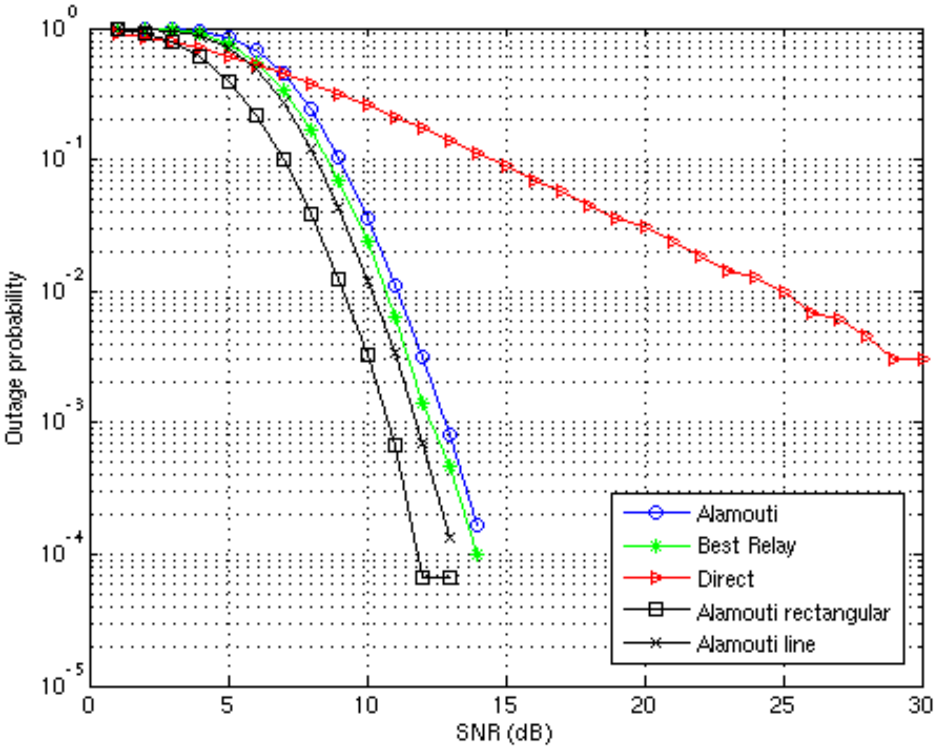

Figure 5.8. Outage probability using different geometries

Figure 5.8 shows a better performance if the relays have a rectangular geometry. This result is interesting if the scenario is static (like fixed sensors) but in a variant scenario the position of the relays are not geometric, they are completely random.

To study different geometries offers information on how the system behaves in different situations but it does not give us a real prospect of the system performance in a mobile scenario. From this section, a cooperative scenario where relays have been placed at random in a $d_x d$ square is considered.

5.8 Feedback Error

The proper behavior of a cooperative system relies heavily on the signaling in the source-to-relays , relays-to-destination and destination-to-relays links (the last one is the feedback link). The feedback link is of particular interest because the choice of the transmitting relays depends of it. To choose the correct relays is very important to have an efficient power transmission, avoiding to use relays with low instantaneous SNR.

In the previous sections the feedback link has been assumed ideal. In this section, we study the system performance when there is a certain probability of a wrong relay selection due to channel fading. To observe the system performance in this case, various feedback error probabilities have been analyzed.

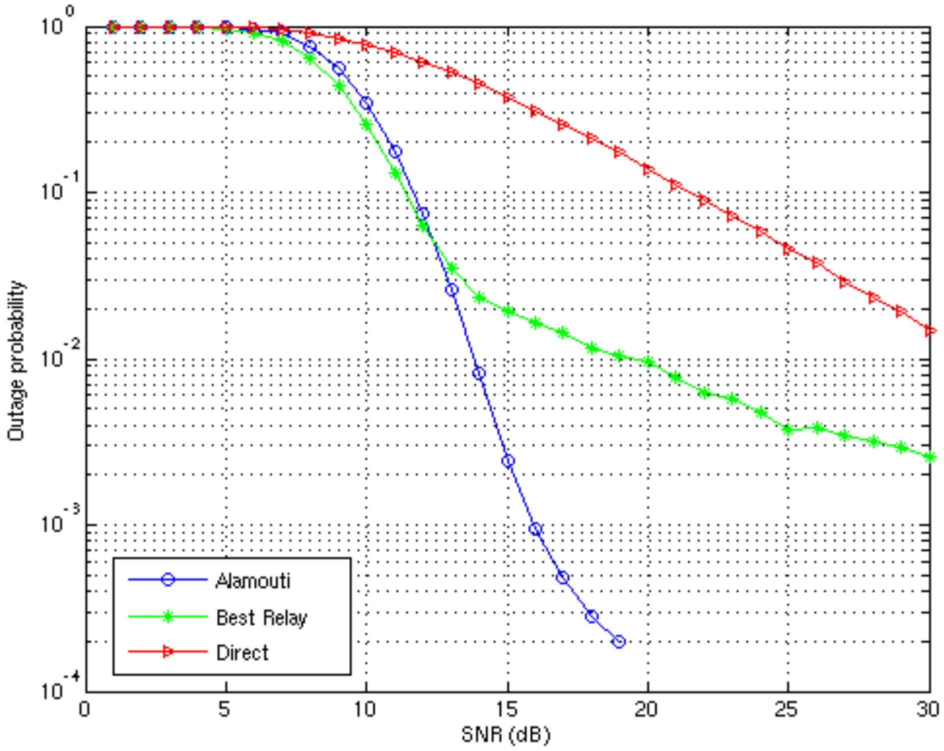

Figure 5.9. Outage probability with a feedback error probability of 1%

Figure 5.9 shows a better performance of the Alamouti option when the SNR is above the 12 dB. This is because the diversity of the Alamouti code is two when h_1 and h_2 are independent, as we have commented in *section 2.6.1*.

It shows why the Alamouti option is a great option in a real scenario, because it always have some feedback error probability.

Increasing the feedback error probability the outage probability of Best Relay and the outage probability of Alamouti increases considerably:

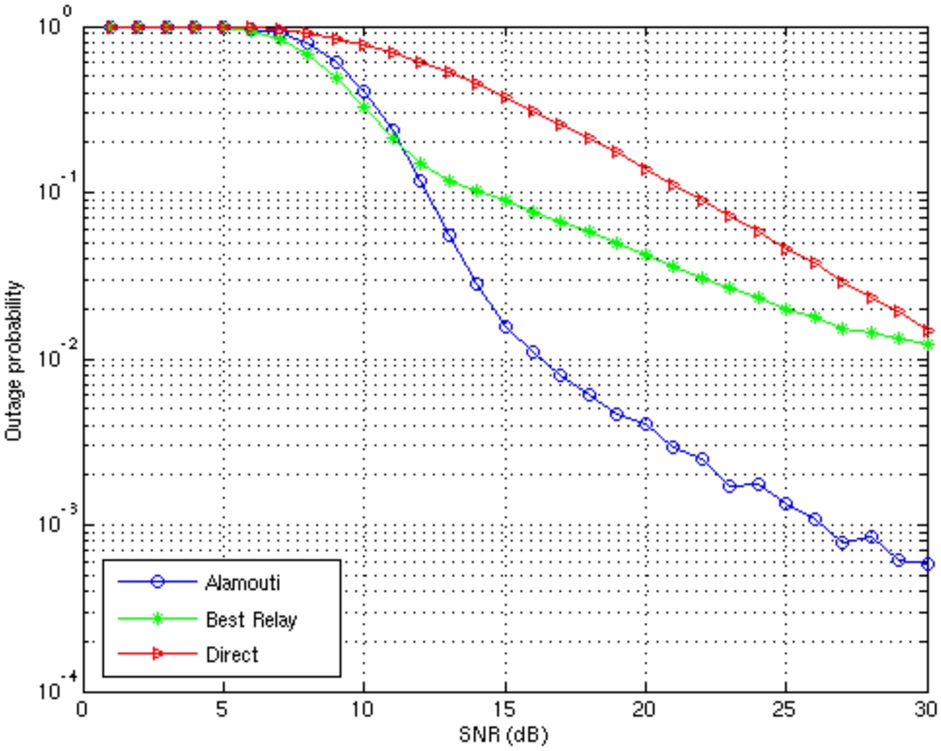

Figure 5.10. Outage probability with a feedback error probability of 5%

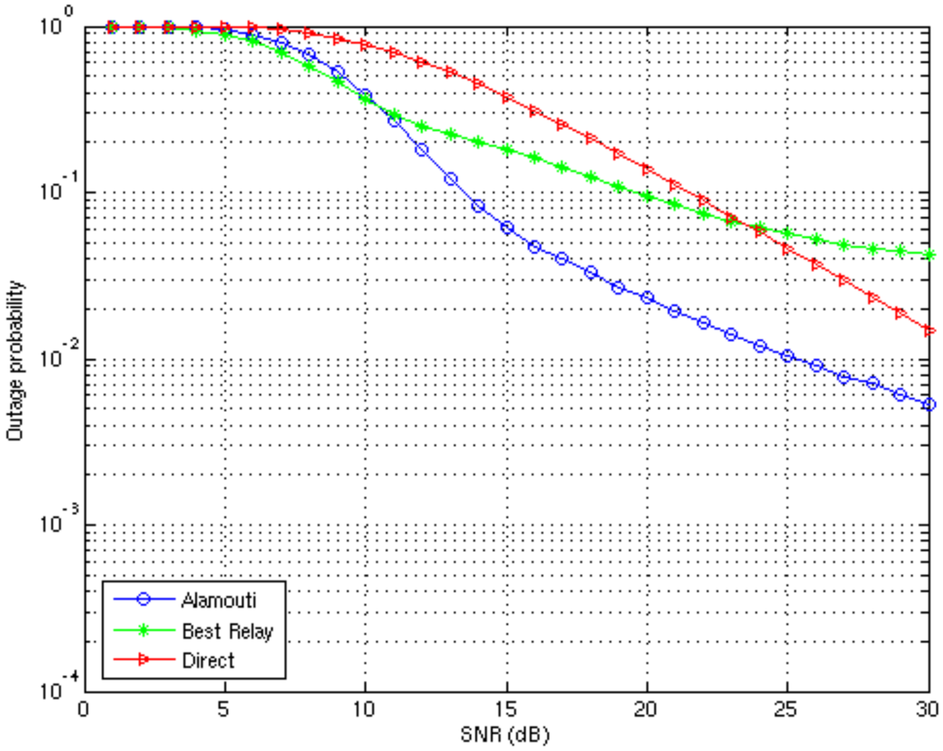

Figure 5.11. Outage probability with a feedback error probability of 10%

Figures 5.10 and 5.11 show that with a high feedback error probability and a high SNR, a cooperative model is useful using the Alamouti option but it could not be useful using the Best Relay option.

If the target outage probability is fixed in 10^{-2}, with a 1% of feedback error probability, the Alamouti option needs to increase the SNR around 3 dB to have the performance of the system when it does never have feedback error and the Best Relay option must increase the SNR up to 8 dB. This difference increases when the feedback error probability increases. In the 5% of feedback error probability case, it is necessary to increase the SNR up to 20 dB using the Best Relay option and only 5 dB using the Alamouti option.

Figures 5.9 to 5.11 show the improvement when a STBC, like Alamouti, is used to transmit the signal, specially in scenarios with a high feedback error probability, so it can be assured that the Alamouti option is more robust to errors due to feedback.

Chapter 6

Real Cooperative System

In *chapter 5* a cooperative scheme has been studied from an information theory point of view in order to establish a baseline scenario. This study has enabled to us to understand theoretically how a cooperative system works, but in a real system we cannot use the information theory to determine the packet or symbol error probability.

This chapter uses the ideas learned from *chapters 3* and *5* to evaluate the performance of a realistic cooperative system where practical modulation schemes are taken into account. In this work a QPSK modulation is used.

First we model a realistic cooperative scheme introducing the new concepts on it, evaluating the system performance when the feedback error probability is not considered and when it is considered. Then we propose several protocols to obtain different relay selection methods comparing their performance with the system performance obtained when these protocols are not used.

6.1 Real Scenario Model

The relays are located in a $d_x d$ square between the source and the destination, as in *figure 6.1,* and the distances between source-relays and relays-destination are random.

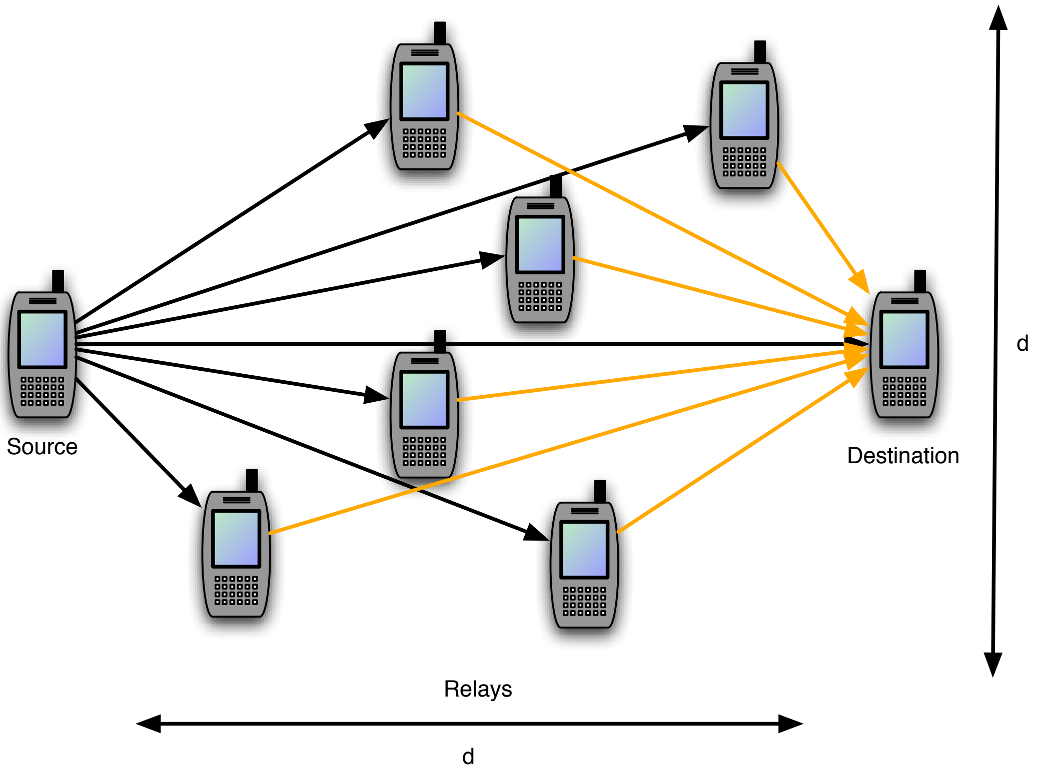

Figure 6.1. Random distances

In the real, model the Alamouti option continues choosing the two relays by the instantaneous SNR criterion, because as we have observed in *section 5.6,* its behavior is better than the two nearer relays criterion.

Until now we have studied the outage probability because the information theory does not consider packets, but in a modern wireless system the messages are divided in packets, so in this chapter it is better to evaluate the *Packet Error Rate* (PER).

To calculate the PER, the following expression is considered

$$P_{packet\_error} = 1 - P_{packet\_ok}$$

and the probability to receive a correct packet is

$$P_{paquet\_ok} = \prod_{k=1}^{L} P_{symbol\_ok}$$

where L is the quantity of symbols in a packet. This expression is equal to

$$\prod_{k=1}^{L} P_{symbol\_ok} = P^{L}{}_{symbol\_ok} = (1 - P_{symbol\_error})^{L}$$

so the PER is

$$P_{packet\_error} = 1 - (1 - P_{symbol\_error})^{L}$$

In this work a packet length of 10 symbols is considered as it is shown in *figure 6.2*.

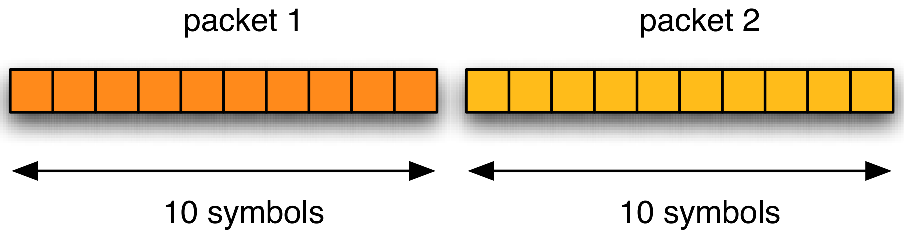

Figure 6.2. Transmission Packets

In this chapter, a QPSK digital modulation is used, so the symbol error probability is defined as

$$Pe_{symbol} = 2Q\left(\sqrt{\frac{Es}{No}}\right) - Q^{2}\left(\sqrt{\frac{Es}{No}}\right)$$

Where Es is the symbol energy and No is the noise spectral density.

Choosing R (spectral efficiency) equal to W (bandwidth), it is possible to rewrite the symbol error probability expression using the SNR expression:

$$SNR = \frac{Es \cdot R}{No \cdot W} = \frac{Es}{No}$$

as

$$Pe_{symbol} = 2Q\left(\sqrt{SNR}\right) - Q^2\left(\sqrt{SNR}\right)$$

The real cooperative model considers random channels, so we have to consider the instantaneous SNR and not the average SNR. The instantaneous SNR expression is different when the Best Relay option is chosen and when the Alamouti option is chosen. This two expressions have been discussed in *section 5.2*.

In this work we consider a system which needs a PER of 10% to consider that a relay belongs to the decoding set.

6.2 System Performance

Using the introduced model, in this section we analyze the performance of a real cooperative system in terms of PER. As in the previous sections, the Alamouti option, the Best Relay option and the Direct transmission are compared.

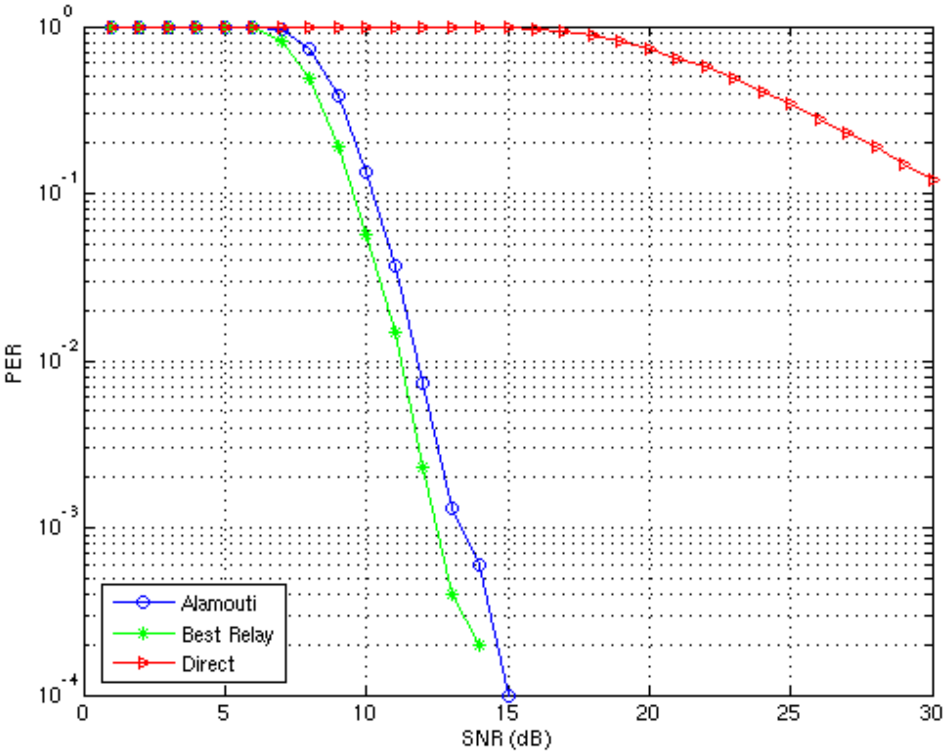

Figure 6.3. PER using 10 Relays without feedback error probability

Figure 6.3 shows a better performance of the Best Relay option because the feedback error probability is not considered, so it is always better to assign all the power to the best relay. There is a wide difference between to use the direct link and to use a cooperative system as Alamouti or Best Relay.

The obtained gain using a cooperative option is approximately 20 dB when the target PER is 10^{-1} and it increases with the SNR.

6.3 Feedback Error Probability

In *chapter 5* the effect of the feedback error probability has been studied. As we have observed, it can make useless trying to improve the communications using a cooperative strategy. In this section we analyze the system performance of the real cooperative system when the feedback channel is not ideal.

In this section we does not introduce manually the feedback error probability. It is not necessary because, as we have commented in *section 3.1*, we are considering a TDD duplex mode so the feedback error probability can be estimated by the destination because it is related to the system SNR. It allows to the destination to estimate the feedback error probability as the *Bit Error Rate* (BER) value of the system. The BER is obtained using the pilot signals sent by the relays

We are using a QPSK modulation so the BER is the half of the *Symbol Error Rate* (SER), because two bits per symbol are sent with this modulation.

In this section we have used some different number of relays to analyze the system performance.

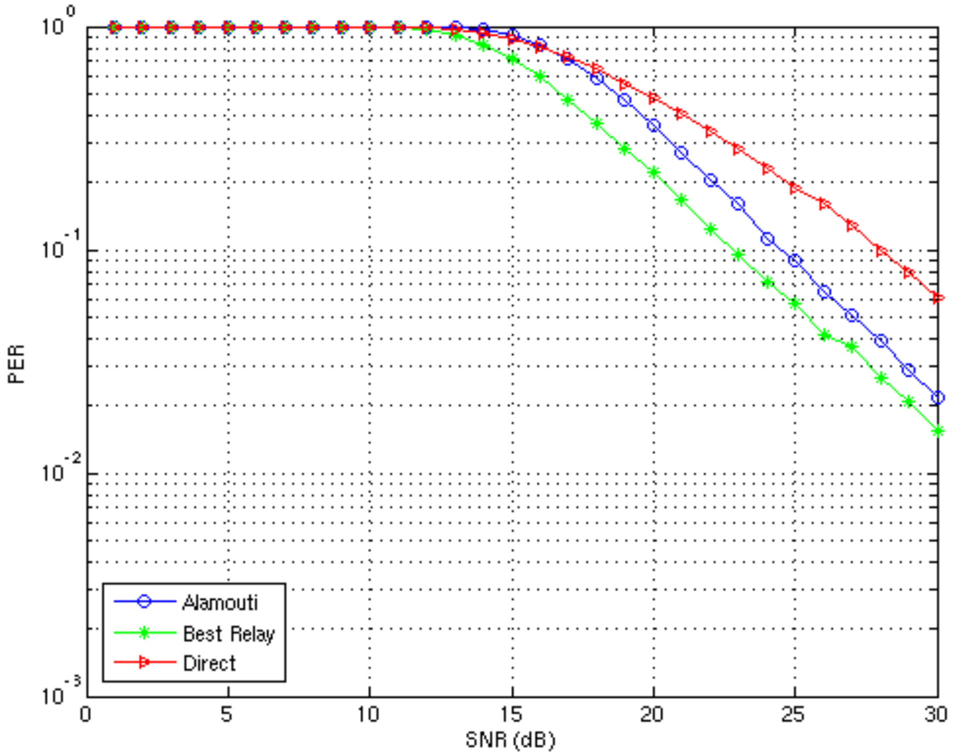

Figure 6.4. PER using 2 Relays

This *figure 6.4* shows the performance when only two relays can be used. The best option in this case is to choose the Best Relay option because the Alamouti option needs two operative relays, and if one of them is not in the decoding set, it would be impossible to transmit the information correctly. The difference between both options becomes smaller when the SNR increases. It is because the BER, and consequently the feedback error probability, decrease.

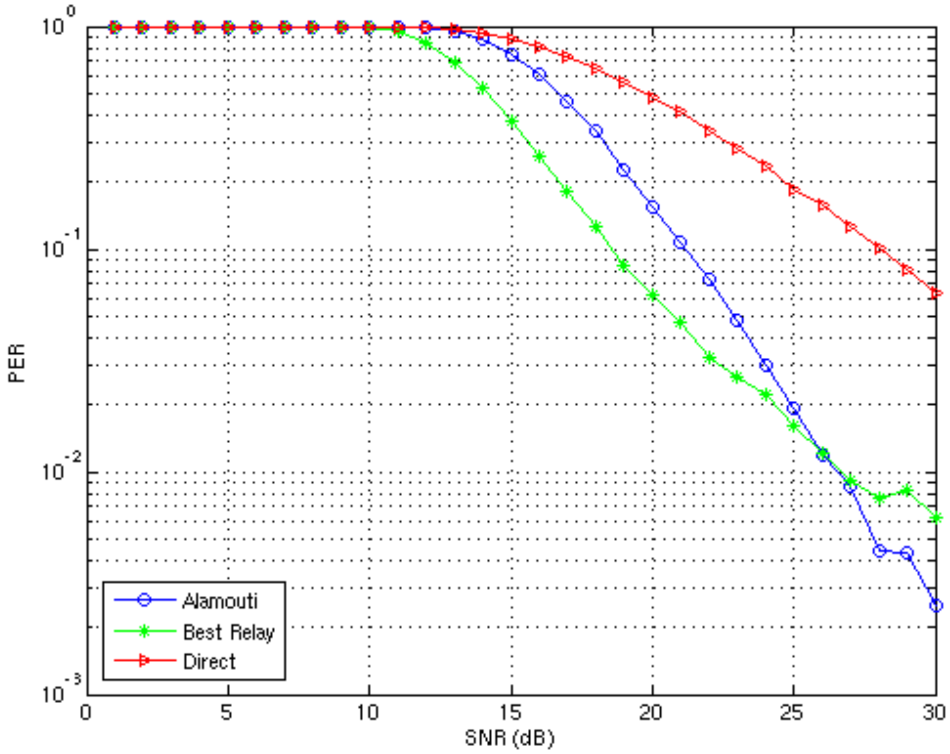

Figure 6.5. PER using 3 relays

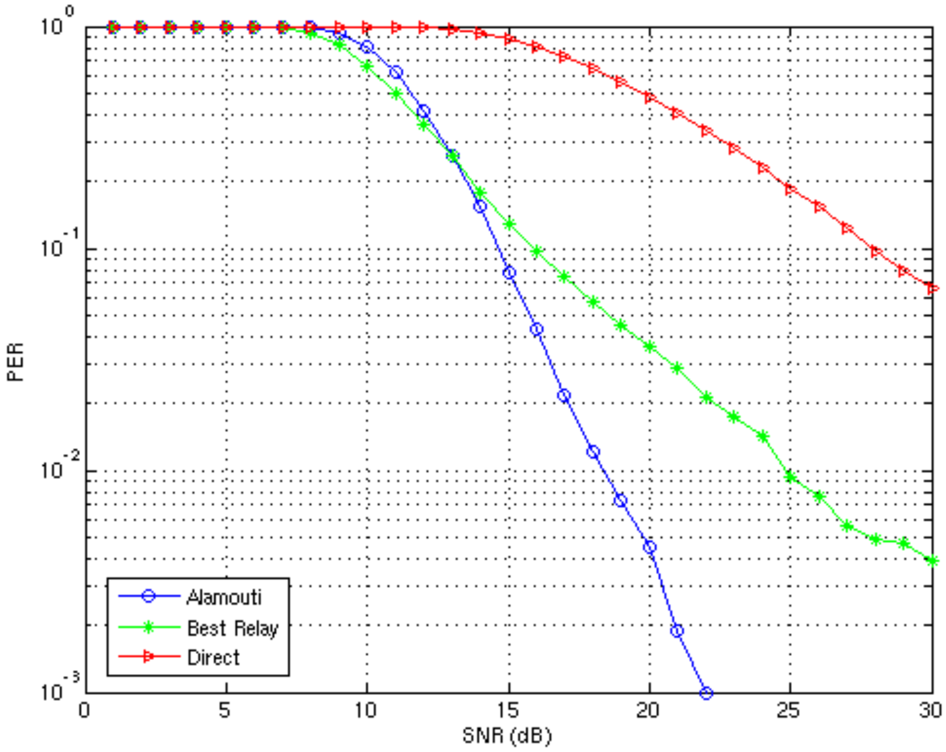

Figure 6.6. PER using 4 relays

With 3 relays (*figure 6.5*) the performance is improved and it shows a better behavior of the Best Relay option in front of the Alamouti option when the SNR is lower than 26 dB.

Increasing the number of relays, the Alamouti option obtains better results than the Best Relay option from lower SNR. In the case of 4 relays, *figure 6.6* shows that this option is better from an SNR of 13 dB.

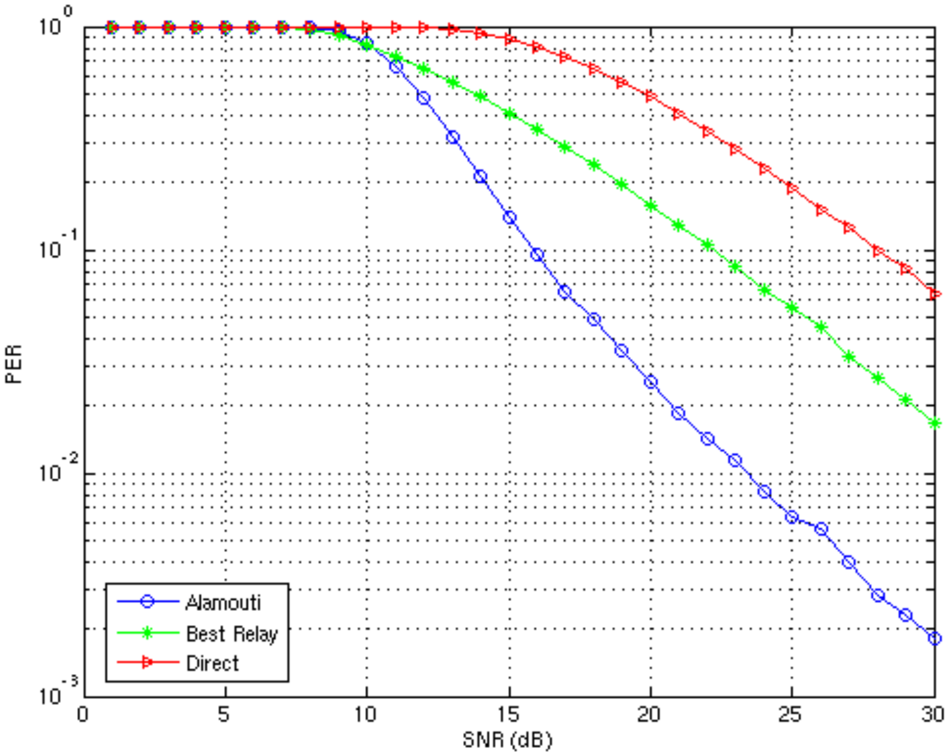

Figure 6.7. PER using 5 relays

Using *figures 6.5* to *6.7*, we can see how much the Alamouti option is more robust than the Best Relay option to feedback errors. It had been observed in *chapter 5*.

When the feedback error probability is taken into account, it is possible to choose a relay that does not exist, e.g. if the destination chooses the relay number 4 (in binary is: 0100), and it exists an error on the third bit, it would choose the relay number 6 (in binary is: 0110), but if we only have 4 available relays, the information will not be transmitted by any relay. To solve it, we have proposed a new strategy: When we choose a relay that does not exist, the nearer relay to the destination (the relays have to know the total number of relays, and their position) transmits the information. Using this strategy we did not get a considerable improvement so this idea was discarded due to the increase of the system complexity. Then we have proposed a new strategy using a new channel model where a certain correlation coefficient is assumed. It is widely explained in *section 6.6.*

6.4 Protocols

As we have proved before, the system performance varies depending on the transmission option chosen. This section proposes several protocols aimed at optimizing the choice of the transmission option depending on different criteria.

A useful protocol must chooses between a cooperative and a non cooperative transmission, and if a cooperative transmission is selected it must chooses the optimal cooperative option, Alamouti or Best Relay.

The proposed protocols use the direct link between the source and the destination if it is possible, when it is not possible, they choose a cooperative option following different criteria.

In this work we evaluate the protocols using 4 relays, but it can be adapted to any number of relays. It is because when the system uses more relays, the Alamouti option is always better, and the Best Relay option becomes useless.

6.4.1 Feedback Error Probability Aware Protocol

The first proposed protocol takes the existence of feedback error probability as a criterion to choose the best option. As we have observed in *sections 6.2* and *6.3*, the Alamouti option has a better performance in front of the Best Relay option, when the feedback error probability is considered, but it has a worse performance when there is not feedback error possibility.

Using this protocol, the system compares the probability to have a feedback error with a threshold. If its value is above the threshold, it considers the existence of feedback error, so it chooses the Alamouti option and if its value is lower, it chooses the Best Relay option.

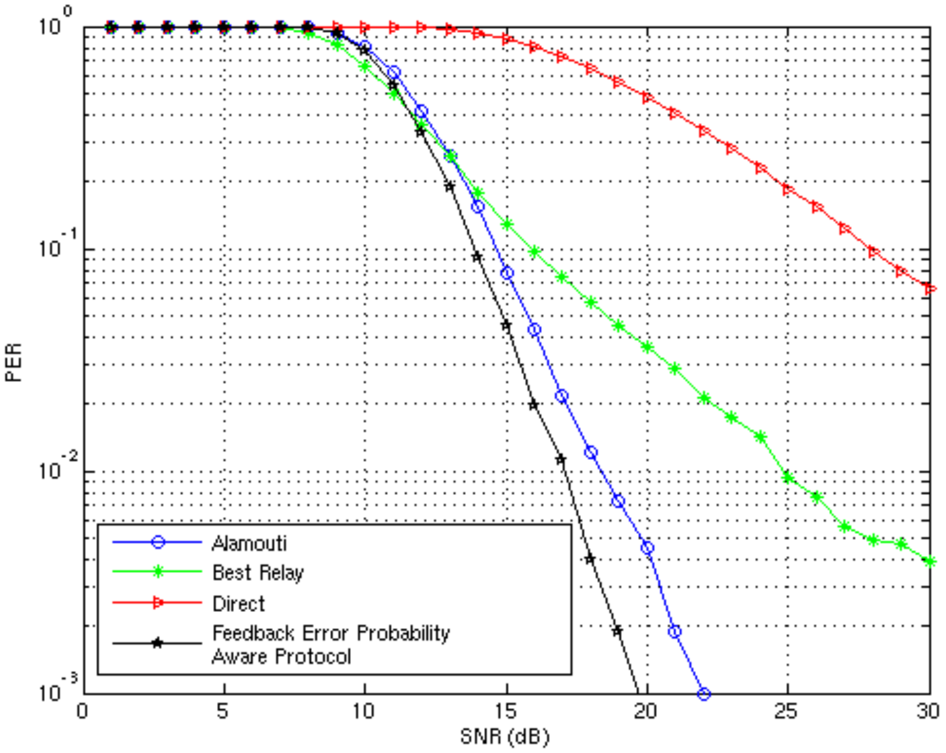

Figure 6.8. PER of the Feedback Error Probability Aware Protocol

Figure 6.8 shows a gain of 0.5 dB over the Alamouti option when a target PER of 10^{-1} is considered and a gain about 1 dB when the target PER is 10^{-2}. This gain grows when the SNR increases.

The estimation of the feedback error probability must be done from the pilot tone using the BER estimation, as we have commented in *section 6.3*.

6.4.2 Feedback Error Probability and Average SNR Aware Protocol

This protocol is an improvement of the previous protocol. It uses also the feedback error probability as the principal criterion.

In this case a guide chart has been developed to choose the best transmission option. To establish the criterion, a study of the PER vs. feedback error probability for different SNR has been done.

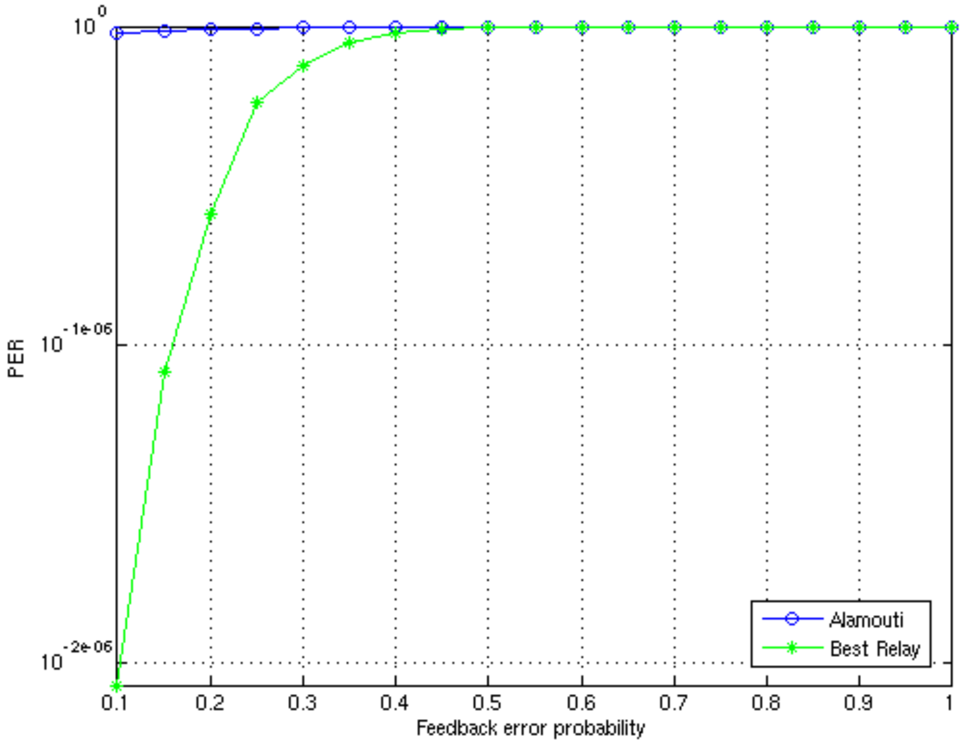

Figure 6.9. PER vs. feedback error probability. SNR=5 dB

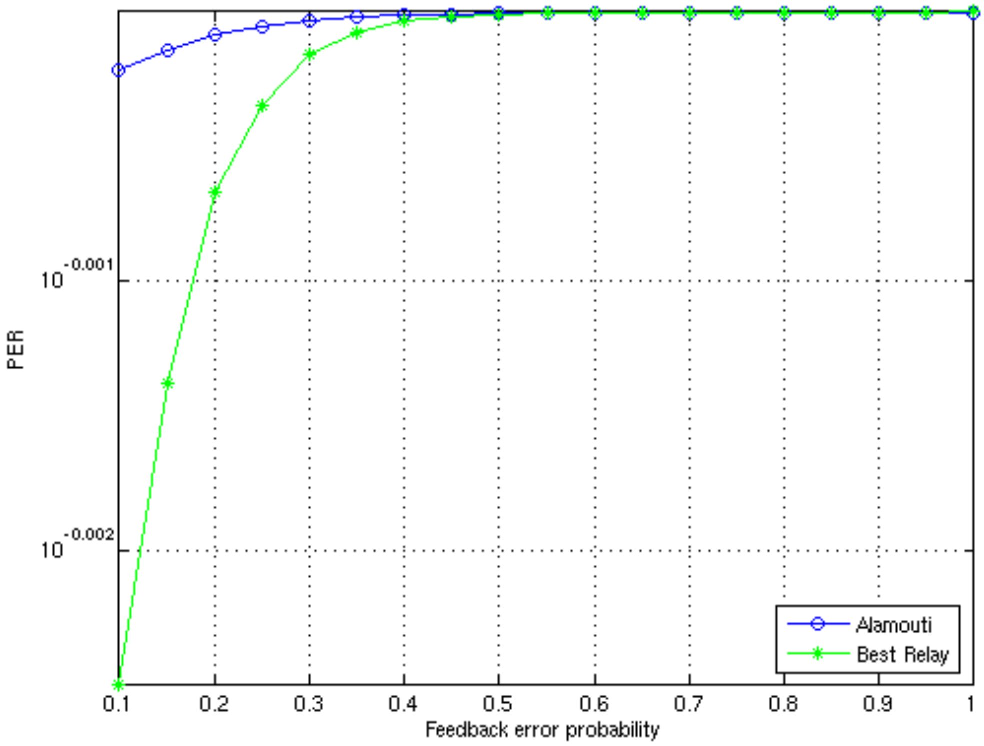

Figure 6.10. PER vs. feedback error probability. SNR=7 dB

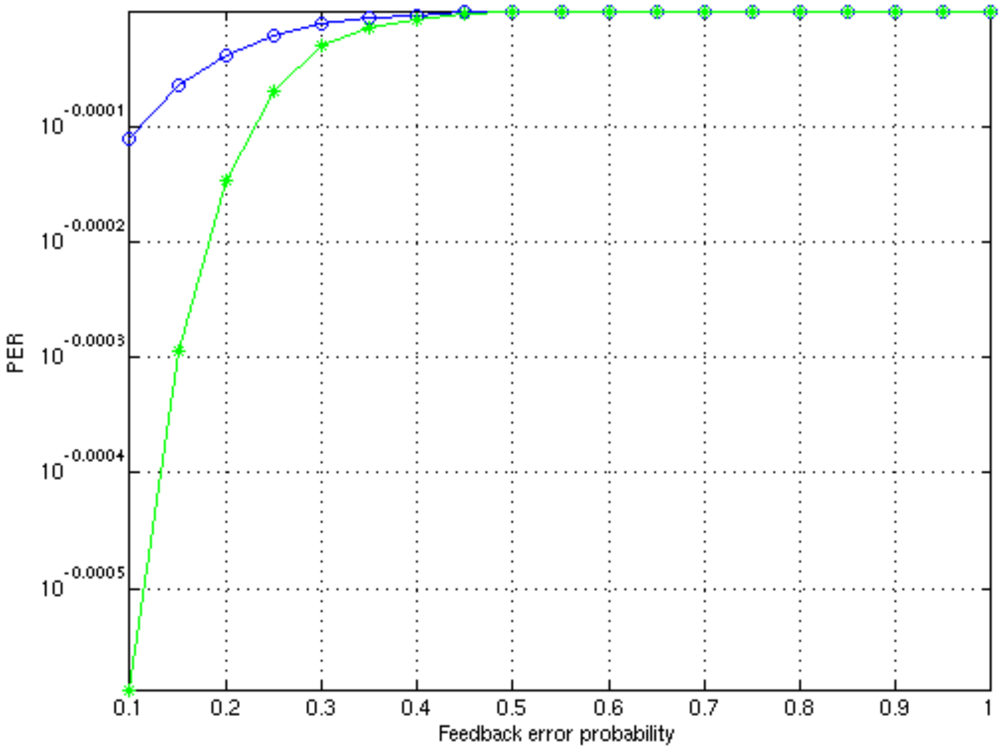

Figure 6.11. PER vs. feedback error probability. SNR=8 dB

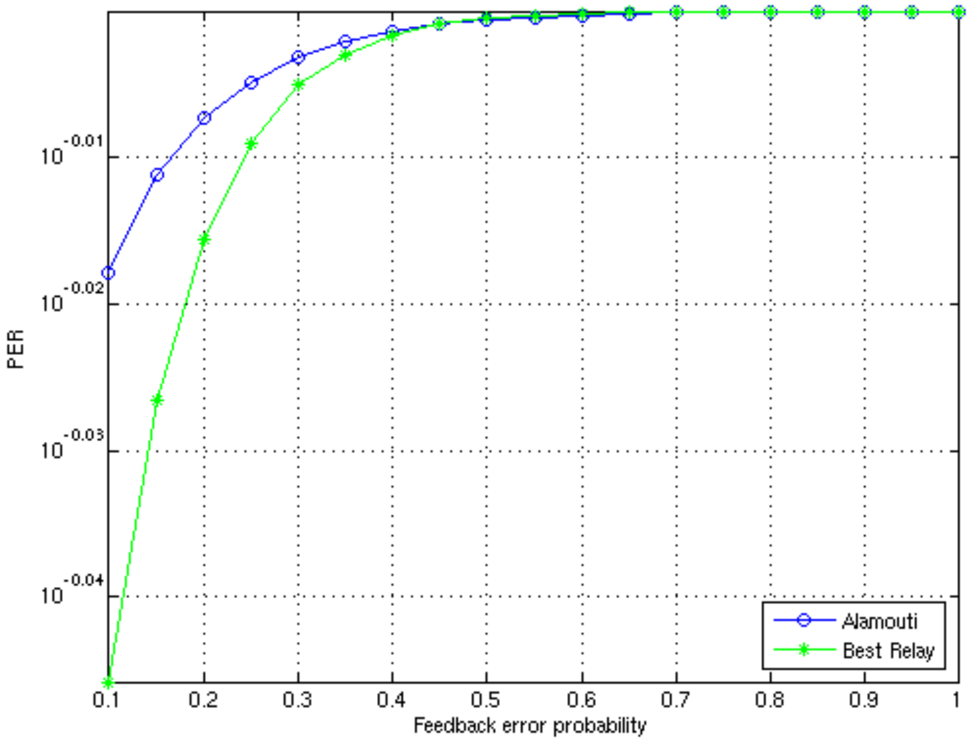

Figure 6.12. PER vs. feedback error probability. SNR=9 dB

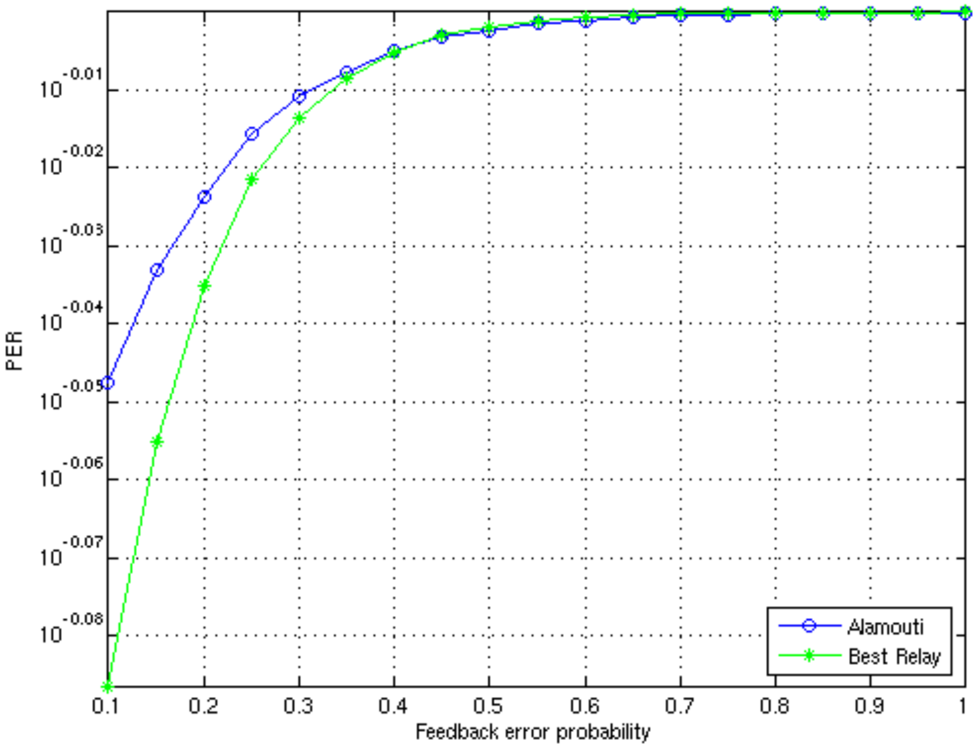

Figure 6.13. PER vs. feedback error probability. SNR=10 dB

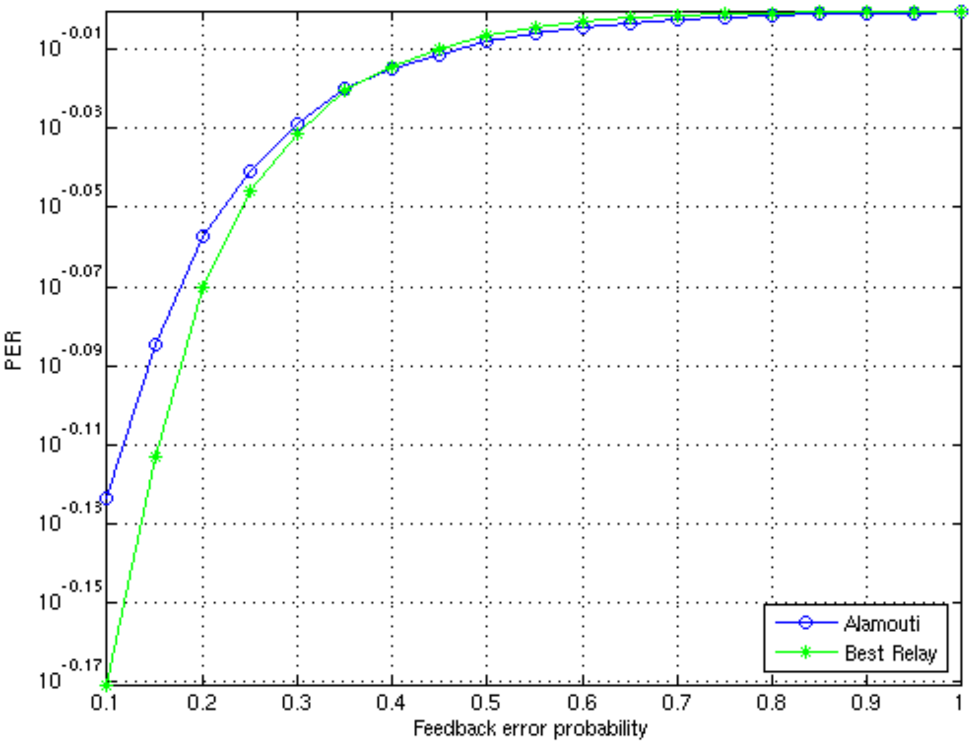

Figure 6.14. PER vs. feedback error probability. SNR=11 dB

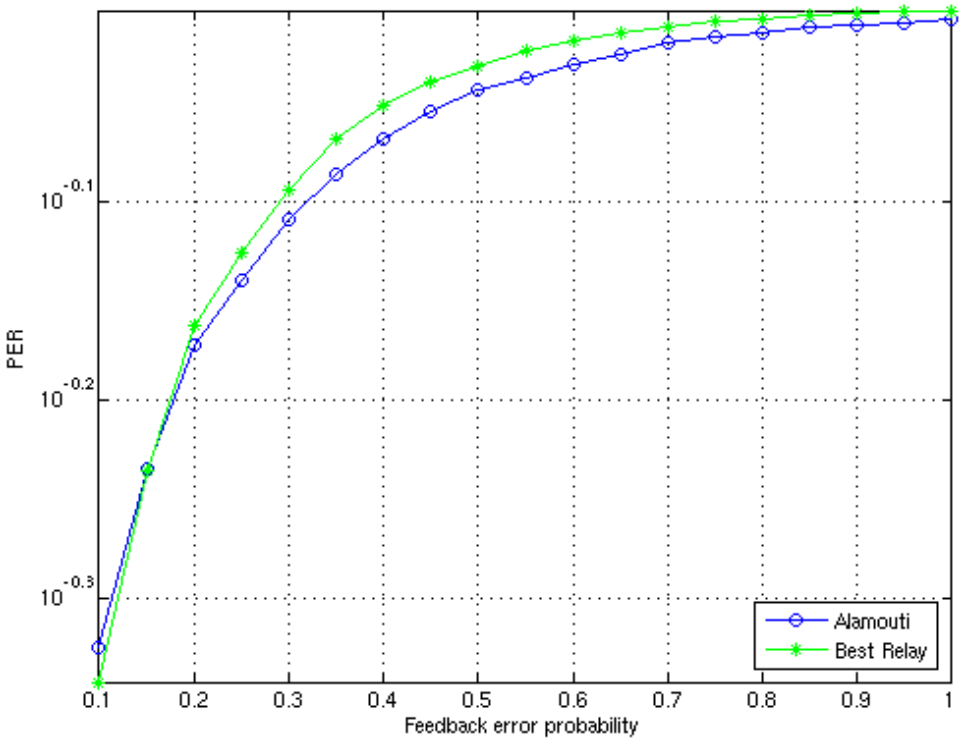

Figure 6.15. PER vs. feedback error probability. SNR=13 dB

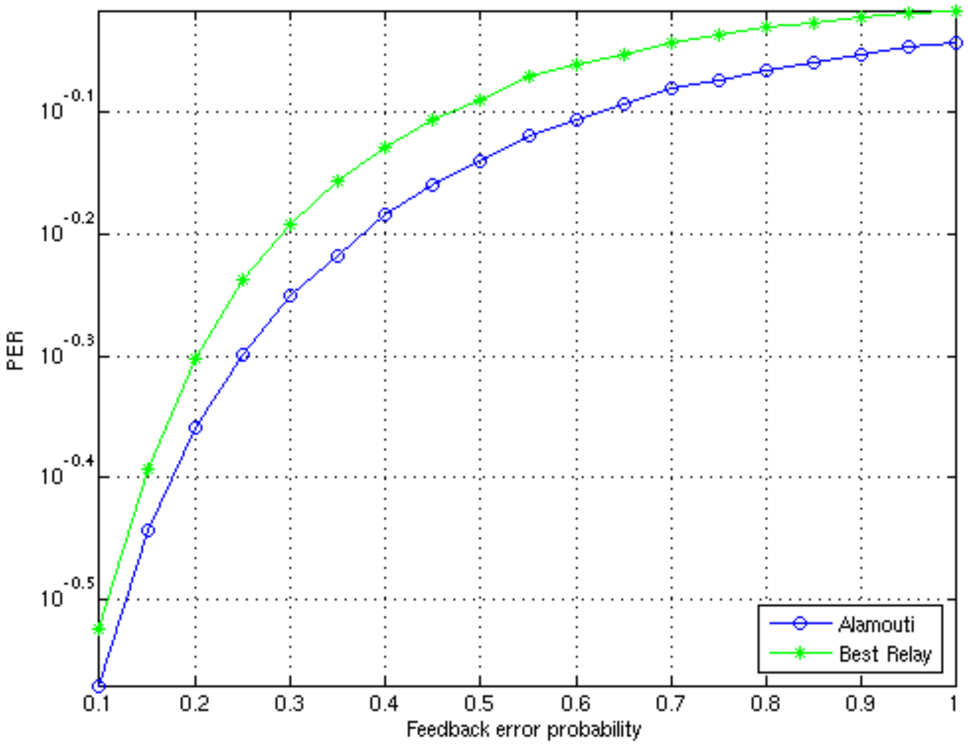

Figure 6.16. PER vs feedback error probability. SNR=15 dB

Using the points where the two curves of *figures 6.9 to 6.16* intersect we can establish a new selection method criterion:

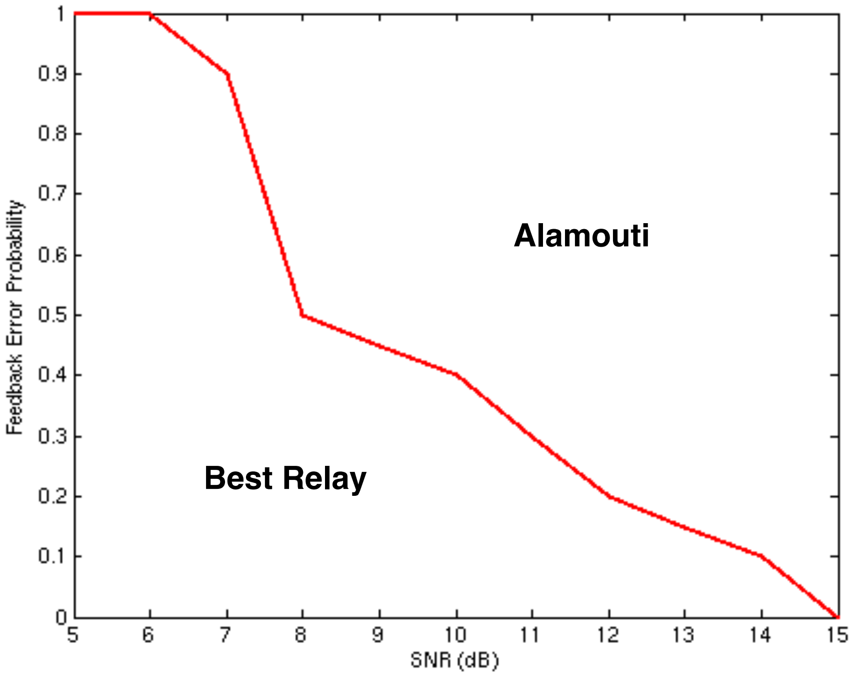

Figure 6.17. "Feedback Error Probability and Average SNR Aware Protocol" criterion

Figure 6.17 indicates when we have to choose the Alamouti and the Best Relay option. For lower SNR than 5 dB the Best Relay option is chosen, and for upper SNR than 15 dB the Alamouti option is chosen.

Applying this criterion it is possible to analyze the improvement of this protocol:

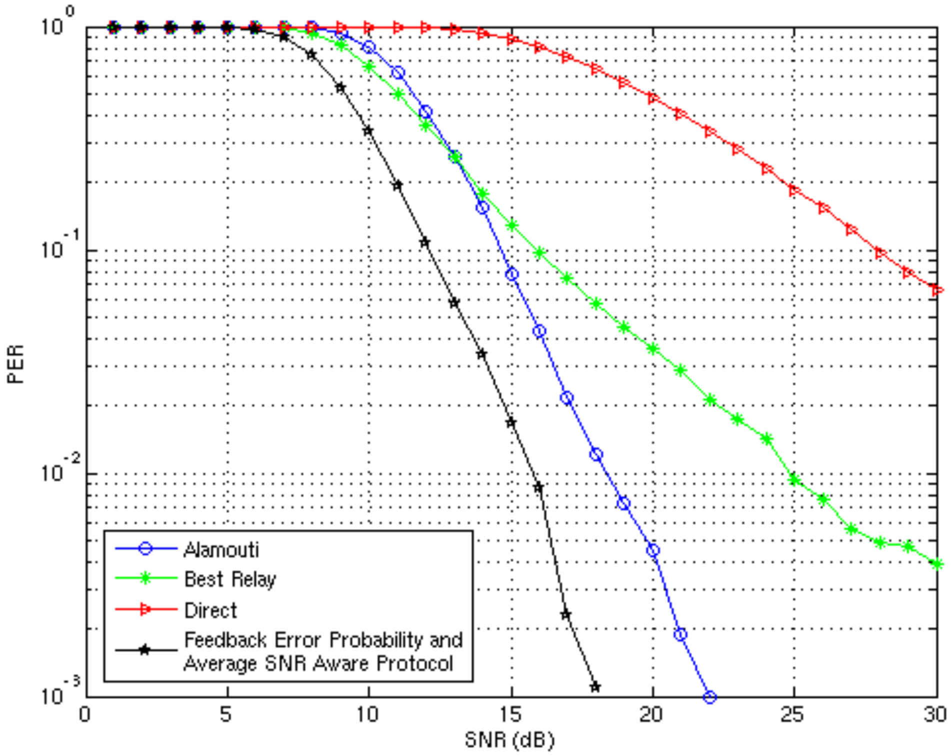

Figure 6.18. PER of the "Feedback Error Probability and Average SNR Aware Protocol"

Figure 6.18 shows that with this protocol we improves the system performance at low and high SNR around 3 dB when the target PER is 10^{-2}, and this gain grows when the SNR increases. Using this protocol we have a more accurate method selection than the "Feedback Error Probability Aware Protocol", so we get a better behavior. The destination must know the feedback error probability too.

6.4.3 Number of Relays and Average SNR Aware Protocol

Another possible criterion to choose the best option is to use the number of available relays in each moment and the average SNR.

Using the results obtained in *section 6.3*, it is possible to know which is the best cooperative method related to the number of relays (*figure 6.19*).

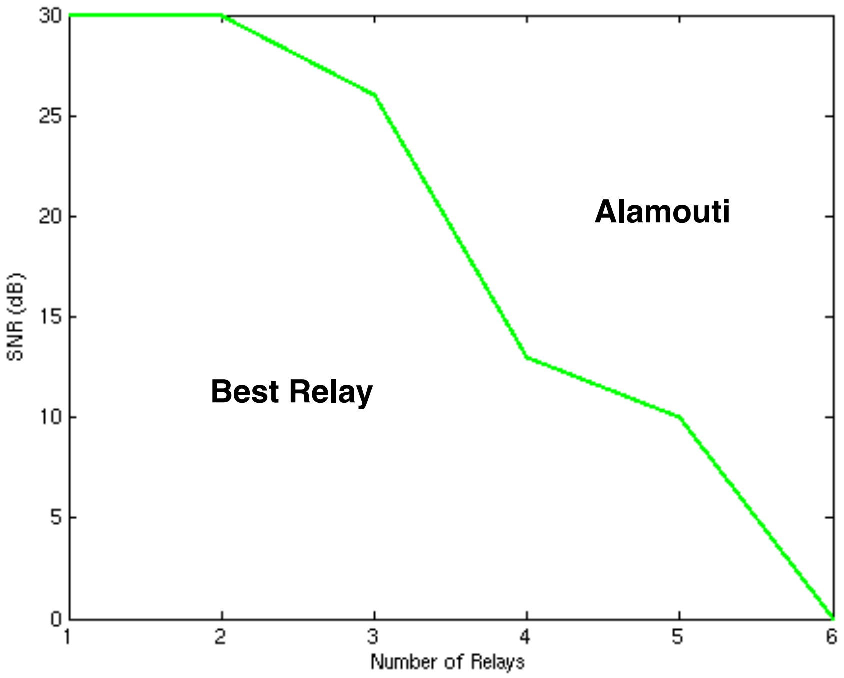

Figure 6.19. "Number of Relays and Average SNR Aware Protocol" criterion

With one relay, the destination always chooses the Best Relay method because the Alamouti option needs a minimum of two available relays. If more than 6 relays are in the decoding set, the destination chooses the Alamouti option because it gets a better performance in all the SNR range.

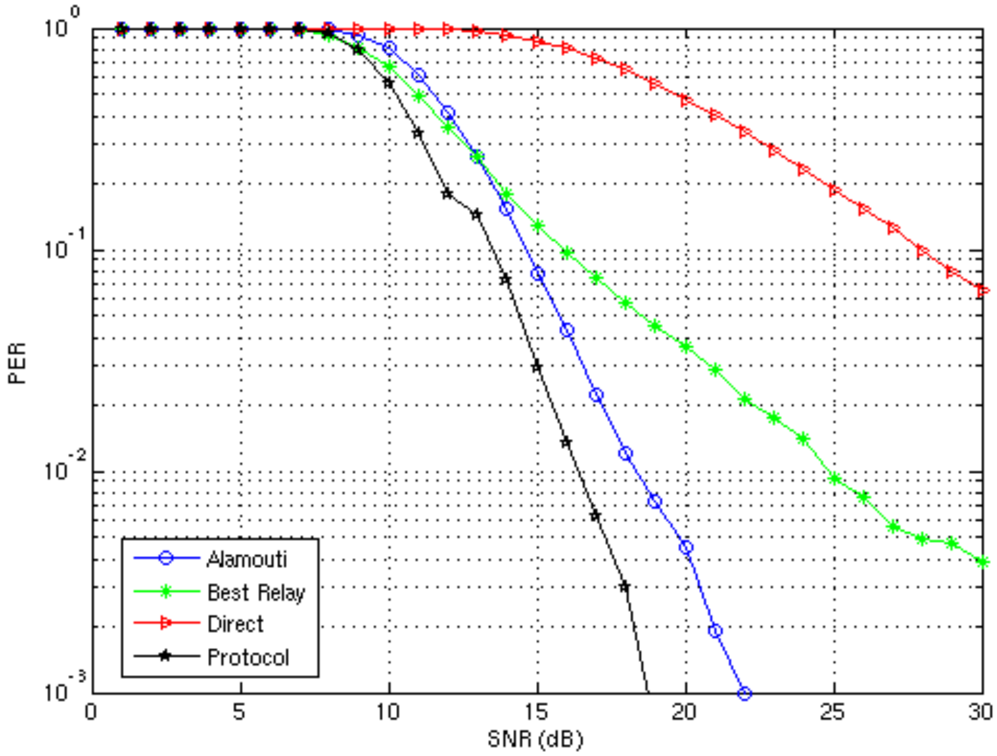

Figure 6.20. PER of the “Number of Relays and Average SNR Aware Protocol”

As it is shown in *figure 6.20*, with this protocol we does not get a result as good as the previous protocol but the destination only must know the number of available relays, and not a BER estimation, so it is easier to implement in a real scenario.

6.5 Power Allocation

Throughout the entire work we have analyzed two operation modes. The first one is to choose a single relay to transmit and the second one is to use two relays, so we have two power allocation options. Recalling that we attach Ps/2 to source-to-relay link and Ps/2 to relay-to-destination link, where Ps is the total power, it is easy to observe that in the first case we attach Ps/2 to the selected relay while in the second case we attach Ps/4 to the two selected relays.

As we have commented in *section 6.4*, to choose one or another option varies with the average SNR, generally choosing the Best Relay option for low SNR and

Alamouti option for medium and high SNR. This section makes a study on the possibility of making a power distribution more gradual and more precise.

In previous studies as [Wan06] complicated methods of power allocation have been used. In this work we propose an alternative method of power allocation based on the study of the PER for different power allocations, to use as a guideline for a new protocol.

For this study we define a new parameter β, allocating (1-β)*Ps/2 to the first selected relay (which has the best instantaneous SNR) and β*Ps/2 to the second selected relay.

From *sections 6.2* and *6.3*, we derive that the optimal SNR depends on the average SNR and the feedback error probability (with an ideal feedback link it is always better to choose only one relay).

The optimal value of the parameter β will be calculated separately for different average SNR, so the system chooses the best option to transmit depending on the value of the system average SNR.

The most interesting SNR interval is between 7 dB and 15 dB because in this zone the Alamouti option (β=0.5) has a worst performance than the Best Relay option (β=0), and it is interesting to study if an intermediate value of β is better.

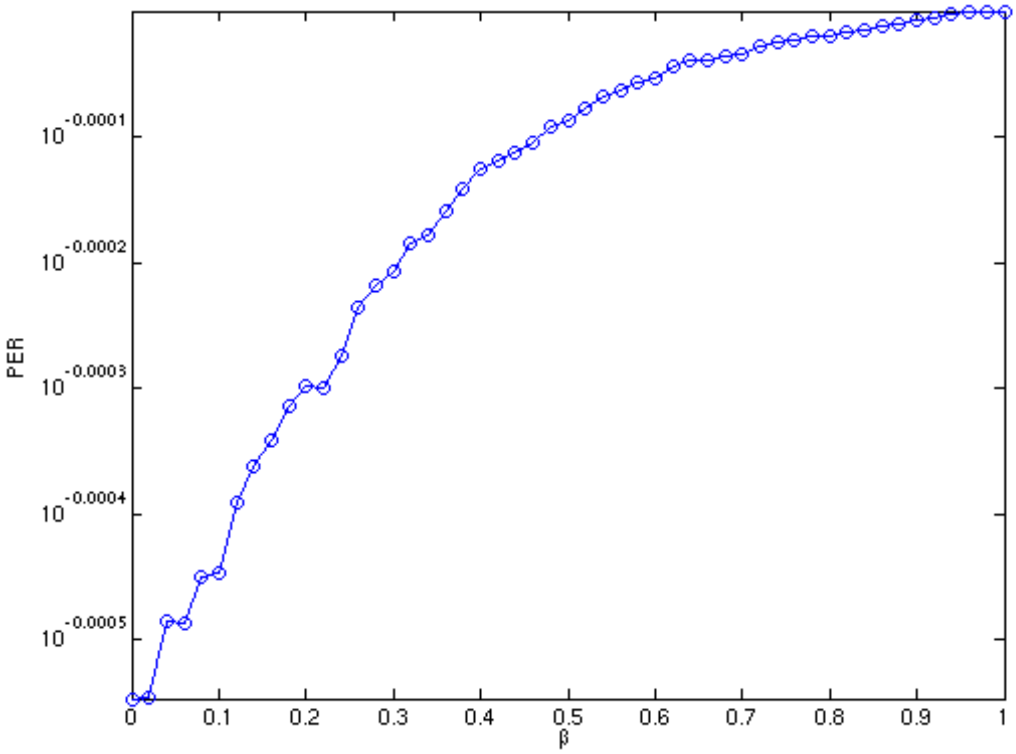

Figure 6.21. PER vs. β. SNR=7 dB

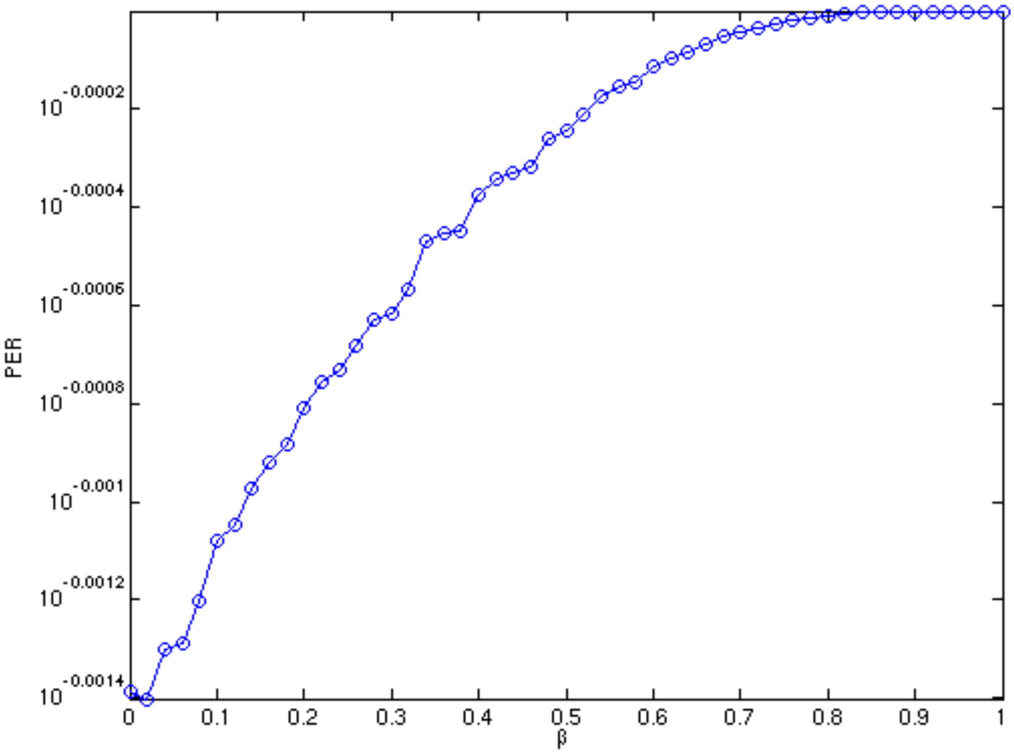

Figure 6.22. PER vs. β. SNR=8 dB

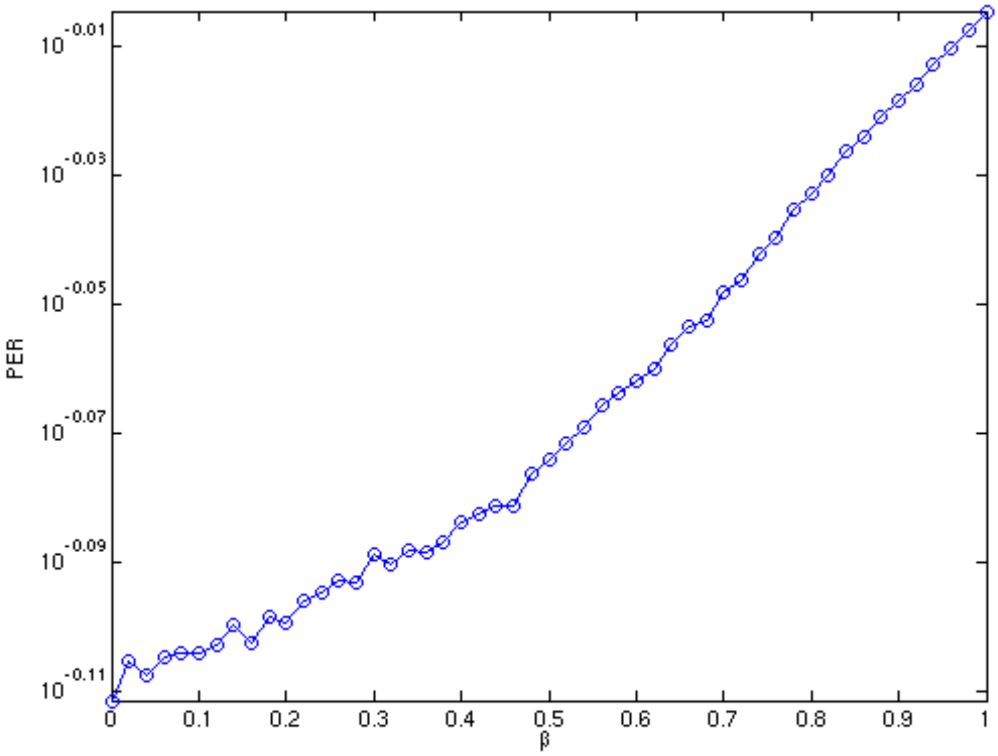

Figure 6.23. PER vs. β. SNR=10 dB

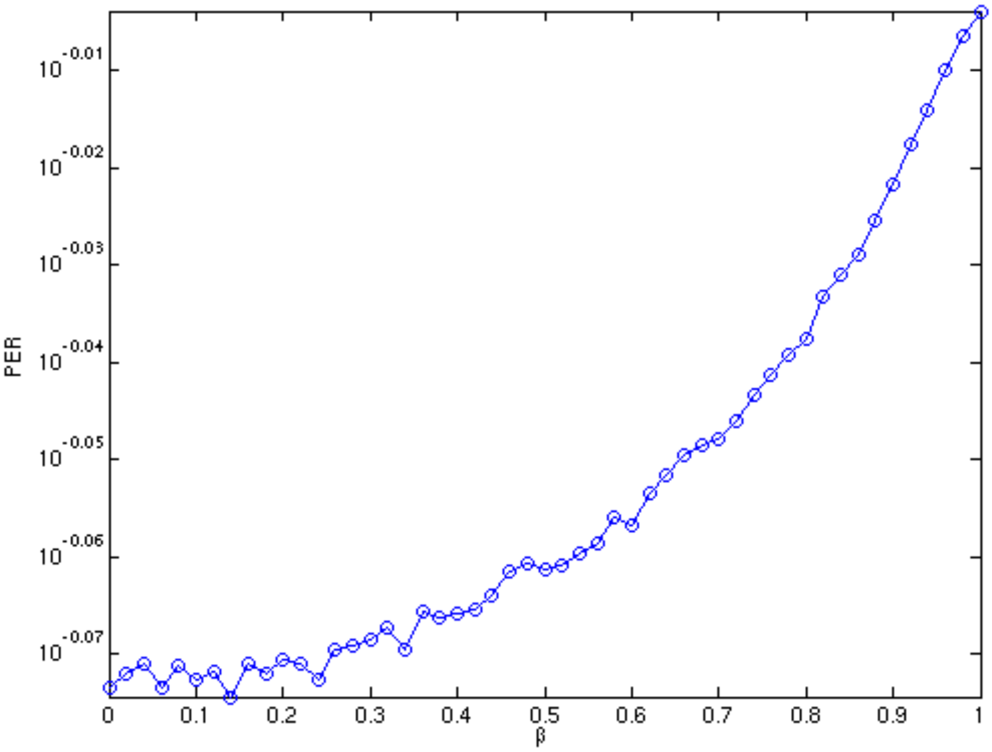

Figure 6.24. PER vs. β. SNR=11 dB

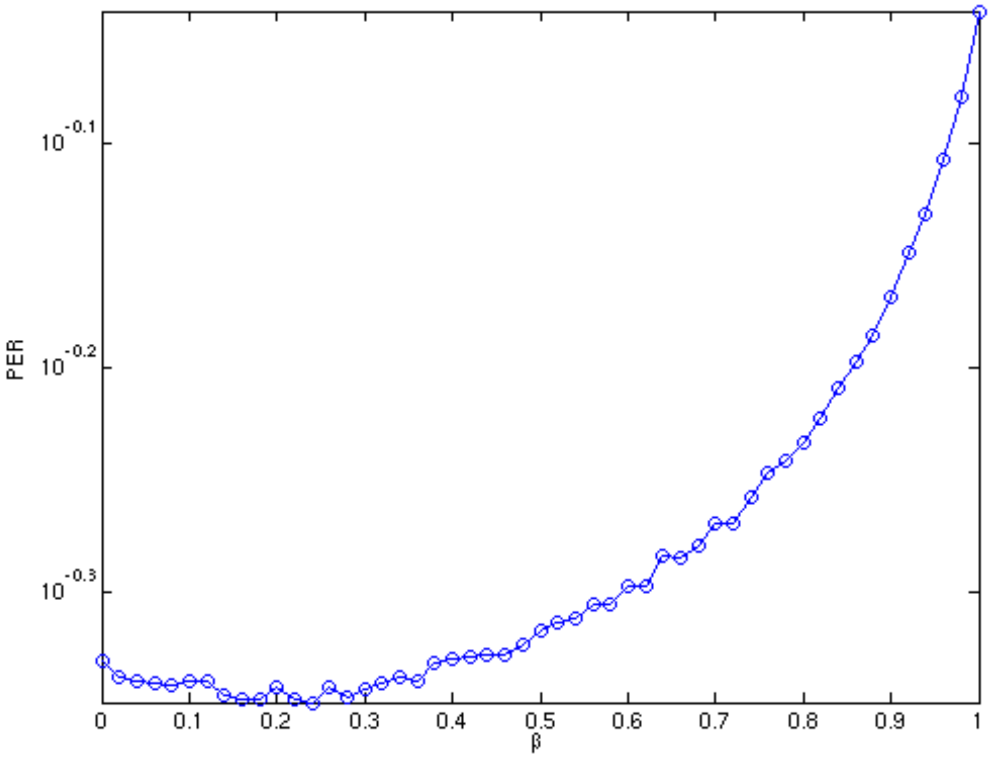

Figure 6.25. PER vs. β. SNR=12 dB

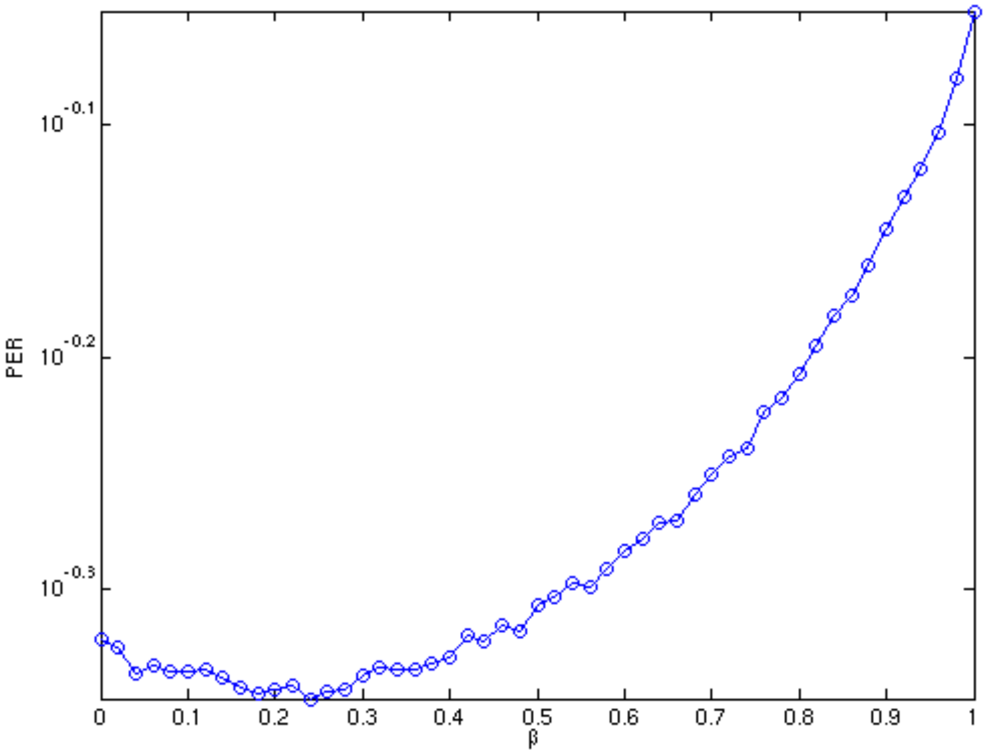

Figure 6.26. PER vs. β. SNR=13 dB

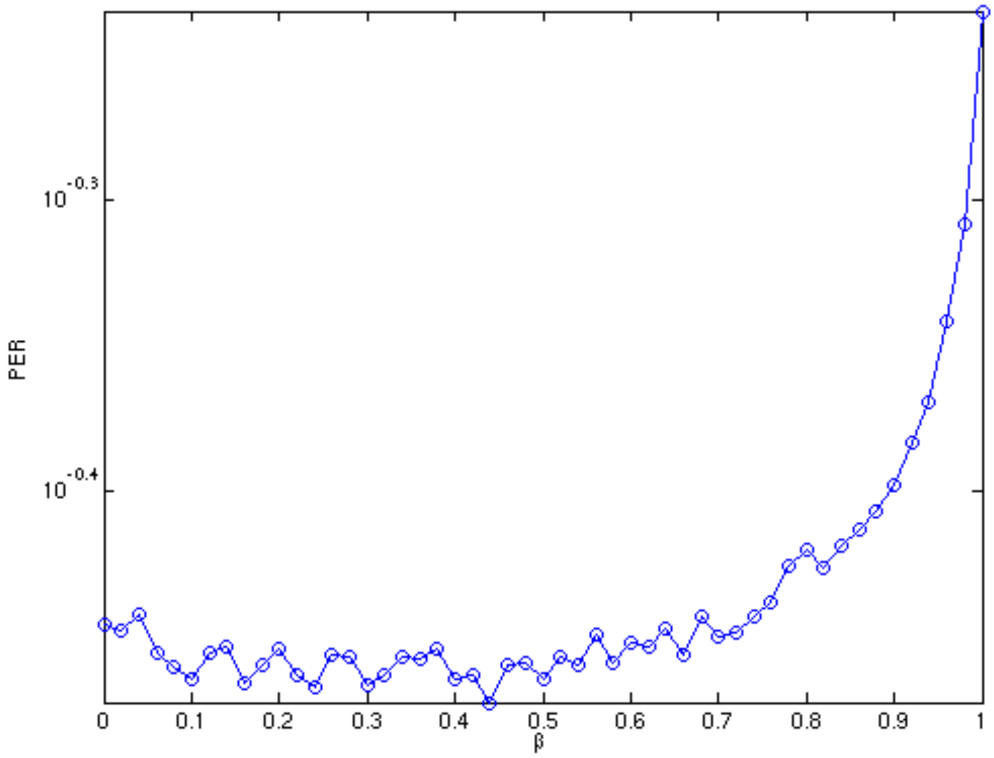

Figure 6.27. PER vs. β. SNR=14 dB

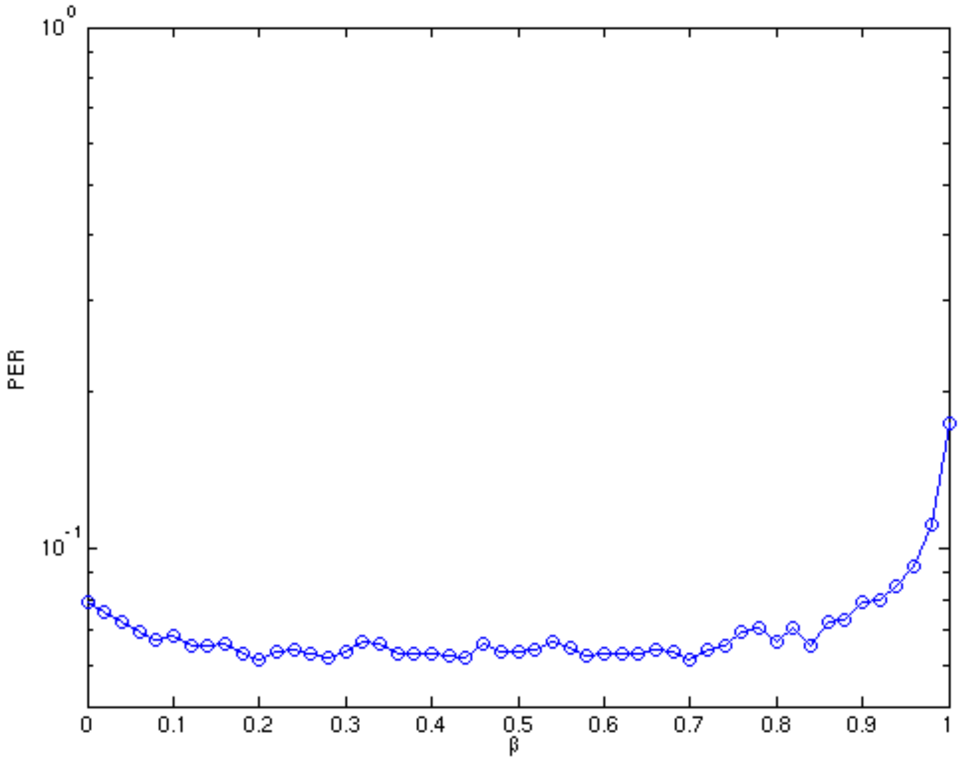

Figure 6.28. PER vs. β. SNR=15 dB

We choose the best value of the β parameter of the PER curves obtained in *figures 6.21* to *6.28*, Using this values we can get the criterion of *figure 6.29* to get the optimal β value depending on the SNR.

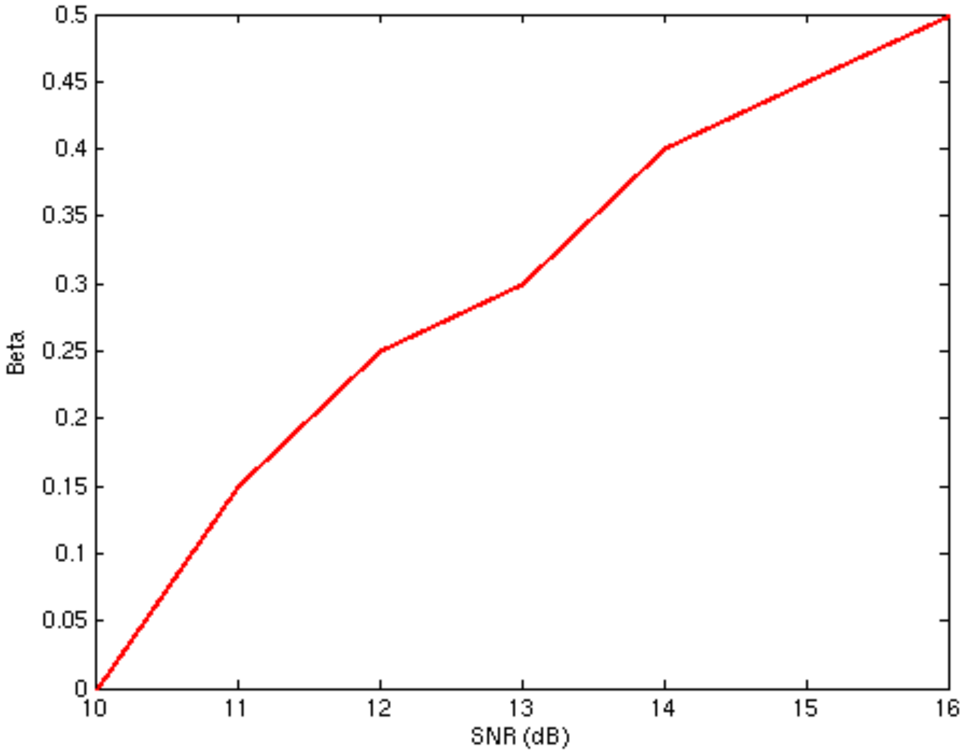

Figure 6.29. "Power Allocation Protocol" criterion

For lower SNR values than 10 dB the system must use a β=0, and for upper SNR values than 16 dB the system must use a β=0.5. Applying this criterion, a new protocol can be obtained:

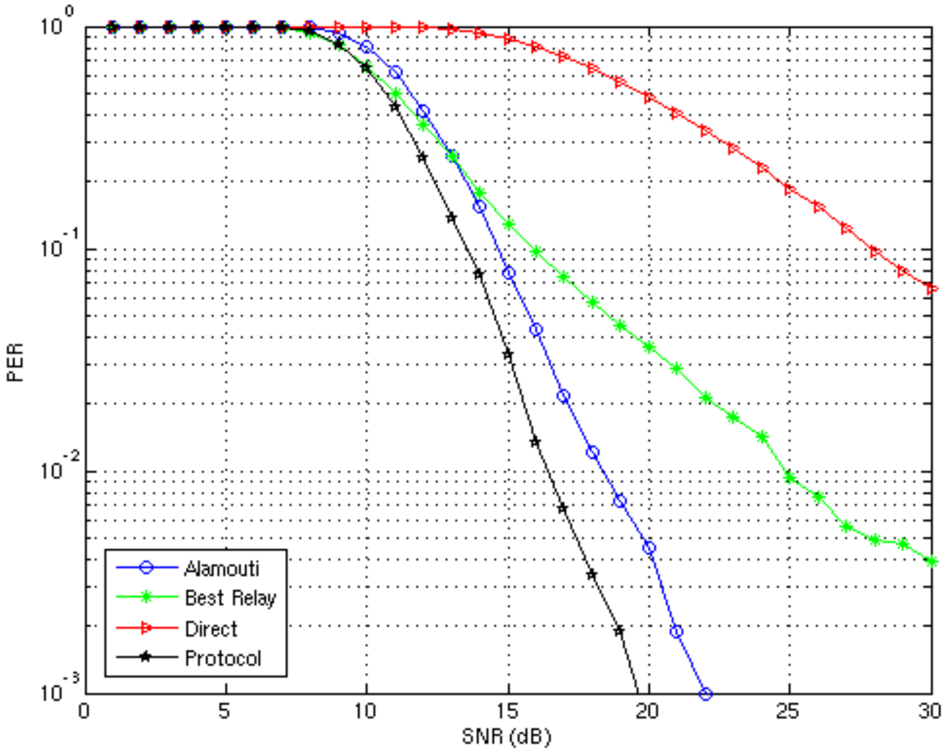

Figure 6.30. PER of the "Power Allocation Protocol"

Figure 6.30 shows a gain of 2 dB assuming a target PER of 10^{-2}. In our cooperative system only two relays can transmit simultaneously, but in a scenario where more relays could transmit at the same time, a power allocation protocol should be really useful. It is possible using a more complex STBC, but this demonstration does not fall within the objectives of this work.

6.6 Correlated Channel

Until now, independent temporally channels have been supposed where the channel state in a defined channel use is totally independent in front of the next channel use, in other words, the coherence time of the channel is less than the time slot.

This section introduces a certain correlation between different samples, using a temporary correlation coefficient $\rho_{K.}$ This parameter can take different values according to the channel model, under the assumption of a Jake's model, for instance, the correlation coefficient takes the value

$$\rho_k = J_0(2\pi \cdot f_{d,k} \cdot T_{D,k})$$

where $f_{d,k}$ is the Doppler frequency, $T_{D,k}$ is the sample time, and $J_0(\cdot)$ denotes the zero-order Bessel function of the first kind.

Consider two consecutive channel samples of the Gaussian process $h_{k,D}$ and $h'_{k,D}$. Then, $h_{k,D}$ conditioned on $h'_{k,D}$ follows a Gaussian distribution:

$$h_{k,D} \left| h'_{k,D} \sim CN(\rho_k h'_{k,D} \;\; ,(1-\rho_k{}^2)\sigma_{k,D}{}^2) \right.$$

and can be represented as in *figure 6.31*.

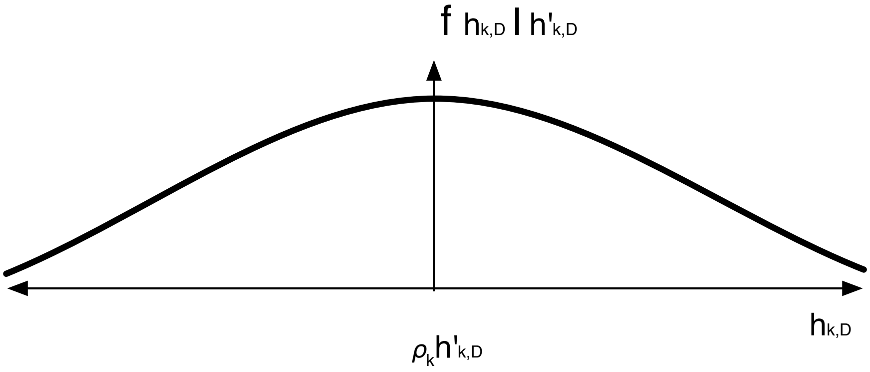

Figure 6.31. Channel correlation

To define the Bessel function parameters we take the WiMAX specifications choosing a frequency of 2.5 GHz and a sample time of 2 ms.

The Doppler frequency is defined as

$$f_D = \frac{v}{\lambda}$$

where v is the mobile speed and λ is the wavelength. A terminal speed of 1 m/s is assumed because it is approximately the humane average walking speed.

This model has been proposed because it allows us to solve or to reduce one of the biggest problems of relay selection. This problem is to choose which relay must transmit when the number of the selected relay exceeds the range of possible relays, which is easily detectable assuming that the relays know the total number of available relays. So if we have for instance 5 relays, and the destination chooses the relay number 7, they would know that an error has been occurred.

We propose a solution to this problem, which is the assumption that if we have a high correlation coefficient, the selected relays at the previous transmission could transmit correctly the message of the new transmission.

This strategy can be applied over the previous protocols to improve their performance in terms of PER.

First this “Correlated Channel Strategy” is applied over the “Number of Relays and Average SNR Aware Protocol”. As we have done before, it requires a brief study of the PER, using different number of relays. The process of selecting the protocol criterion has been commented in *section 6.4.3,* so only the simulation using 4 relays has been included in this work.

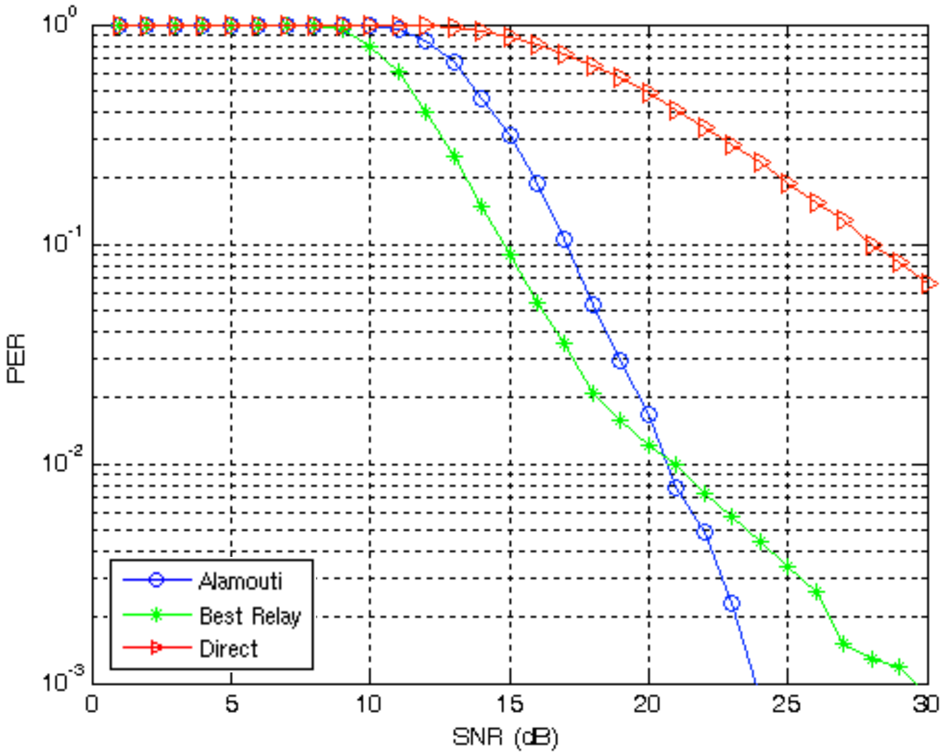

Figure 6.32. PER of the system using "Channel Correlation Strategy". 4 relays

From *figure 6.32* we can know the protocol criterion for the 4 relays case:

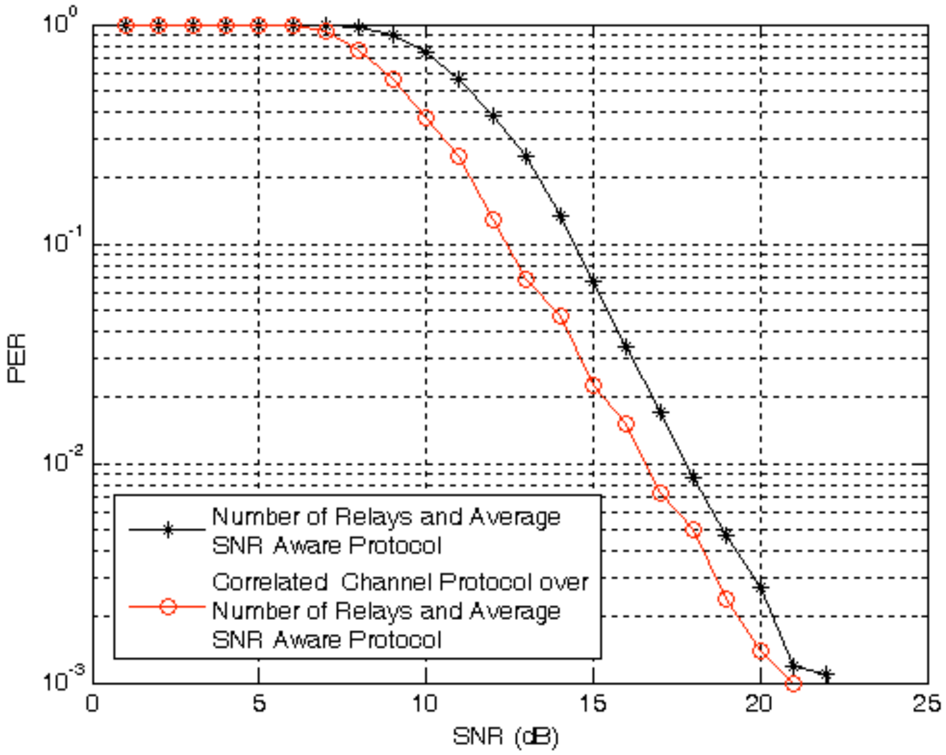

Figure 6.32. "Correlated Channel Protocol" over "Number of Relays and Average SNR Aware Protocol"

Figure 6.32 shows an improvement of approximately 2 dB with a target PER of 10^{-1} and 1 dB if a target PER of 10^{-2} is considered.

Now the "Correlated Channel Strategy" is applied over the "Feedback Error Probability and Average SNR Aware Protocol". As in the previous case, the study of the protocol criterion has been studied before (*section 6.4.2*), so the simulations have been attached to the annex and only the criterion has been included in this chapter (*figure 6.33*).

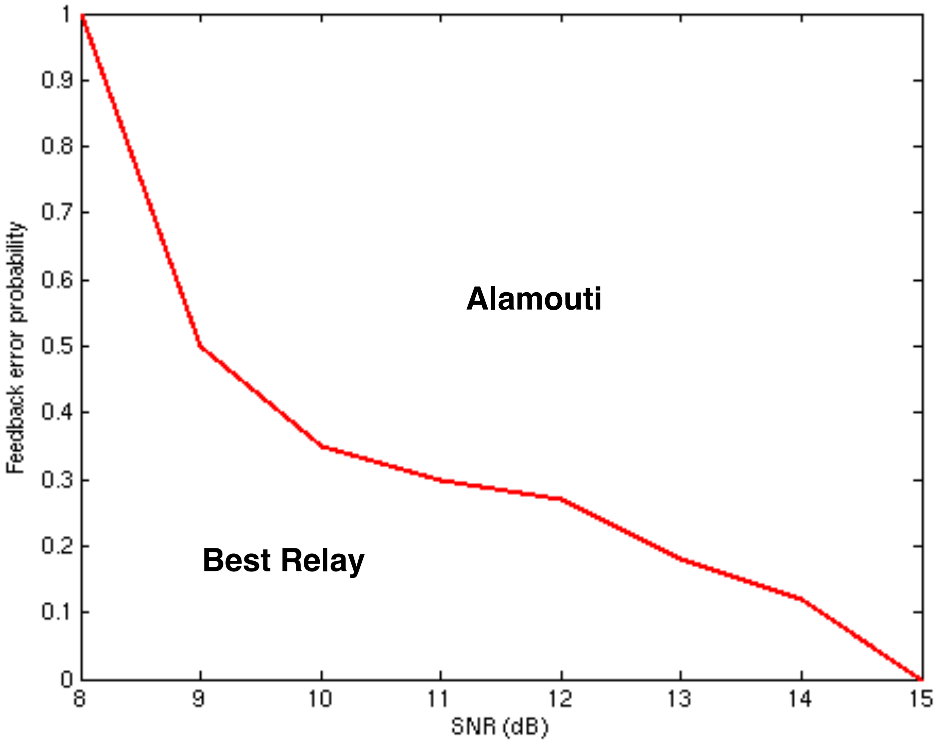

Figure 6.33. "Correlated Channel Protocol" over "Feedback Error Probability and Average SNR Protocol" criterion

Applying the criterion presented in *figure 6.33* over the "Feedback Error Probability and Average SNR Protocol", and comparing it with the performance without using this strategy:

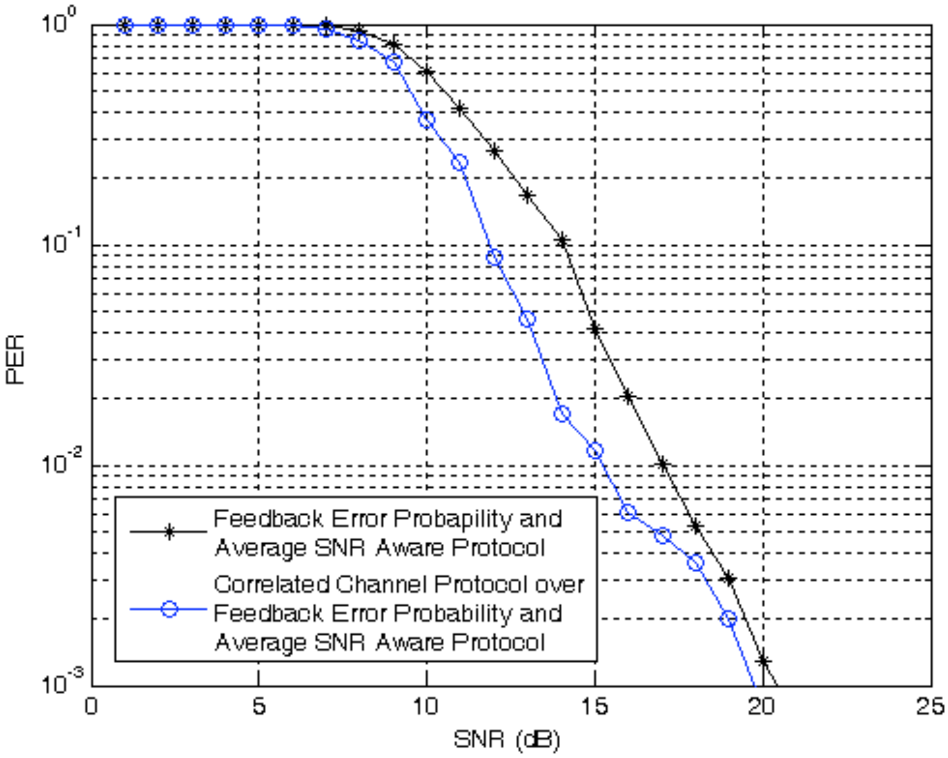

Figure 6.34. "Correlated Channel Protocol" over "Feedback Error Probability and Average SNR Aware Protocol"

Figure 6.34 shows that using the "Correlated Channel Strategy" over this protocol we gain about 2 dB extra with a target PER of 10^{-1} and about 1.5 dB with a target PER of 10^{-2}.

To use this strategy, the system only needs to storage the previous selected relays, so the "Correlated Channel Strategy" improves the system performance without any extra hardware implementation.

Using this strategy the difference between the two protocols becomes smaller when the SNR increases because the BER, and therefore the feedback error probability, decrement, so the wrong relay selection happens with a minor frequency.

Chapter 7

Conclusions and Future Work

7.1 Summary of the Work

This work has two very important parts, and the rest on the work is done to complement them. In the first one, we study a cooperative communications system from an information theory point of view, and in the second one, a real cooperative system is studied, and several cooperative protocols have been proposed.

First we have made an introduction to cooperative communications. Here we have explained what is it, and which are the necessary elements in them. A brief summary of the state of the art has been made after explaining the main advantages of this new strategy. To conclude the introduction, the main algorithms of cooperation have been studied. We have explained the space-time block codes, and the Alamouti code in particular.

Once presented the cooperative communications, we have defined the reference model used in this work, defining the channels, the power allocation and the duplex mode.

Before beginning the first of the two principal parts of the work, which uses the information theory, it has been necessary to do a brief introduction of the main concepts used directly or indirectly in this work of the information theory.

Subsequently we have begun the two main chapters of the work. In the first one we have established the basis of the system, described step by step as we have done.

In the second main chapter we have defined an entire real system. First, we have defined a new model based on the previous one, to study the real performance of the cooperative system. Then we have proposed several protocols aimed at serving as a guide for the relay selection methods of a real system. The first three protocols are based on the previous results and how would be the best combination of Direct, Best Relay and Alamouti options. Subsequently we have proposed two more protocols based on additional studies changing the previous system model. In the first one, we have change the power allocation, and in the second one, a correlation parameter has been introduced to the system model.

7.2 Conclusions

In this work we have justified the use of cooperation techniques in wireless communications.

After introducing the foundations of our study, by introducing the cooperative systems, the used model and the information theory, we have begun with the most important part of the work, *chapter 5* and *chapter 6*.

From the first moment we have presented two cooperation techniques, the first one uses the simple method of choosing the best relay to transmit the information to the destination and the second one uses the Alamouti codification. Throughout the work we have compared these two techniques with a non cooperative system. In this comparison, it can be observed with the performance curves and analyzed using the theoretical demonstration, that the Alamouti method gets better results when the feedback link is not ideal.

In *chapter 5* we have studied the impact of different relay locations, first using different geometries and then positioning the relays randomly, which is closer to the a real system that we want to study where the relays can move freely. For the Alamouti system, we have studied the possibility of choosing the two closest relays to the destination and the possibility of choosing the two relays with better SNR. Observing that the latter method get a better results, we have continued with this Alamouti mode. One of the most important part of this chapter has been the introduction of the feedback error probability. Here we have observed that the Alamouti option is more robust than the Best Relay option when a non-ideal feedback link is assumed.

In *chapter 6* a real system has been studied, and the Best Relay, the Alamouti and the Direct option have been compared in terms of PER. First we have analyzed the performance of the system without feedback error and it has subsequently been included. The most important part of this chapter is the introduction of several protocols. The performance and the difficult of implementing them have been studied. We have proved that good results can be obtained by studying PER vs. feedback error probability and by applying these results to the relay selection protocol. We have also established a new power control method, dividing the available power between two relays, using a new criterion. We have obtained good results using this protocol, but this method could be more useful when a more complex STBC is used, and more relays could transmit at the same time.

Finally we have introduced a variation on the previous system model by introducing a correlation factor between two consecutive samples in order to improve the system performance when the destination chooses a non existent relay. With this variation, we have improved the system behavior of the previous protocols and a future research line has been established.

The proposed protocols in this work are summarized in the Table 7.1.

	Key Parameter	**Approximated Gain***	**Implementation**
Feedback Error Probability Aware Protocol	Feedback error probability	1.5 dB	Destination Needs to know the Feedback error probability estimation using the pilot signal
Feedback Error Probability and Average SNR Aware Protocol	Feedback error probability and average SNR	3 dB	Destination Needs to know the Feedback error probability estimation using the pilot signal
Number of Relays and Average SNR Aware Protocol	Number of available relays and average SNR	2 dB	Destination needs to know the number of available relays
Power Allocation Protocol	β	2 dB	Destination must be able to send the necessary power information to the relays, and they must adjust their transmission power
Channel Correlation Protocol	ρ_k	1 dB-1.5 dB extra	Destination must storage the selected relays

Table 7.1. Summary of the proposed protocols

*Gain calculated using 4 relays and a target PER of 10^{-2}.

7.3 Future Work

While many key results for cooperative communications have already been obtained, there are many more issues that remain to be addressed.

An important question is how partners nodes are assigned and managed in multi-users networks, in other words, how is it determined which users cooperate with each other, and how often partners are reassigned. Systems such as cellular, in which the users communicate with a central base station, offer possibility of a centralized mechanism. Assuming that the base station has some knowledge of the all the channels between users, partners could be assigned to optimize a given performance criterion, such as the average PER for all users in the network. In contrast, systems such as ad hoc networks does not have centralized control so they require a distribute cooperative control, in which users are able to independently decide who is a relay at any given time.

Other important issues can be treated in the future with the aim of extending this work as:

- To try other STBC to get more spatial diversity.
- To adapt the “Channel Correlation Protocol” to other strategies.
- To include transmission delays to the model to get a more realistic results.
- To particularize this general study to another particular technology as mobile networks,WI-FI networks...

The challenge here is to develop a scheme that treats all users fairly, does not require significant additional system resources, and can be implemented easily.

Bibliography

[Ala98] S.M. Alamouti, “A simple transmit diversity technique for wireless communications”, IEEE J. Sel. Ar. Commun., 1998.

[Car82] A. B. Carleial, “Multiple-access channels with different generalized feedback signals,” IEEE Trans. Inform. Theory, vol. IT-28, pp. 841–850, Nov. 1982.

[Cov79] T.M Cover and A. A. E. Gamal, “Capacity Theorems for the Relay Channel”, IEEE Trans. Info. Theory, vol.25, no. 5, Sept. 1979, pp. 572-84.

[Gas02] M. Gastpar and M. Vetterli, “On the capacity of wireless networks:The relay case,” in Proc. IEEE INFOCOM, New York, June 2002.

[Gol05] A. Goldsmith, “Wireless Communications”, Cambridge University Press, 2005.

[Gup01] P. Gupta and P. R. Kumar, “Toward and information theory of large networks: An achievable rate region,” in Proc. IEEE Int. Symp. Information Theory (ISIT), Washington, DC, June 2001, p. 150.

[Jin06] Y. Jing and B. Hassibi, “Distributed space-time coding in wireless relay networks," IEEE Trans. Wireless Commun., vol. 5, pp. 3524-3536, Dec. 2006.

[Kin78] R. C. King, “Multiple access channels with generalized feedback,” Ph.D. dissertation, Stanford Univ., Palo Alto, CA, 1978.

[Kra00] G. Kramer and A. J. van Wijngaarden, “On the white Gaussian multiple-access relay channel,” in Proc. IEEE Int. Symp. Information Theory (ISIT), Sorrento, Italy, June 2000, p. 40.

[Lan01] J.N. Laneman, G.W. Wornell, and D.N.C. Tse, “An efficient protocol for realizing cooperative diversity in wireless networks,” Proc. IEEE ISIT, Washington, DC, June 2001, p.220.

[Lin05] J. C. H. Lin and A. Stefanov, “Coded cooperation for OFDM systems," in. Proc. IEEE WirelessCom, Maui, Hi, June 2005.

[Rez02] A. Reznik, S. Kulkarni, and S. Verdú, “Capacity and optimal resource allocation in the degraded Gaussian relay channel with multiple relays,” in Proc. Allerton Conf. Communications, Control, and Computing, Monticello, IL, Oct. 2002.

[Rom92] S. Roman. “Coding and information theory”. New York [etc.]: Springer Verlag, cop. XVII, 486 p. : il., gràf.; 24 cm. Graduate texts in mathematics ; 134. 1992

[Sch00] B. Schein and R. G. Gallager, “The Gaussian parallel relay network,” in Proc. IEEE Int. Symp. Information Theory (ISIT), Sorrento, Italy, June 2000, p. 22.

[Sch01] B. Schein, “Distributed coordination in network information theory,” Ph.D. dissertation, MIT, Cambridge, MA, 2001.

[Scu05] G. Scutari and S. Barbarossa, “Distributed space-time coding for regenerative relay networks,” IEEE Trans. Wireless Commun., vol. 4, pp. 2387-2399, Sep. 2005.

[Sen98] A. Sendonaris, E. Erkip, and B. Aazhang, “Increasing uplink capacity via user cooperation diversity,” in Proc. IEEE Int. Symp. Information Theory (ISIT), Cambridge, MA, Aug. 1998, p. 156.

[Tar99] V. Tarokh, H. Jafarkhani, and A. Calderbank, “Space-time block codes from orthogonal designs,” IEEE Trans. Inform. Theory, vol. 45, no. 5, pp. 1456–1467, July 1999.

[Vic08] J. L. Vicario, A. Bel, J.A. López-Salcedo, G. Seco-Granados, “Opportunistic Relay Selection with Outdated CSI: Outage Probability and Diversity Analysis”, submitted to IEEE SPAWC 2008.

[Zha05] B. Zhao and M.C. Valenti, “Practical Relay Networks: A Generalization of Hybrid-ARQ”, IEEE Journ. on Sel. Areas in Comm., Vol. 23, No. 1,Jan 2005

[Wan06] T. Wang, R. Wang, and G. B. Giannakis, “`Smart Regenerative Relays for Link-Adaptive Cooperative Communications”, Proc. of 40th Conf. on Info. Sciences and Systems, Princeton Univ., NJ, March 22-24, 2006.

Appendix

Ap1. Correlated Channel simulations

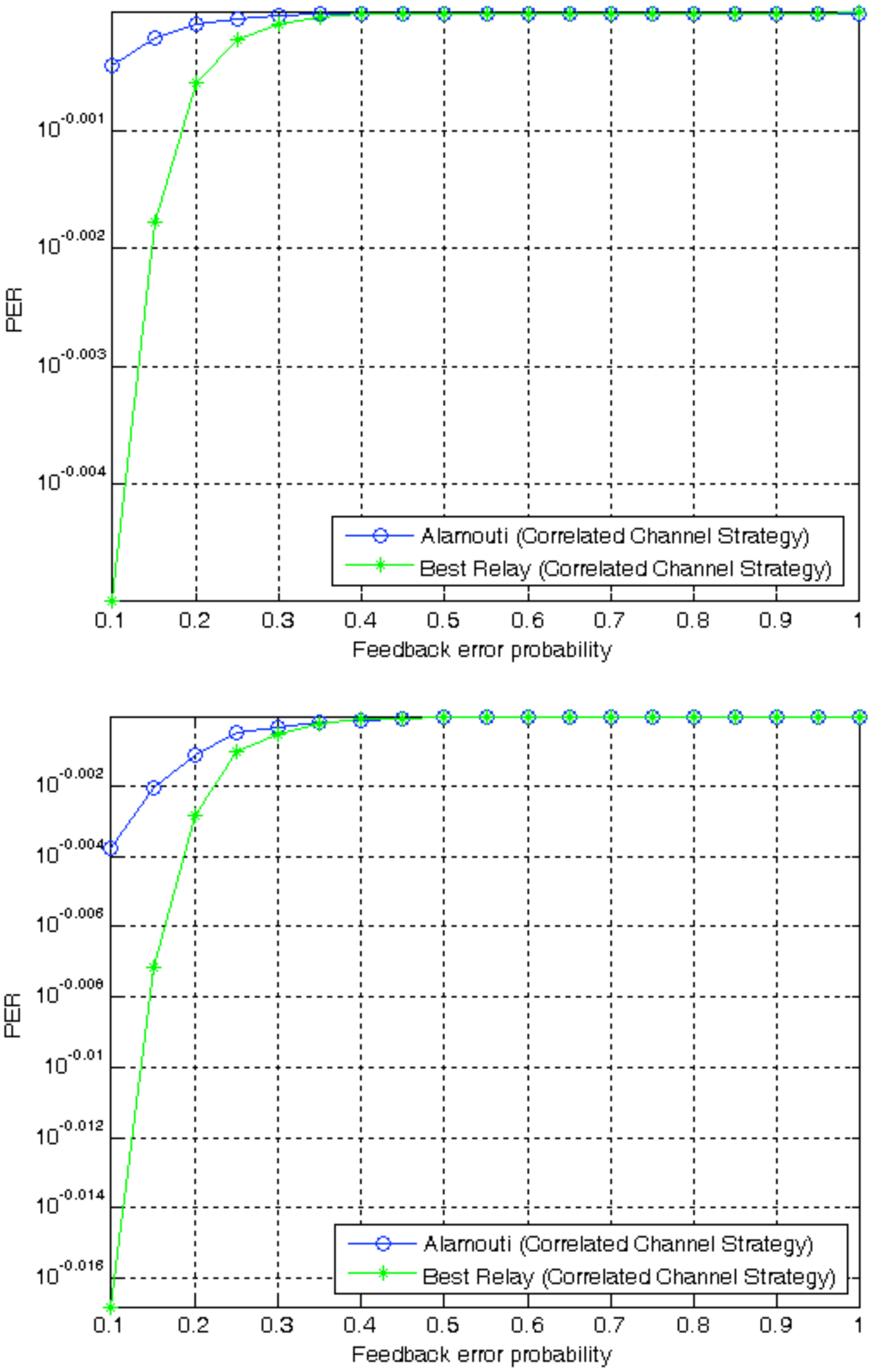

Figure Ap.1 PER vs. feedback error probability. SNR=8 dB

Figure Ap.2 PER vs. feedback error probability. SNR=9 dB

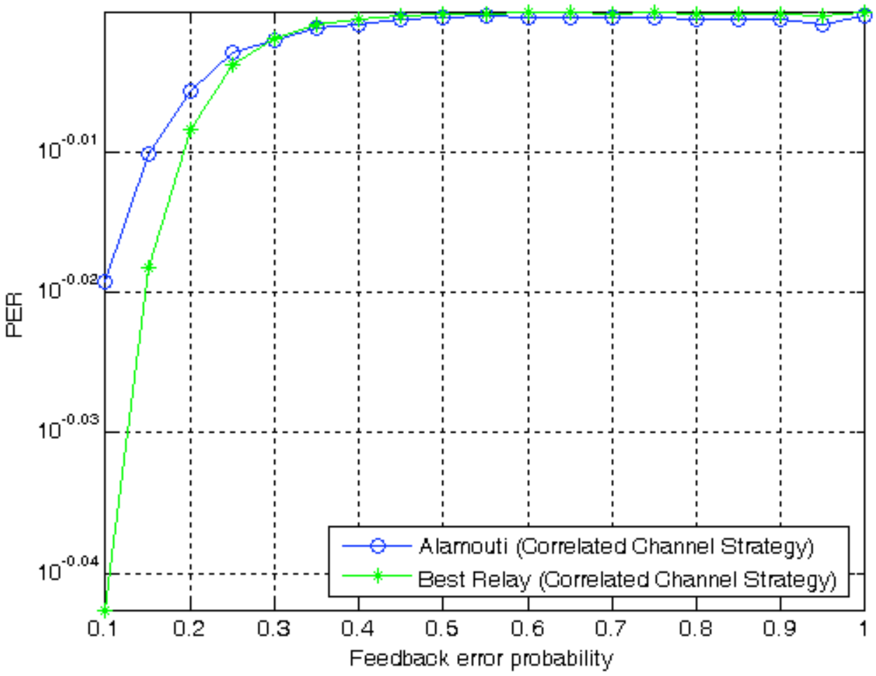

Figure Ap.3 PER vs. feedback error probability. SNR=10 dB

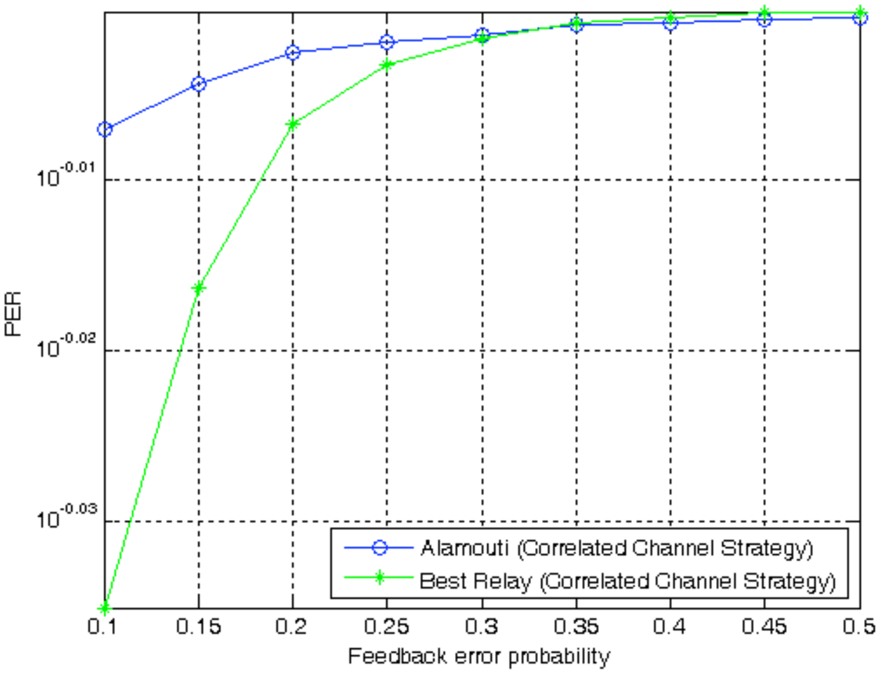

Figure Ap.4 PER vs. feedback error probability. SNR=11 dB

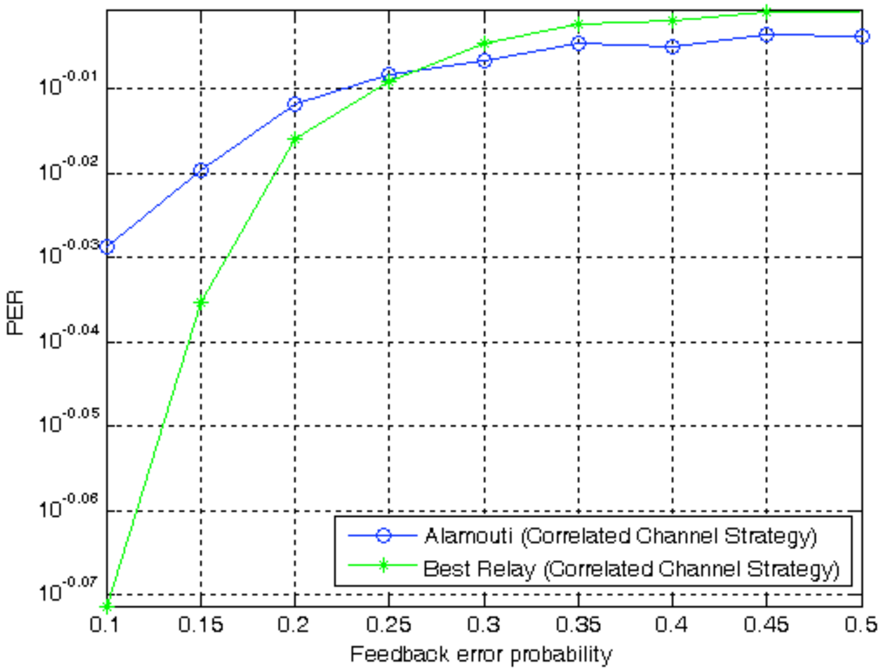

Figure Ap.5 PER vs. feedback error probability. SNR=12 dB

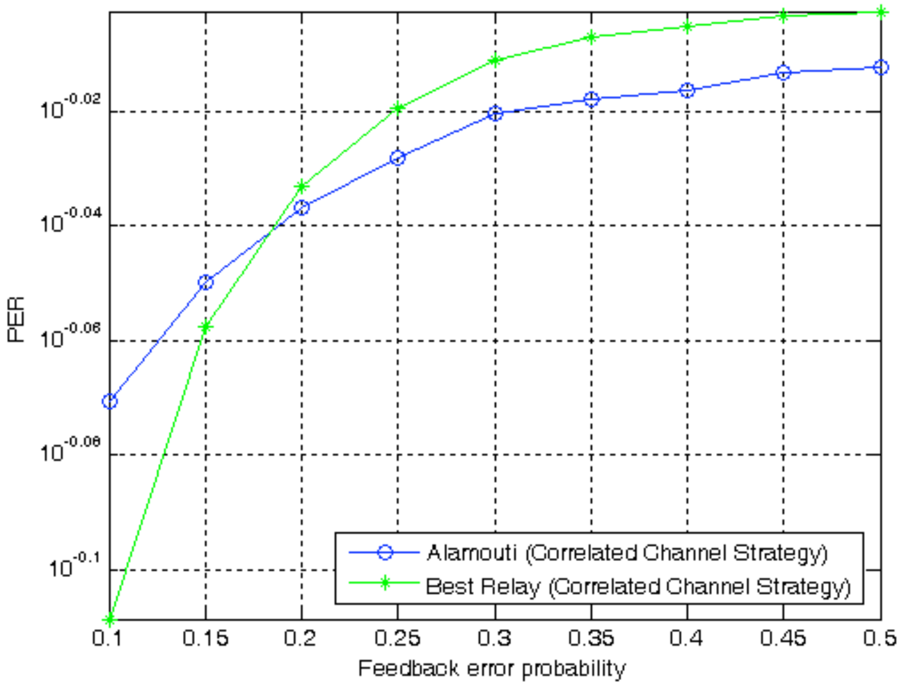

Figure Ap.6 PER vs. feedback error probability. SNR=13 dB

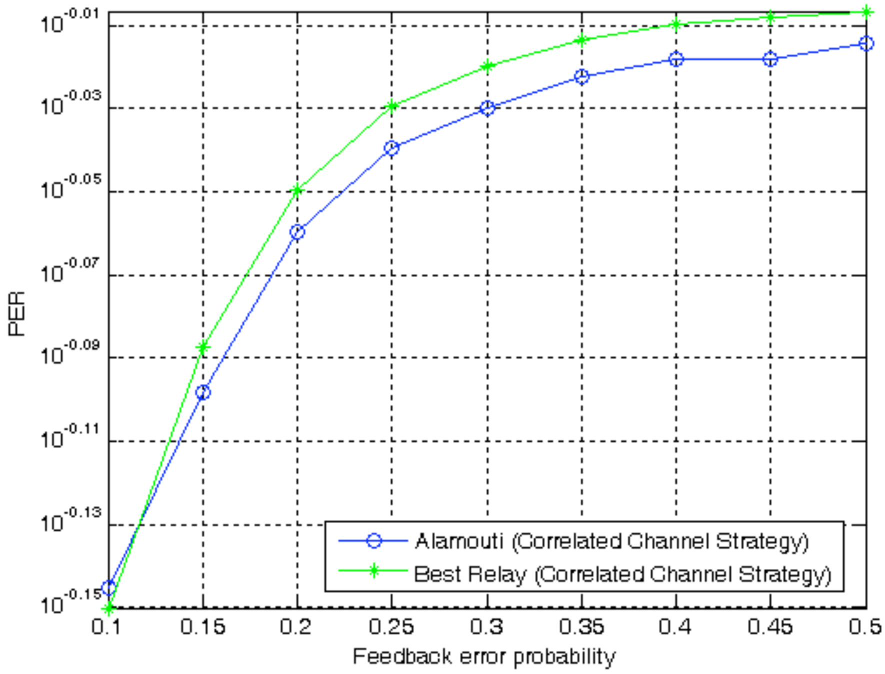

Figure Ap.7 PER vs. feedback error probability. SNR=14 dB

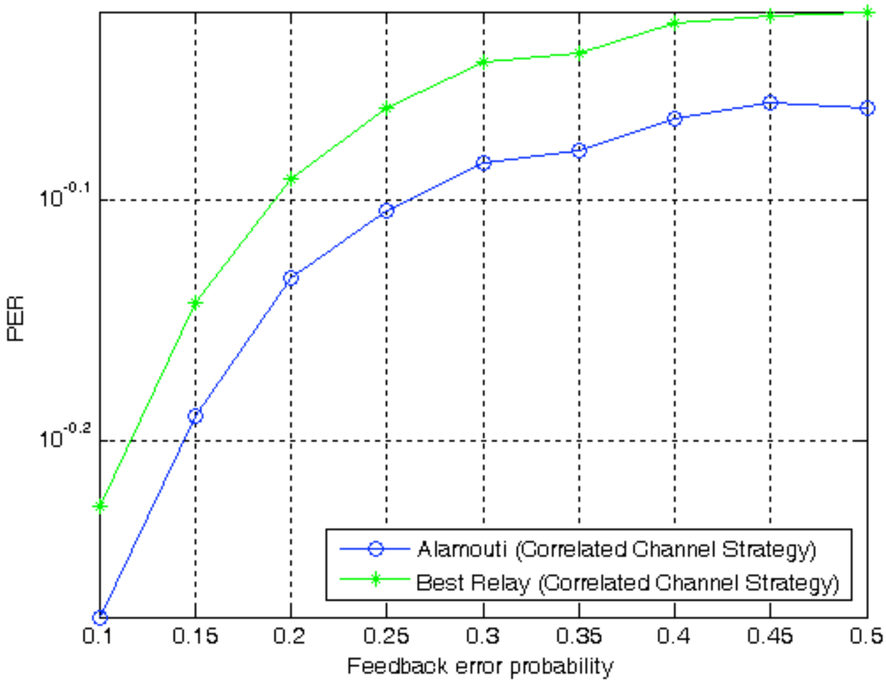

Figure Ap.8 PER vs. feedback error probability. SNR=15 dB

Ap2. Matlab Code

Due to the extension of the Matlab Code used along this work, in this appendix, it is only included some of the used programs. The complet Matlab code is included in the CD.

Ap2.1 Model from an Information theory point of view

```
clear all;

I=100000;
N=10;
R=2;
d=100;

distanciax=d*rand(N,1);
distanciay=(d/2)*rand(N,1);

distancia=sqrt(distanciax.^2+distanciay.^2);

distanciax2=d-distanciax;
distancia2=sqrt(distanciax2.^2+distanciay.^2);


fc=2.5*10^9;
c=3*10^8;
landac=c/fc;
d0=1;
mu=3;
Pn=1;
Probe=0.3;
L=10;

for snrm=1:30
    canales_outage_Alamouti=0;
    rate_medio_Alamouti=0;
    rateslot=R;
    snrml=10^(snrm/10);
    Nslots=0;

    varianza1(snrm)=((d/d0)^(-mu))*(landac/(4*pi*d0))^2;
    Ps=snrml*Pn/varianza1(snrm);

    for aux=1:I

        for cont=1:N

            varianza2=((distancia(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;

r(cont1)=sqrt(0.5)*sqrt(varianza2)*(randn(1,1)+j*randn(1,1));

        end
```

```
        for cont=1:N

            varianza3=((distancia2(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;

h(cont1)=sqrt(0.5)*sqrt(varianza3)*(randn(1,1)+j*randn(1,1));

        end

        Nslots=Nslots+1;
        snrr1=(abs(r).^2)*Ps/(Pn*2);
        vect_outage=snrr1>=(2^(2*R)-1);

        if(sum(vect_outage)>=2)

            vect_sec_link=abs(h).^2.*vect_outage;
            [maxim,pos1]=max(vect_sec_link);         %Cogemos el
m‡ximo
            vect_sec_link2(1)=vect_sec_link(pos1);   %Lo guardamos
            vect_sec_link(pos1)=0;                   %Ponemos su
posicion a 0
            [maxim2,pos2]=max(vect_sec_link);       %Cogemos el
siguiente maximo
            vect_sec_link(pos1)=vect_sec_link2(1);   %Volvemos a
poner el valor m‡ximo en su posicion

            pos1=pos1-1; %Para que empiezen las posiciones en 0
            pos2=pos2-1;

            b1=dec2bin(pos1); %Pasamos las posiciones a binario
            b2=dec2bin(pos2);

           a=rand(1);
           b=rand(1);
           c=rand(1);
           dd=rand(1);

           l=0;
             l(4)=b1(length(b1));

           if pos1>1
               l(3)=b1(length(b1)-1);
           end

           if pos1>3
               l(2)=b1(length(b1)-2);
           end

           if pos1>7
               l(1)=b1(length(b1)-3);
           end

           l=l>'0';
           if a<=Probe
               l(4)=not(l(4));
           end

           if b<=Probe && pos1>1
               l(3)=not(l(3));
           end

           if c<=Probe && pos1>3
```

```
    l(2)=not(l(2));
end

if dd<=Probe && pos1>7
    l(1)=not(l(1));
end

%Convertimos binario a decimal

eleccion1=l(4)+2*l(3)+4*l(2)+8*l(1);
a=rand(1);
b=rand(1);
c=rand(1);
dd=rand(1);

l=0;

l(4)=b2(length(b2));

if pos2>1
    l(3)=b2(length(b2)-1);
end

if pos2>3
    l(2)=b2(length(b2)-2);
end

if pos2>7
    l(1)=b2(length(b2)-3);
end

l=l>'0';

if a<=Probe
    l(4)=not(l(4));

end

if b<=Probe && pos1>1
    l(3)=not(l(3));
end

if c<=Probe && pos1>3
    l(2)=not(l(2));
end

if dd<=Probe && pos1>7
    l(1)=not(l(1));
end


%Convertimos binario a decimal

eleccion2=l(4)+2*l(3)+4*l(2)+8*l(1);
eleccion1=eleccion1+1;
eleccion2=eleccion2+1;

if N<eleccion1
    vect_sec_link(eleccion1)=0;
end

if N<eleccion2
    vect_sec_link(eleccion2)=0;
```

```
           end

            snrr2=(Ps/(Pn*4))
*(vect_sec_link(eleccion1)+vect_sec_link(eleccion2));

            canales_outage_Alamouti=canales_outage_Alamouti+
(snrr2<(2^(2*R)-1));

            if snrr2>=(2^(2*R)-1)
                rate_medio_Alamouti=rate_medio_Alamouti+rateslot;
            else
                rate_medio_Alamouti=rate_medio_Alamouti+0;
            end

            else

            canales_outage_Alamouti=canales_outage_Alamouti+1;
            rate_medio_Alamouti=rate_medio_Alamouti+0;

        end
    end

    vector_snrm(snrm)=snrm;
    prob_outage_Alamouti(snrm)=canales_outage_Alamouti/I;
     prob_error_paquete_Alamouti(snrm)=1-(1-
prob_outage_Alamouti(snrm))^(L);
    rate_Alamouti(snrm)=rate_medio_Alamouti/Nslots;

end

%Eligiendo solamente 1

for snrm=1:30
    Nslots=0;
    rate_medio_Mejor=0;
    canales_outage_Mejor=0;
    snrml=10^(snrm/10);

    varianza1(snrm)=((d/d0)^(-mu))*(landac/(4*pi*d0))^2;
    Ps=snrml*Pn/varianza1(snrm);

    for aux=1:I

        for cont=1:N

            varianza2=((distancia(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
            r(cont,
1)=sqrt(0.5)*sqrt(varianza2)*(randn(1,1)+j*randn(1,1));

        end
           for cont=1:N

            varianza3=((distancia2(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
            h(cont,
      1)=sqrt(0.5)*sqrt(varianza3)*(randn(1,1)+j*randn(1,1));

        end

        snrr1=(abs(r).^2)*Ps/(Pn*2);
        vect_outage=snrr1>=(2^(2*R)-1);
        Nslots=Nslots+1;
```

```
        if(sum(vect_outage)>=1)

            vect_sec_link=abs(h).^2.*vect_outage;
            [maxim,pos1]=max(vect_sec_link);         %Cogemos el 
máximo
            pos1=pos1-1;                            %Para que 
empiecen las posiciones en 0
            b1=dec2bin(pos1);                    %Pasamos las 
posiciones a binario
           a=rand(1);
           b=rand(1);
           c=rand(1);
           dd=rand(1);

           l=0;

           l(4)=b1(length(b1));

           if pos1>1

               l(3)=b1(length(b1)-1);
           end

           if pos1>3
               l(2)=b1(length(b1)-2);
           end

           if pos1>7
               l(1)=b1(length(b1)-3);
           end

           l=l>'0';

           if a<=Probe
               l(4)=not(l(4));
           end

           if b<=Probe && pos1>1
               l(3)=not(l(3));
           end

           if c<=Probe && pos1>3
               l(2)=not(l(2));
           end

           if dd<=Probe && pos1>7
               l(1)=not(l(1));
           end

           %Convertimos binario a decimal

           eleccion1=l(4)+2*l(3)+4*l(2)+8*l(1);
           eleccion1=eleccion1+1;


           if N<eleccion1
               vect_sec_link(eleccion1)=0;
           end
            snrr2=(Ps/(Pn*2))*(vect_sec_link(eleccion1));

            canales_outage_Mejor=canales_outage_Mejor
      (snrr2<(2^(2*R)-1));
```

```
            if snrr2>=(2^(2*R)-1)
                rate_medio_Mejor=rate_medio_Mejor+rateslot;
            else
                rate_medio_Mejor=rate_medio_Mejor+0;
            end

        else

            canales_outage_Mejor=canales_outage_Mejor+1;
            rate_medio_Mejor=rate_medio_Mejor+0;

        end

    end

    vector_snrm(snrm)=snrm;
    prob_outage_Mejor(snrm)=canales_outage_Mejor/I;
    prob_error_paquete_Mejor(snrm)=1-(1-
prob_outage_Mejor(snrm))^(L);
    rate_relay_Mejor(snrm)=rate_medio_Mejor/Nslots;

end

for snrm=1:30
    Nslots=0;
    rate_medio_Directo=0;
    canales_outage_Directo=0;
    snrml=10^(snrm/10);
    rateslot=R;

    varianza1(snrm)=((d/d0)^(-mu))*(landac/(4*pi*d0))^2;
    Ps=snrml*Pn/varianza1(snrm);

    for aux=1:I

    e=sqrt(0.5)*sqrt(varianza1)*(randn(1,1)+j*randn(1,1));

        snrr1=(abs(e).^2)*Ps/(Pn);

        vect_outage=snrr1>=(2^(2*R)-1);

        Nslots=Nslots+1;

        if(vect_outage==1)

            if snrr1>=(2^(2*R)-1)
                rate_medio_Directo=rate_medio_Directo+rateslot;
            else
                rate_medio_Directo=rate_medio_Directo+0;
            end

        else

            canales_outage_Directo=canales_outage_Directo+1;
            rate_medio_Directo=rate_medio_Directo+0;

        end

    end

    vector_snrm(snrm)=snrm;
    prob_outage_Directo(snrm)=canales_outage_Directo/I;
```

```
    prob_error_paquete_Directo(snrm)=1-(1-
prob_outage_Directo(snrm))^(L);
    rate_relay_Directo(snrm)=rate_medio_Directo/Nslots;

end
```

Ap2.2 Real Cooperative System

```
clear all;

I=300000;
N=7;
d=100;


distanciax=d*rand(N,1);
distanciay=(d/2)*rand(N,1);
distancia=sqrt(distanciax.^2+distanciay.^2);
distanciax2=d-distanciax;
distancia2=sqrt(distanciax2.^2+distanciay.^2);
fc=2.5*10^9;
c=3*10^8;
landac=c/fc;
d0=1;
mu=3;
Pn=1;
L=10; %Numero de simbolos en un paquete.

%PER Directo

for snrm=1:30

    rate_medio=0;
    fed_error=0;
    Nslots=0;
    error_simbolo_Directo=0;
    error_simbolo_Mejor=0;
    error_simbolo_Alamouti=0;
    snrml=10^(snrm/10);
    Pesim=10^(-2);
    varianza1=((d/d0)^(-mu))*(landac/(4*pi*d0))^2;
    Ps=snrml*Pn/varianza1;

    for aux=1:I

        e=sqrt(0.5)*sqrt(varianza1)*(randn(1,1)+j*randn(1,1));
        snr_directo=(abs(e).^2)*Ps/(Pn);
        Pesim_directo=2*(1/2*erfc(sqrt(snr_directo/2)))-
(1/2*erfc(sqrt(snr_directo/2))).^2;
        outage_directo=Pesim_directo<Pesim;

        if outage_directo==0
            error_simbolo_Directo=error_simbolo_Directo+1;
        else
            error_simbolo_Directo=error_simbolo_Directo+0;
        end
    end

    vector_snrm(snrm)=snrm;
```

```
    prob_outage_Directo(snrm)=error_simbolo_Directo/I;
    prob_error_paquete_Directo(snrm)=1-(1-
prob_outage_Directo(snrm))^(L);

    %Best Relay

    for aux=1:I

        for cont=1:N
         varianza2=((distancia(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
        r(cont,
1)=sqrt(0.5)*sqrt(varianza2)*(randn(1,1)+j*randn(1,1));
          end


    for cont=1:N
         varianza3=((distancia2(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
        h(cont,
1)=sqrt(0.5)*sqrt(varianza3)*(randn(1,1)+j*randn(1,1));
     end

    %Calculamos el decoding set

    snr_relays=(abs(r).^2)*Ps/(Pn*2);
    Pesim_relays=2*(1/2*erfc(sqrt(snr_relays/2)))-
(1/2*erfc(sqrt(snr_relays/2))).^2;
    outage_relays=Pesim_relays<Pesim;
    Pesim_relays=Pesim_relays.*outage_relays;

    if sum(outage_relays(1:end))==0
        Pesim_relays=inf;
    else
        Pesim_relays=sum(Pesim_relays(1:end))/
sum(outage_relays(1:end));
    end

    %Miramos los relays que estan en outage para transmitir al
Destination

    vect_sec_link=abs(h).^2.*outage_relays;
    snr_relays2=(Ps/(Pn*2)).*vect_sec_link;
    Pesim_relays2=2*(1/2*erfc(sqrt(snr_relays2/2)))-
(1/2*erfc(sqrt(snr_relays2/2))).^2;
    Pesim_relays2=Pesim_relays2.*outage_relays;
    outage_relays2=Pesim_relays2<Pesim;

    if sum(outage_relays2(1:end))==0
        Pesim_relays2=inf;
    else
        Pesim_relays2=sum(Pesim_relays2(1:end))/
sum(outage_relays2(1:end));
    end
     Pebit=Pesim_relays2/2;

    %Calculamos las mejores posiciones

    vect_sec_link=abs(h).^2.*outage_relays;
    [maxim,pos1]=max(vect_sec_link);         %Cogemos el m‡ximo
    vect_sec_link2=vect_sec_link(pos1);   %Lo guardamos
    vect_sec_link(pos1)=0;                   %Ponemos su posicion a
0
```

```
    [maxim2,pos2]=max(vect_sec_link);       %Cogemos el siguiente
maximo
    vect_sec_link(pos1)=vect_sec_link2;

    a=rand(1);
    b=rand(1);
    c=rand(1);
    dd=rand(1);

    pos1=pos1-1; %Para que empiezen las posiciones en 0
     pos2=pos2-1;

     b1=dec2bin(pos1); %Pasamos las posiciones a binario
     b2=dec2bin(pos2);
      l=0;
      l(4)=b1(length(b1));

     if pos1>1

        l(3)=b1(length(b1)-1);
     end
      if pos1>3
          l(2)=b1(length(b1)-2);
     end

    if pos1>7
         l(1)=b1(length(b1)-3);
    end

     l=l>'0';
        if a<=Pebit
           l(4)=not(l(4));
        end

       if b<=Pebit && pos1>1
           l(3)=not(l(3));
       end

       if c<=Pebit && pos1>3
           l(2)=not(l(2));
       end

       if dd<=Pebit && pos1>7
           l(1)=not(l(1));
       end

      %Convertimos binario a decimal

           eleccion1=l(4)+2*l(3)+4*l(2)+8*l(1);
           eleccion1=eleccion1+1;

           if N<eleccion1
                vect_sec_link(eleccion1)=0;
           end
       if(sum(outage_relays)>0)

        snr_mejor=(Ps/(Pn*2))*vect_sec_link(eleccion1);
        Pesim_mejor=2*(1/2*erfc(sqrt(snr_mejor/2)))-
(1/2*erfc(sqrt(snr_mejor/2))).^2;
        outage_mejor=Pesim_mejor<Pesim;

        if outage_mejor==0
            error_simbolo_Mejor=error_simbolo_Mejor+1;
```

```
        else
            error_simbolo_Mejor=error_simbolo_Mejor+0;
        end

    else
        error_simbolo_Mejor=error_simbolo_Mejor+1;
        rate_medio=rate_medio+0;
    end
    end
    vector_snrm(snrm)=snrm;
    prob_outage_Mejor(snrm)=error_simbolo_Mejor/I;
    prob_error_paquete_Mejor(snrm)=1-(1-
prob_outage_Mejor(snrm))^(L);



    %Alamouti

    for aux=1:I

        for cont=1:N

        varianza2=((distancia(cont)/d0)^(-mu))*(landac/(4*pi*d0))^2;
        r(cont,
1)=sqrt(0.5)*sqrt(varianza2)*(randn(1,1)+j*randn(1,1));

        end
     for cont=1:N

        varianza3=((distancia2(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
        h(cont,
1)=sqrt(0.5)*sqrt(varianza3)*(randn(1,1)+j*randn(1,1));

    end

    %Calculamos el decoding set

    snr_relays=(abs(r).^2)*Ps/(Pn*2);
    Pesim_relays=2*(1/2*erfc(sqrt(snr_relays/2)))-
(1/2*erfc(sqrt(snr_relays/2))).^2;
    outage_relays=Pesim_relays<Pesim;
    Pesim_relays=Pesim_relays.*outage_relays;

    if sum(outage_relays(1:end))==0
        Pesim_relays=inf;
    else
        Pesim_relays=sum(Pesim_relays(1:end))/
sum(outage_relays(1:end));
    end

    %Miramos los relays que est‡n en outage para transmitir al
Destination

    vect_sec_link=abs(h).^2.*outage_relays;
    snr_relays2=(Ps/(Pn*2)).*vect_sec_link;
    Pesim_relays2=2*(1/2*erfc(sqrt(snr_relays2/2)))-
(1/2*erfc(sqrt(snr_relays2/2))).^2;
    Pesim_relays2=Pesim_relays2.*outage_relays;
    outage_relays2=Pesim_relays2<Pesim;

    if sum(outage_relays2(1:end))==0
        Pesim_relays2=inf;
```

```
    else
        Pesim_relays2=sum(Pesim_relays2(1:end))/
sum(outage_relays2(1:end));
    end

    Pebit=Pesim_relays2/2;

    %Calculamos las mejores posiciones

    vect_sec_link=abs(h).^2.*outage_relays;
    [maxim,pos1]=max(vect_sec_link);         %Cogemos el m‡ximo
    vect_sec_link2=vect_sec_link(pos1);   %Lo guardamos
    vect_sec_link(pos1)=0;                   %Ponemos su posicion a
0
    [maxim2,pos2]=max(vect_sec_link);       %Cogemos el siguiente
maximo
    vect_sec_link(pos1)=vect_sec_link2;

    a=rand(1);
    b=rand(1);
    c=rand(1);
    dd=rand(1);

     pos1=pos1-1; %Para que empiezen las posiciones en 0
     pos2=pos2-1;

     b1=dec2bin(pos1); %Pasamos las posiciones a binario
     b2=dec2bin(pos2);

     l=0;

     l(4)=b1(length(b1));

     if pos1>1

        l(3)=b1(length(b1)-1);
     end

     if pos1>3
          l(2)=b1(length(b1)-2);
     end

    if pos1>7
         l(1)=b1(length(b1)-3);
    end

     l=l>'0';

       if a<=Pebit
           l(4)=not(l(4));

       end

       if b<=Pebit && pos1>1
           l(3)=not(l(3));
       end

       if c<=Pebit && pos1>3
           l(2)=not(l(2));
       end

       if dd<=Pebit && pos1>7
           l(1)=not(l(1));
```

```
   end

%Convertimos binario a decimal

   eleccion1=l(4)+2*l(3)+4*l(2)+8*l(1);

  %Ahora hacemos lo mismo para la segunda seleccion

   l=0;

   l(4)=b2(length(b2));

   if pos2>1

       l(3)=b2(length(b2)-1);
   end

   if pos2>3
       l(2)=b2(length(b2)-2);
   end

   if pos2>7
       l(1)=b2(length(b2)-3);
   end

   l=l>'0';

   a=rand(1);
   b=rand(1);
   c=rand(1);
   dd=rand(1);
    if a<=Pebit
       l(4)=not(l(4));

   end

   if b<=Pebit && pos1>1
       l(3)=not(l(3));
   end

   if c<=Pebit && pos1>3
       l(2)=not(l(2));
   end

   if dd<=Pebit && pos1>7
       l(1)=not(l(1));
   end

   %Convertimos binario a decimal

   eleccion2=l(4)+2*l(3)+4*l(2)+8*l(1);
   eleccion1=eleccion1+1;
   eleccion2=eleccion2+1;

   if N<eleccion1
       vect_sec_link(eleccion1)=0;
   end

   if N<eleccion2
       vect_sec_link(eleccion2)=0;
   end

if(sum(outage_relays)>=2)
```

```
           snr_Alamouti=(Ps/
(Pn*4))*(vect_sec_link(eleccion1)+vect_sec_link(eleccion2));
           Pesim_Alamouti=2*(2*(1/2*erfc(sqrt(snr_Alamouti/2)))-
(1/2*erfc(sqrt(snr_Alamouti/2))).^2);
           outage_Alamouti=Pesim_Alamouti<Pesim;

           if outage_Alamouti==0
                error_simbolo_Alamouti=error_simbolo_Alamouti+1;
            else
                error_simbolo_Alamouti=error_simbolo_Alamouti+0;
            end
    else
        error_simbolo_Alamouti=error_simbolo_Alamouti+1;

    end
    end

    vector_snrm(snrm)=snrm;
    prob_outage_Alamouti(snrm)=error_simbolo_Alamouti/I;
    prob_error_paquete_Alamouti(snrm)=1-(1-
prob_outage_Alamouti(snrm))^(L);
end
```

Ap2.3 Correlated Channel Model

```
clear all;

I=100000;
N=4;
d=100;

distanciax=d*rand(N,1);
distanciay=(d/2)*rand(N,1);

distancia=sqrt(distanciax.^2+distanciay.^2);

distanciax2=d-distanciax;
distancia2=sqrt(distanciax2.^2+distanciay.^2);


fc=2.5*10^9;
c=3*10^8;
landac=c/fc;
d0=1;
mu=3;
Pn=1;
L=10; %Numero de s’mbolos en un paquete.
v=1; %Velocidad media
fd=v/landac; %Frecuencia Doppler
Tm=2*10^-3; %Tiempo de muestra
ro=besselj(0,2*pi*fd*Tm); %Correlaci—n entre dos instantes
temporales
%ro=0.6;
r(N,1)=0;
h(N,1)=0;
w(1)=0; %Variables de correlaci—n
x(1)=0;
usado_Mejor=1;
```

```
usado_Alamouti1=1;
usado_Alamouti2=2;

for snrm=1:30

    rate_medio=0;
    fed_error=0;
    Nslots=0;
    error_simbolo=0;
    snrml=10^(snrm/10);
    Pesim=10^(-2);

    varianza1=((d/d0)^(-mu))*(landac/(4*pi*d0))^2;
    Ps=snrml*Pn/varianza1;

    for aux=1:I

        mejor=0;

        for cont=1:N
            if aux==1
                varianza2=((distancia(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
                r(cont,
1)=sqrt(0.5)*sqrt(varianza2)*(randn(1,1)+j*randn(1,1));
                w(cont,1)=r(cont,1)/sqrt(varianza2);
            else

                varianza2=((distancia(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
                canal1(cont,1)=ro*w(cont,1)+sqrt(0.5)*sqrt(1-
ro^2)*(randn(1,1)+j*randn(1,1));
                r(cont,1)=canal1(cont,1)*sqrt(varianza2);
                w(cont,1)=canal1(cont,1);
            end
        end
        for cont=1:N

            if aux==1
                varianza3=((distancia2(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
                h(cont,
1)=sqrt(0.5)*sqrt(varianza3)*(randn(1,1)+j*randn(1,1));
                x(cont,1)=h(cont,1)/sqrt(varianza3);
            else

                varianza3=((distancia2(cont)/d0)^(-mu))*(landac/
(4*pi*d0))^2;
                canal2(cont,1)=ro*x(cont,1)+sqrt(0.5)*sqrt(1-
ro^2)*(randn(1,1)+j*randn(1,1));
                h(cont,1)=canal2(cont,1)*sqrt(varianza3);
                x(cont,1)=canal2(cont,1);
            end
    end
        e=sqrt(0.5)*sqrt(varianza1)*(randn(1,1)+j*randn(1,1));

        snr_directo=(abs(e).^2)*Ps/(Pn);
        Pesim_directo=2*(1/2*erfc(sqrt(snr_directo/2)))-
(1/2*erfc(sqrt(snr_directo/2))).^2;
        outage_directo=Pesim_directo<Pesim;


        %Calculamos el decoding set
```

```
    snr_relays=(abs(r).^2)*Ps/(Pn*2);
    Pesim_relays=2*(1/2*erfc(sqrt(snr_relays/2)))-
(1/2*erfc(sqrt(snr_relays/2))).^2;
    outage_relays=Pesim_relays<Pesim;
    Pesim_relays=Pesim_relays.*outage_relays;

    if sum(outage_relays(1:end))==0
        Pesim_relays=inf;
    else
        Pesim_relays=sum(Pesim_relays(1:end))/
sum(outage_relays(1:end));
    end



    %Miramos los relays que est‡n en outage para transmitir al
Destination

    vect_sec_link=abs(h).^2.*outage_relays;
    snr_relays2=(Ps/(Pn*2)).*vect_sec_link;
    Pesim_relays2=2*(1/2*erfc(sqrt(snr_relays2/2)))-
(1/2*erfc(sqrt(snr_relays2/2))).^2;
    Pesim_relays2=Pesim_relays2.*outage_relays;
    outage_relays2=Pesim_relays2<Pesim;

    if sum(outage_relays2(1:end))==0
        Pesim_relays2=inf;
    else
        Pesim_relays2=sum(Pesim_relays2(1:end))/
sum(outage_relays2(1:end));
    end




        %Miramos cual va a ser la velocidad necesaria

        if outage_directo==1
            rate=4;
            Nslots=Nslots+1;
            Pebit=Pesim_directo/2;
        else
            rate=2;
            Nslots=Nslots+1;
            Pebit=Pesim_relays2/2;

        end

        a=rand(1);
        b=rand(1);
        c=rand(1);
        dd=rand(1);

            %Segun SNR

        if outage_directo==0
            if N<=2
                mejor=1;
            else if N<=5 && snrm<21
                    mejor=1;
                else if N<=10 && snrm<10
                        mejor=1;
```

```
                    else if N<=15 && snrm<7
                            mejor=1;
                        end
                    end
                end
            end
        end



        vect_sec_link=abs(h).^2.*outage_relays;
        [maxim,pos1]=max(vect_sec_link);         %Cogemos el m‡ximo
        vect_sec_link2=vect_sec_link(pos1);   %Lo guardamos
        vect_sec_link(pos1)=0;                   %Ponemos su
posicion a 0
        [maxim2,pos2]=max(vect_sec_link);       %Cogemos el
siguiente maximo
         vect_sec_link(pos1)=vect_sec_link2;   %Volvemos a poner el
valor m‡ximo en su posicion


         pos1=pos1-1; %Para que empiezen las posiciones en 0
         pos2=pos2-1;

         b1=dec2bin(pos1); %Pasamos las posiciones a binario
         b2=dec2bin(pos2);

         l=0;

         l(4)=b1(length(b1));

         if pos1>1

            l(3)=b1(length(b1)-1);
         end

         if pos1>3
              l(2)=b1(length(b1)-2);
         end

        if pos1>7
             l(1)=b1(length(b1)-3);
        end

         l=l>'0';
           if a<=Pebit
               l(4)=not(l(4));
           end

           if b<=Pebit %&& pos1>1
               l(3)=not(l(3));
           end

           if c<=Pebit %&& pos1>3
               l(2)=not(l(2));
           end

           if dd<=Pebit %&& pos1>7
               l(1)=not(l(1));
           end

      %Convertimos binario a decimal
```

```
           eleccion1=l(4)+2*l(3)+4*l(2)+8*l(1);

           l=0;

           l(4)=b2(length(b2));

           if pos2>1

               l(3)=b2(length(b2)-1);
           end

           if pos2>3
               l(2)=b2(length(b2)-2);
           end

           if pos2>7
               l(1)=b2(length(b2)-3);
           end

           l=l>'0';

           a=rand(1);
           b=rand(1);
           c=rand(1);
           dd=rand(1);

           if a<=Pebit
               l(4)=not(l(4));
           end

           if b<=Pebit %&& pos1>1
               l(3)=not(l(3));
           end

           if c<=Pebit %&& pos1>3
               l(2)=not(l(2));
           end

           if dd<=Pebit %&& pos1>7
               l(1)=not(l(1));
           end

           %Convertimos binario a decimal

           eleccion2=l(4)+2*l(3)+4*l(2)+8*l(1);
           eleccion1=eleccion1+1;
           eleccion2=eleccion2+1;

        if outage_directo==0                         %Miramos que
pasa cuando no se puede transmitir de forma directa

            if mejor==1                          %Mejor Relay

                if(sum(outage_relays)>0)

                   if N<eleccion1
                        eleccion1=usado_Mejor;
                    end

                    snr_mejor=(Ps/(Pn*2))*vect_sec_link(eleccion1);
                    Pesim_mejor=2*(1/2*erfc(sqrt(snr_mejor/2)))-
(1/2*erfc(sqrt(snr_mejor/2))).^2;
                    outage_mejor=Pesim_mejor<Pesim;
```

```
                    if outage_mejor==1
                        rate_medio=rate_medio+rate;
                    else
                        rate_medio=rate_medio+0;
                        error_simbolo=error_simbolo+1;
                    end

                else
                    error_simbolo=error_simbolo+1;
                    rate_medio=rate_medio+0;
                end
               usado_Mejor=eleccion1;

            else                                %Usamos Alamouti
                   if sum(outage_relays)>=2

                       %En caso de haber seleccionado un relay que
no existe
                      %ponemos ele elegido anteriormente

                        if N<eleccion1
                            if usado_Alamouti1==eleccion2
                                 eleccion1=usado_Alamouti2;
                            else
                            eleccion1=usado_Alamouti1;
                         end

                        if N<eleccion2
                            if usado_Alamouti2==eleccion1
                                eleccion2=usado_Alamouti1;
                            else
                            eleccion2=usado_Alamouti2;
                        end

                         snr_Alamouti=(Ps/
(Pn*4))*(vect_sec_link(eleccion1)+vect_sec_link(eleccion2));

Pesim_Alamouti=2*2*(1/2*erfc(sqrt(snr_Alamouti/2)))-
(1/2*erfc(sqrt(snr_Alamouti/2))).^2;
                         outage_Alamouti=Pesim_Alamouti<Pesim;

                         if outage_Alamouti==1
                             rate_medio=rate_medio+rate;
                         else
                             rate_medio=rate_medio+0;
                             error_simbolo=error_simbolo+1;
                          end

                else
                         error_simbolo=error_simbolo+1;
                         rate_medio=rate_medio+0;
                end
                        usado_Alamouti1=eleccion1;
                        usado_Alamouti2=eleccion2;

                end
                end
            end
        end
    end

    Pebit(snrm)=Pebit;
```

```
    vector_snrm(snrm)=snrm;
    prob_error_simbolo(snrm)=error_simbolo/I;
    prob_error_paquete(snrm)=1-(1-prob_error_simbolo(snrm))^(L);
    vector_rate(snrm)=rate_medio/Nslots;
end
```

Abstract

Les comunicacions cooperatives estan guanyant un gran interès en les comunicacions modernes degut a que permeten millorar la transmissió d'informació entre un emissor i un receptor utilitzant una sèrie de terminals situats entre ells. Aquest projecte és un estudi complet del sistemes cooperatius, analitzant el seu rendiment i comparant l'ús d'un sol d'aquests terminals amb l'ús del còdi Alamouti, que utilitza dos terminals. Primer hi ha una introducció als sistemes cooperatius y a la teoria de la informació. Després hem estudiat un sistema cooperatiu amb la teoria de la informació com a base, en termes de probabilitat de fallada del sistema, y posteriorment l'hem adaptat a un sistema cooperatiu real utilitzant una modulació QPSK, estudiant la seva probabilitat d'error de paquet. Finalment es proposen diversos protocols que permeten millorar el rendiment del sistema cooperatiu estudiat.

……

Las comunicaciones cooperativas están ganando mucho interés en las comunicaciones modernas debido a que permiten mejorar la transmisión de información entre un emisor y un receptor usando una serie de terminales intermedios. Este proyecto es un estudio completo de los sistemas cooperativos, analizando su rendimiento y comparando el uso de uno solo de estos terminales con el uso del código Alamouti, el cual usa dos terminales. Primero hay una introducción a los sistemas cooperativos y a la teoría de la información. Después hemos estudiado un sistema cooperativo con la teoría de la información como base, en términos de probabilidad de fallo del sistema, y posteriormente lo hemos adaptado a un sistema cooperativo real usando una modulación QPSK, estudiando su probabilidad de error de paquete. Finalmente se proponen varios protocolos que permiten mejorar el rendimiento del sistema cooperativo estudiado.

……

Cooperative communications are gaining much interest in modern communications because they allow to improve the information transmission between a source and a destination using various intermediate terminals. This project is a complete study of cooperative systems, analyzing its performance and comparing the use of a single terminal with the use of the Alamouti code, which uses two terminals. First, there is an introduction to cooperative systems and to information theory. Then we have studied a cooperative system using the information theory, in terms of outage probability, and subsequently we have adapted it to a real cooperative system using a QPSK modulation, studying its packet error probability. Finally several protocols are proposed to improve the performance of the studied cooperative system.