Low complexity algorithms for faster-than-Nyquist signaling

Using coding to avoid an NP-hard problem

EMIL RINGH

Master of Science Thesis
Stockholm, Sweden 2013
Low complexity algorithms for faster-than-Nyquist signaling

Using coding to avoid an NP-hard problem

Emil Ringh

Master’s Thesis in Optimization and Systems Theory (30 ECTS credits)
Master Programme in Mathematics (120 credits)
Royal Institute of Technology year 2013
Supervisor at Ericsson AB was Ather Gattami
Supervisor at KTH was Johan Karlsson
Examiner was Johan Karlsson

TRITA-MAT-E 2013:57
ISRN-KTH/MAT/E--13/57--SE
Abstract

This thesis is an investigation of what happens when communication links are pushed towards their limits and the data-bearing-pulses are packed tighter in time than previously done. This is called faster-than-Nyquist (FTN) signaling and it will violate the Nyquist inter-symbol interference criterion, implying that the data-pulses are no longer orthogonal and thus that the samples at the receiver will be dependent on more than one of the transmitted symbols. Inter-symbol interference (ISI) has occurred and the consequences of it are studied for the AWGN-channel model. Here it is shown that in order to do maximum likelihood estimation on these samples the receiver will face an NP-hard problem. The standard algorithm to make good estimations in the ISI case is the Viterbi algorithm, but applied on a block with $N$ bits and interference among $K$ bits the complexity is $O(N \cdot 2^K)$, hence limiting the practical applicability. Here, a precoding scheme is proposed together with a decoding that reduce the estimation complexity. By applying the proposed precoding/decoding to a data block of length $N$ the estimation can be done in $O(N^2)$ operations preceded by a single off-line $O(N^3)$ calculation. The precoding itself is also done in $O(N^2)$ operations, with a single off-line operation of $O(N^3)$ complexity.

The strength of the precoding is shown in simulations. In the first it was tested together with turbo codes of code rate $2/3$ and block length of 6000 bits. When sending 25% more data (FTN) the non-precoded case needed about 2.5 dB higher signal-to-noise ratio (SNR) to have the same error rate as the precoded case. When the precoded case performed without any block errors, the non-precoded case still had a block error rate almost equal to 1.

We also studied the scenario of transmission with low latency and high reliability. Here, 600 bits were transmitted with a code rate of $2/3$, and hence the target was to communicate 400 bits of data. Applying FTN with double packing, that is transmitting 1200 bits during the same amount of time, it was possible to lower the code rate to $1/3$ since only 400 bits of data was to be communicated. This technique greatly improves the robustness. When the FTN case performed error free, the classical Nyquist case still had a block error rate of 0.19. To reach error free performance the Nyquist case needed 1.25 dB higher SNR compared to the precoded FTN case with lower code rate.
Sammanfattning


Ett annat scenario som testades var det med korta koder, liten fördröjning och hög robusthet. I detta scenario skickades 600 bitar med en kodningsgrad på 2/3, alltså 400 bitar ren data. Genom att använda FTN med en dubbel packningsgrad, vilket innebär att 1200 bitar skickades under samma tid, var det möjligt att sänka kodningsgraden till 1/3, eftersom det bara var 400 bitar ren data som skulle överföras. Detta ökade robustheten i systemet ty då FTN fallet gjorde felfritt hade det klassiska Nyquist fallet fortfarande en felfrekvens på 0.19 för sina block. Det krävdes 1.25 dB högre SNR för Nyquist fallet att bli felfritt jämfört med FTN och lägre kodningsgrad.

**Titel:** Algoritmer med låg komplexitet för snabbare-än-Nyquist signalering
Acknowledgements

I would like to thank my supervisor Ather Gattami for providing me with such an interesting topic, supporting me in my work, and believing in me despite my different academic background. I would also like to thank Ericsson and the RAT-department for inviting me to work with them, it has been a great experience.

Moreover I would like to thank Johan Karlsson for interesting discussions on the topic and helpful comments on the report. Among other things, the relationship in Theorem 15 was pointed out to me by him.

Lastly but not least I aim my thanks towards my family and friends, especially my musketeers Martin, Björn, and Axel as well as my girlfriend Frida; without you life had not been this much fun!
Contents

1 Preliminaries 1
   1.1 Information theory ............................................. 1
   1.2 Signals and Communication ..................................... 6
   1.3 Matrix theory .................................................. 14

2 Motivation and related research 17
   2.1 Idea, motivations, and challenges with FTN .................. 17
   2.2 Background, Saltzberg and Mazo ............................... 18
   2.3 Related research ............................................... 21

3 Receiving and detecting FTN signals 23
   3.1 Premises for Faster-Than-Nyquist Signaling .................. 24
   3.2 The difficulty of solving the ML-estimation problem .......... 29
   3.3 SVD-precoding .................................................. 33
   3.4 Root-raised-Cosine pulses in FTN ............................. 36

4 Linear Precoding 43
   4.1 GTMH-precoding ............................................... 43
   4.2 Some words on general linear precoding ...................... 45

5 Numerical results 47
   5.1 Numerical investigation of the Eigenvalues .................. 47
   5.2 Simulation 1; description and benchmark simulation ......... 48
   5.3 Simulation 2; comparing GTMH, SVD, and no precoding in FTN 52
   5.4 Simulation 3; non-turbo-coded comparison of the precoding schemes 52
   5.5 Simulation 4; application to low latency communication, improving short code performance ............................ 55

A Deriving the Szegö function for root-raised-cosine 57

B Eigenvalue spectra of the Gram matrix and frequency spectra of the pulse 61
C Saltzberg formulation, the FTN dual of Mazo formulation 63
Bibliography 65
Chapter 1

Preliminaries

Faster-than-Nyquist signaling (FTN) is a topic that relates to three different areas: Information theory, Signal processing, and Communication theory. This chapter covers some of the basics in order to bring the reader up to speed and provide a foundation for further discussion of the topics of this thesis. Most of it can be found in textbooks on respective subject.

1.1 Information theory

We start this text on information transmission from the ever fundamental question: What is information? The full aspects of this rather philosophical question will not be studied here. However, what we do need is a measure of the information content of a given entity. This question was addressed already in the 1920’s and the 1940’s by researchers like Shannon, Hartley, and Nyquist who argued that an adequate measure of information must reflect the number of possible choices at the sender end [1–3].

Measure of information

Intuitively and much like the previous argumentation, information is connected to revealing uncertainties. Conveying a specific choice from a set of possible choices. After communicating ’1 + 1 = ’ then the statement ’2’ reveals no information since it is deterministic and involves no uncertainty, choice, or randomness. Mathematically, choice or randomness is connected to random variables and their corresponding probabilities. In [1, 2, 4] we find the definition of information.

Definition 1 (Information). The amount of information associated with an outcome $x$ of a discrete random variable $X$ is

$$h(x) = -K \cdot \ln(p(x))$$  \hspace{1cm} (1.1)
where $\ln(\cdot)$ denotes the natural logarithm, $p(x)$ is the probability for the outcome $x$ of the stochastic variable $X$, and $K > 0$ is a constant.

This definition has a lot of intuitive properties, for example:

1. It is increasing with decreasing probability. An observation revealing an improbable outcome has conveyed more information than one that resulted in a very probable outcome. Something that is deterministic we know for sure and hence it constitutes no information, for example ’$1 + 1 = X’$ has $p(X = 2) = 1$, and hence no information is gained.

2. It is additive for independent variables, $h(x, y) = -K \cdot \ln(p(x, y)) = -K \cdot \ln(p(x)) - K \cdot \ln(p(y)) = h(x) + h(y)$ if $X$ and $Y$ are independent, but not otherwise since then knowledge of one outcome will contribute to knowledge about the other (rather reduce the uncertainty).

There is no corresponding definition for the continuous random variable, this is because the probability of a continuous variable taking a specific value is zero $p(X = a) = \int_a^a f(x) \, dx = 0$.

The constant $K$ determines the units in which information is measured, thus having $K$ as a unit-less constant we see that information has no explicit unit in which information is measured. Therefore it would in principle be wrong to talk about information units, but there is no more error in this than in talking about units for angles like for example degrees or radians. Normally we will have $K = 1 / \ln(2)$ in which case an equivalent definition is $h(x) = -\log(p(x)) = \log(1/p(x))$ and the unit is called bit. Other values of $K$ are $K = 1$ and the corresponding unit is then nat, or $K = 1 / \ln(10)$ with the corresponding unit dit (from decimal digit), ban, or Hart (from Hartley). I will keep the constant $K$ throughout this section, but in the following sections the unit bits will be used unless stated otherwise.

To quantify the uncertainty in a random variable $X$, Shannon introduced the term entropy [1,5] and defined it as follows.

**Definition 2** (Entropy). The entropy of a random variable $X$ is

$$
H(X) = \begin{cases} 
-K \cdot \sum_{x \in X} p(x) \cdot \ln(p(x)) & \text{if } X \text{ is a discrete random variable} \\
-K \cdot \int_{-\infty}^{\infty} f(x) \cdot \ln(f(x)) \, dx & \text{if } X \text{ is a continuous random variable} 
\end{cases} 
$$

for some constant $K > 0$. $p(x)$ is the probability of the outcome $x$ and $f(x)$ is the probability density function. The summation over $x \in X$ means that $x$ should range through all values in the sample space of $X$.

\footnote{Throughout this thesis I will use $\ln(\cdot)$ to denote the natural base logarithm, $\log(\cdot)$ to denote the base 2 logarithm, and $\log_b(\cdot)$ to denote the base $b$ logarithm.}
Here we take it as a definition that $0 \cdot \log(0) = 0$ since $\lim_{p \to 0^+} p \cdot \log(p) = 0$.

Having defined information we see that the entropy is the expected information gain of observing the random variable

$$H(X) = -K \cdot \sum_{x \in X} p(x) \cdot \log(p(x)) = \sum_{x \in X} p(x) \cdot \left( -K \cdot \log(p(x)) \right)$$

$$= \sum_{x \in X} p(x) \cdot h(x) = E[h(X)].$$

Thus, entropy is a measure of the uncertainty of a random variable.

If we have more than one random variable we can also define the conditional entropy as well as the mutual information [1,4,5].

**Definition 3** (Conditional entropy). The conditional entropy for the random variable $X$ conditioned on the random variable $Y$ is given as

$$H(X|Y) = E[h(X|Y)] = \sum_{y \in Y} p(y) \cdot \left( -K \cdot \sum_{x \in X} p(x|y) \cdot \log(p(x|y)) \right)$$

$$= -K \cdot \sum_{x \in X, y \in Y} p(x,y) \cdot \log(p(x|y)) \quad (1.3)$$

This measures the average uncertainty left in $X$ if we know the outcome of $Y$. This is useful if we want to relate two random variables, for example a sent message $X$ and a received message $Y$. One can observe that if $X$ and $Y$ are independent, then $H(X|Y) = H(X)$, since $Y$ gives no information about $X$ and hence the average uncertainty in $X$ is unchanged.

**Definition 4** (Mutual information). The mutual information for two random variables $X$ and $Y$ is given as

$$I(X,Y) = H(X) - H(X|Y) \quad (1.4)$$

This in turn measures the average information gained on $X$, by observing $Y$, and vice versa (the property is symmetric, $I(X,Y) = I(Y,X)$), thus the following definition comes naturally.

**Definition 5** (Rate of transmission). The rate of transmission in a channel having $X$ as an input and $Y$ as an output is defined as

$$R = I(X,Y) \quad (1.5)$$

The rate measures how much we get to know about the input signal given the output signal, and since the input signal is considered to be the information source, this is how much information that is transmitted. One can see that for a noiseless channel, where $p(x|y) = 1$ if $x = y$ and 0 otherwise, we have $H(X|Y) = 0$ and hence...
the rate is $H(X)$ which is the amount of information produced by the information source at sender.\footnote{This can be considered a rate since the entropy is given as information per observed symbol (cf. Theorem 1), and thus by knowing the symbol/time we easily get data/time.}

In communication the message is often coded to increase robustness against noise, and there may also occur errors of different kinds. We define these as in [4].

**Definition 6** (Block code). A block code is a code that works on a sequence of digits called a block, $B = (b_1, b_2, \ldots, b_K)$. To this sequence the code appends some redundancy bits (possibly mixing the bits as well), creating a sequence of length $N$, $\beta = (\beta_1, \beta_2, \ldots, \beta_N)$, $N > K$.

In communication some of the $\beta_i$:s may then be corrupted, but it might still be possible to recover the vector $B$ as communicated by the sender. The code rate is defined as:

$$\text{Code rate} = \frac{K}{N}. \tag{1.6}$$

**Definition 7** (Errors). A block of bits as in Definition 6 is coded, sent and then received as $\hat{\beta} = (\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_N)$. It is then decoded to $\hat{B} = (\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_K)$. From this notion we define the probability of bit error as:

$$p_b = \frac{1}{K} \sum_{i=1}^{K} P(\hat{b}_i \neq b_i). \tag{1.7}$$

that is, the average probability that a bit is flipped in the decoded received vector. We also define the probability of block error as:

$$p_B = P(\hat{B} \neq B). \tag{1.8}$$

### Shannon’s theorems

This section presents some fundamental theorems that links the quantities defined in the previous section to physical representation of data. All three theorems presented here, Theorem 1, 2, and 3 are due to Shannon [1,6] but Theorems 1 and 2 has more accessible formulations in [4]. Rephrased versions can be found in any standard text on information theory, for example [5].

**Theorem 1** (Shannon source coding theorem). Let $X_1, X_2, \ldots, X_N$ be $N$ independent and identically distributed random variables with entropy $H(X)$. Then as $N \to \infty$, there exists a coding such that these can be compressed into $N \cdot H(X)$ bits, having an arbitrarily small probability for information loss. Conversely if one compress to less than $N \cdot H(X)$ bits information will almost certainly be lost.\footnote{Almost certainly, also known as almost surely [7] is a technical definition meaning that it is not the only theoretical outcome, but any other outcome has a probability smaller than any fixed positive number.}
Thus from this theorem we can see that the entropy is actually a statistically adequate measure of the information content of a random variable. In the case where we sent a message this message will consist of a series of symbols all unknown to the receiver and hence to be regarded as random. In this case we can look at entropy as the average information per symbol.

A channel is a model of a medium used to communicate the information. Shannon defines the capacity of a noisy channel in terms of the mutual information, how much we get to know about the sent signal $X$ when receiving the signal $Y$.

**Definition 8 (Capacity of a channel).** The capacity of a discrete, noisy channel is

$$C = \max_X H(X) - H(X|Y) = \max_X I(X,Y) = \max_X R$$  \hspace{1cm} (1.9)

where the maximum is taken over all different random variables $X$. The random variable $Y$ is the output and will be determined by $X$ and the properties of the channel.

The capacity is thus defined as the maximum rate achievable with $X$ as the sent signal and $Y$ as the output signal (although $I(X,Y)$ is symmetric). The physical interpretation of the max is that any channel should be judged by the best suited input. From the next theorem, it follows that this is a physically relevant definition.

**Theorem 2 (Noisy-channel coding theorem).** For every discrete memoryless channel with capacity $C$, and given any $\epsilon > 0$ and $r < C$ there exists a code of length $N$. This code can for large enough $N$ be transmitted over the channel with rate $R \geq r$ and with block error probability $p_B < \epsilon$.

Furthermore for a given bit error probability $p_b$, $R(p_b) = \frac{C}{1 - H(X(p_b))}$ is the highest possible rate and this rate is indeed achievable. Here $X(p_b)$ is a binary variable with $p(0) = p_b$ and $p(1) = 1 - p_b$.

Thus we see that the capacity defined as in Definition 8 is indeed a valid measurement on the capacity of a channel since it limits the rates at which we can communicate error-free over a noisy channel. Here a discrete channel is a channel that accepts a finite set of inputs $X$ and produces a finite set of outputs $Y$, perturbed by a noise $Z$ that is independent of $X$. The memoryless property is that the input-output at one instant is unaffected by the previous times, this will however not be the case later when we look at inter-symbol interference. More formally one uses a sequence of identically distributed channels, one at each signaling time.

The third theorem gives the capacity of a so called AWGN channel (additive white Gaussian noise) and together with Theorem 2 this gives an upper limit for how much data that can reliably be transferred over a channel with these properties \cite{1,4,8}.

**Theorem 3 (AWGN Channel capacity, band limited signaling).** For a communication channel with bandwidth $\Delta f$ and average transmission power $P$, that is affected
by additive white Gaussian noise with power spectral density $N_0/2$, the communication capacity is

$$C = \Delta f \cdot \log \left(1 + \frac{P}{\Delta f \cdot N_0}\right). \quad (1.10)$$

This theorem describes the maximum capacity for optimal usage of spectrum. To make predictions for all types of signals, not only optimal but non-realizable ones, the following extension is used [9]. Here $H(f)$ is the Fourier transform of the pulse.

$$C = \int_0^\infty \log \left(1 + \frac{2P}{N_0} |H(f)|^2\right) \, df \quad (1.11)$$

A further description of the AWGN channel follows next.

## 1.2 Signals and Communication

In the previous section we saw how much data that could be sent over an AWGN channel (c.f. Theorem 3). In this section we will look closer on how data is actually transmitted.

### Signals and signal space

A signal is a function of time, $h(t)$. We will call it time limited if it has finite support\(^4\), and band limited if the Fourier transform has finite support. These are mathematical definitions and there is a fundamental theorem that says that a signal cannot be both band limited and time limited at the same time [10]. In practice though we must have signals that are both time and band limited but that is a different problem altogether [11]. For the mathematical investigation we assume band limited signals and we will only consider pulses $h(t)$ that are in $L^2(-\infty, \infty)$, this is a Hilbert space and consists of all functions $h(t)$ such that $\int_{-\infty}^{\infty} |h(t)|^2 \, dt < \infty$.

**Definition 9** (Inner product). Let $\overline{\cdot}$ denote the complex conjugate. Then the inner product in $L^2(-\infty, \infty)$ is

$$\langle h(t), g(t) \rangle = \int_{-\infty}^{\infty} h(t) \cdot \overline{g(t)} \, dt \quad (1.12)$$

As usual, two signals are orthogonal if they have inner product equal to zero. The norm of a signal $h(t)$ is defined as $|h(t)| = \langle h(t), h(t) \rangle^{1/2}$ and the square is called energy. This is because the norm-square is proportional to energy of a corresponding physical realization of the signal.

---

\(^4\)i.e. identically zero outside some bounded interval.
1.2. SIGNALS AND COMMUNICATION

**Definition 10 (Norm).** The square of the norm, or energy, of a signal $h(t)$ is

$$E_{h(t)} = ||h(t)||^2 = \langle h(t), h(t) \rangle = \int_{-\infty}^{\infty} |h(t)|^2 \, dt$$

(1.13)

where the subscript can be omitted if it is obvious what signal we are talking about. The norm is then the positive root of this expression.

We now turn our attention to some special types of signals. These are called $T$-orthogonal or shift orthogonal, since if shifted an integer time $T$ they are orthogonal with respect to the inner product defined [9, 10].

**Definition 11 (Shift orthogonal pulse).** A $T$-orthogonal pulse is a signal $h_T(t)$ such that

$$\langle h_T(t), h_T(t + nT) \rangle = 0 \quad \forall n \in \mathbb{Z}\{0\}.$$  

(1.14)

Here the subscript $T$ denotes the time at which it is orthogonal.

This is a useful set of signals since the message decoding becomes very easy (as seen later in this section). Another set of signals often used are so called Nyquist pulses [8, 10]. They have their name from that they fulfill a criterion introduced by Nyquist in 1928 [3]. This criterion assures that the pulses does not introduce inter-symbol interference (ISI) [8, 10]. This means that at receiving the pulses are seen independent carriers of information.

**Definition 12 (Nyquist pulse).** A Nyquist pulse $h(t)$ is a pulse that fulfills the Nyquist ISI criterion

$$h(nT) = \begin{cases} 1 & n = 0 \\ 0 & n \in \mathbb{Z}\{0\} \end{cases} \iff \frac{1}{T} \sum_{k=-\infty}^{+\infty} H \left( f + \frac{k}{T} \right) = 1$$

(1.15)

where $H$ is the Fourier transform of $h$ and the right hand side sum holds for all $f$ except for possibly a set of frequencies of measure zero.

Lapidoth gives good insight about the relation between Nyquist pulses and shift orthogonal pulses with the following theorem [10].

**Theorem 4 (Characterization of shift orthogonal pulses).** Let $h(t)$ be a band limited pulse with finite energy and let $H(f)$ be the Fourier transform of it. Then it is a $T$-orthogonal pulse if and only if

$$\frac{1}{T} \sum_{k=-\infty}^{+\infty} \left| H \left( f + \frac{k}{T} \right) \right|^2 = 1$$

(1.16)

and the sum holds for all $f$ except for possibly a set of frequencies of measure zero.
Thus, any shift orthogonal pulse can be used to create a Nyquist pulse by convoluting it with itself in time (that corresponds to multiplication in frequency domain).

A special type of pulse that is both a shift orthogonal pulse and a Nyquist pulse is the sinc pulse:

\[ \tilde{g}_T(t) = \text{sinc}_T(t) = \frac{\sin(\pi \cdot t/T)}{(\pi \cdot t/T)} \]  

(1.17)  

with Fourier transform

\[ \tilde{G}_T(\omega) = T \text{rect}[\pi/T, \pi/T](\omega) = \begin{cases} T & \omega \in [-\pi/T, \pi/T] \\ 0 & \text{otherwise.} \end{cases} \]  

(1.18)  

When explicitly written out as sinc(\(t\)) it will be assumed that \(T = 1\). The shift orthogonality is going to be explicitly shown since the calculations will resemble some that are to come later.

\[ \langle \tilde{g}_T(t - m \cdot T), \tilde{g}_T(t - n \cdot T) \rangle = \int_{-\infty}^{\infty} \text{sinc}_T(t - m \cdot T) \cdot \overline{\text{sinc}_T(t - n \cdot T)} \, dt = \{\text{Fourier transform, Plancherel’s formulae} \} = \frac{T}{2} \]  

Thus, we see that the sinc pulse of parameter \(T\) is orthogonal to another sinc pulse of parameter \(T\) that is shifted an integer number of steps. As a bonus we also get to see that the energy of a sinc pulse of parameter \(T\) is the parameter itself. Normally we are going to work with the normalized sinc pulse,

\[ g_T(t) = \frac{\sin(\pi \cdot t/T)}{(\pi \cdot t/\sqrt{T})} , \]  

(1.19)  

having the Fourier transform

\[ G_T(\omega) = \sqrt{T} \text{rect}[\pi/T, \pi/T](\omega) = \begin{cases} \sqrt{T} & \omega \in [-\pi/T, \pi/T] \\ 0 & \text{otherwise.} \end{cases} \]  

(1.20)  

Another shift orthogonal pulse, mostly used in communication, is the root-raised-cosine pulse [10,12], which is defined as:

\[ g_{T,\beta}(t) = \frac{4\beta}{\pi \sqrt{T}} \cdot \cos\left((1 + \beta) \frac{\pi t}{T}\right) + \frac{\sin((1-\beta) \pi t)}{4\beta^2 T} \cdot \frac{1}{1 - (4\beta^2 T)^2} . \]  

(1.21)
It is not as spectral efficient as the sinc pulse (1.19) since it uses a factor $1 + \beta$ wider frequency spectrum, but for $\beta = 0$ it reduces to the sinc pulse. This pulse is however not a Nyquist pulse but sampled using a matched filter (see further below) the whole transfer function becomes a Nyquist pulse/filter namely the raised-cosine. The Fourier transform (frequency response) of the root-raised-cosine is given by:

$$G_{T,\beta}(\omega) = \begin{cases} \sqrt{T} & 0 \leq |\omega| \leq (1 - \beta) \frac{\pi}{T} \\ \sqrt{\frac{T}{2}} \sqrt{1 - \sin \left( \frac{T}{2\beta} \left( |\omega| - \frac{\pi}{T} \right) \right)} & (1 - \beta) \frac{\pi}{T} < |\omega| \leq (1 + \beta) \frac{\pi}{T} \\ 0 & \text{otherwise.} \end{cases} \quad (1.22)$$

The root-raised-cosine is defined for $\beta \in [0, 1]$. 

**Modulation techniques**

Pulse Amplitude Modulation, (PAM) is a modulation technique where the information is stored in the amplitude of a carrier pulse. It can be used for both baseband or passband signaling, where the former is using frequencies around 0 and up to some cut-off frequency $f_{\text{sig}}$ and the latter is using frequencies in the interval $f_c - f_{\text{sig}}/2$ to $f_c + f_{\text{sig}}/2$, where $f_c$ is the frequency of a carrier wave. We are going to look at it as blocks of data being transmitted. A block of length $N$ is described as

$$s(t) = \sum_{n=1}^{N} a_n g(t - n \cdot T) \quad (1.23)$$

for the baseband and $\sum_{n=1}^{N} a_n g(t - n \cdot T) \cdot \cos(2\pi f_c t)$ for the passband case. Thus, the information is carried in the variable amplitude $a_n$, where $a_n$ can be chosen from different alphabets depending on the system, e.g. $a_n = \pm 1$ or $a_n = \pm 1, \pm 2$ etc. This model may be accused of being non-causal, since a pulse coming later in time is allowed to affect the signal at times before (one may argue that if $m < n$ then the bit sent at time $nT$ should not affect the bit sent at time $mT$ since that bit is sent before). This can on the other hand be defended by arguing that in order to get a continuous signal one might wait for a set of bits and then generate the full signal $s(t)$ according to the formula before transmitting it. This set of bits is going to be called a block (compare with block codes in Definition 6).

Nyquist observed that using the passband, or carrier wave, one could increase the amount of data sent by also using a sinusoidal carrier wave [13]. This technique is called Quadrature Amplitude Modulation (QAM) and the modulation is given
by [8]:

\[ s(t) = \sum_{n=1}^{N} a_n g(t - n \cdot T) \cos(2\pi f_c t) - \sum_{n=1}^{N} b_n g(t - n \cdot T) \sin(2\pi f_c t) \]

\[ = \Re \left\{ \sum_{n=1}^{N} (a_n + ib_n) g(t - n \cdot T)e^{i2\pi f_c t} \right\}. \]

The success of this modulation technique is based on the fact that the carrier waves are "almost orthogonal" over the symbol time \( T \) or even "more orthogonal" over a block period \( N \cdot T \) [8, (5.9) and (5.10)], thus at the receiver end it is possible to consider both terms separately, provided that the pulse shape \( g(t) \) preserve the orthogonality (note that the sinc pulse is such). For simplicity only baseband PAM will be considered, since other modulation techniques like QAM are usually straight forward extensions but could possibly cloud the theory.

The AWGN channel

In communication a signal is then sent over a channel, which in reality often is an electric pulse in a cable or as an electromagnetic wave. This means that the signal sent by the sender, is not the signal that will be received by the receiver, and this distortion needs to be modeled as well. The simplest and one of the most commonly used channel models is the AWGN channel, Additive White Gaussian Noise, which means that a noise \( w(t) \) is added to the sent signal:

\[ r(t) = s(t) + w(t) . \quad (1.24) \]

Here \( w(t) \) is a white Gaussian noise, meaning that it is a Gaussian time-continuous stochastic process that is stationary, zero-mean, and has constant two-sided power spectral density\(^6\) \( N_0/2 \). This in turns poses some technical difficulties and different approaches are discussed in [10,14] respectively. Here the Dirac-delta notion will be formally used. A motivation why white noise is a useful model is that the encountered noise need only to behave white within the frequencies of interest. Projecting this noise on an orthogonal basis of the signal space the samples are themselves independent identically distributed (i.i.d.), zero-mean, Gaussian variables, a property we shall see is very convenient in constructing a receiver.

The structure of the channel, together with the sending and receiving, is displayed in Figure 1.1. This notation will be used through the thesis.

The receiver: Matched filter and sampled noise

On receiving the time continuous signal \( r(t) \) one must somehow get relevant sampled data from it. This is typically done by using a matched filter, i.e. computing the

\(^6\)This is counting both positive and negative frequencies, for an example of negative frequencies see that calculations following directly after (1.17).
inner product,
\[ y_n = \langle r(t), g_T(t - n \cdot T) \rangle, \quad (1.25) \]
for all different \( n \) used in the sending (see theorem 5.8.2 in [10]). This vector of discrete points provide sufficient statistics for the estimation, meaning that no more information is needed in order to estimate the transmitted message optimally [8,10].

When using shift orthogonal pulses the receiver structure becomes fairly simple. Since the pulses carrying the information are orthogonal, they are in some sense invisible to each other. Using what we know about the received signal \( r(t) \) from (1.23) and (1.24), we get
\[
y_n = \langle r(t), g_T(t - n \cdot T) \rangle \\
= \langle s(t), g_T(t - n \cdot T) \rangle + \langle w(t), g_T(t - n \cdot T) \rangle \\
= \sum m a_m \langle g_T(t - m \cdot T), g_T(t - n \cdot T) \rangle + \eta_n \\
= \{ \text{ orthogonality } \} \\
= a_n + \eta_n.
\]
In this case \( a_n \) is the sought amplitude carrying the information and \( \eta_n \) is a random Gaussian variable independent of each other and with variance \( N_0/2 \), that is \( \eta \in \mathcal{N}(0, N_0/2 \cdot I) \). Presenting it on vector form for the whole block of size \( N \) we have
\[ y = a + \eta \quad (1.26) \]
where \( \eta \) is the noise vector (i.i.d. Gaussian), and \( y \) is the received statistics.

The above model is valid for unit energy pulses. In practical situations however one might want to increase the signal power; this will be visible as a factor \( \sqrt{P} \) in front of the symbols \( a \). Rearranging (1.26) we can arrive at:
\[ y = a + \sigma \hat{\eta} \quad (1.27) \]
where \( \hat{\eta} \in \mathcal{N}(0, I) \) and \( \sigma = \sqrt{N_0/2P} \). We see that the exact values of \( P \) and \( N_0 \) are not relevant, only the quotient since that will describe the standard deviation.
of the sampled noise, and hence serve as a measure of the corrupting strength of the noise. The variance $\sigma^2$ is the inverse of the linear signal-to-noise-ratio (SNR). This is often measured in decibel (dB) and we relate the SNR measured in dB to $\sigma$ as

$$\sigma = \sqrt{\frac{N_0}{2P}} = 10^{-\frac{\text{SNR}}{20}}.$$  \hspace{0.5cm} (1.28)

Thus a high SNR\(^7\) gives a low standard deviation of the sampled noise implying that the samples are closer to their actual intended value and hence the estimation becomes more accurate. For the ease of calculations we are going to continue to look at unit energy pulses and the actual energy will be incorporated in the $\sigma$-expression.

Based on the previous discussion we will from here on regard the autocovariance of the added Gaussian process be $\sigma^2 \cdot \delta(t - s)$ and only work with unit-energy pulses, actually that is not entirely true; in FTN we will work with pulses that has energy $\rho$ but that will be visible elsewhere and hence the SNR will always be given by (1.28).

### Maximum likelihood estimation

We are here going to assume that all input characters are equiprobable and that the alphabet is of finite size. From the data $y$ we want to find an estimate $x$ to the sent data vector $a$ in such a way that the probability of error is minimized, that is given the received signal $r(t)$ we want to find the argument $x$ such that it minimizes $\min_x P(x \neq a \mid r(t))$. Now since

$$P(x \neq a \mid r(t)) = 1 - P(x = a \mid r(t))$$

this is equivalent to finding the maximizing argument for

$$\max_x P(x = a \mid r(t)).$$

This in turn is done by choosing an estimate $x$ such that the probability $P(a \mid r(t))$ is maximized \(^8\). Modifying the expression further we see that $P(a \mid r(t)) = P(a \mid y, w(t))$ where $w(t)$ is a component of the noise that is orthogonal to the space span\{\(g^*_n(t), n \in \mathbb{Z}\}\} on which we have projected the received signal $r(t)$\(^9\). The orthogonality also means that it is probabilistically independent of

\(^7\)We will always talk about SNR in dB unless otherwise explicitly stated.

\(^8\)Taking $x = \arg \max_x P(a \mid r(t))$. Or in words it means that the probability that our estimate $x$ is the sent data $a$, given the received signal; is maximized if we choose an estimate that maximizes the probability that the data $a$ was sent given the received signal.

\(^9\)Remember that the noise in the AWGN channel was not necessarily band limited, neither does it need to be in $L^2(-\infty, \infty)$, and hence it may have components not reflected by a projection on the base.
1.2. SIGNALS AND COMMUNICATION

the projected components [8]. Hence \( P(a \mid y, w \perp (t)) = P(a \mid y) \) and by Bayes rule
\[
P(a \mid y) = \frac{P(y \mid a)P(a)}{P(y)} ,
\]
thus we get:
\[
x = \arg \max_a P(a \mid r(t)) \tag{1.29}
\]
\[
= \arg \max_a \frac{P(y \mid a) \cdot P(a)}{P(y)}
\]
\[
= \{\text{with equiprobable input distribution } P(a), \text{ and } P(y) \text{ independent of } a\}
\]
\[
= \arg \max_a P(y \mid a) . \tag{1.30}
\]

This means that the maximum likelihood estimation (ML estimation) is the estimation that results in the smallest error probability [8]. The probability \( P(y \mid a) \) is normally referred to as the likelihood, hence the name of the estimation.

For a given \( a \) we have that \( a + \eta \) is a i.i.d. Gaussian vector with distribution \( \mathcal{N}(a, \sigma \cdot I) \) and thus the likelihood for \( y \), given \( a \) is [7]:
\[
P(y \mid a) = \left( \frac{1}{2\pi} \right)^{N/2} \cdot \frac{1}{\sqrt{\det[\sigma^2 \cdot I]}} \cdot e^{-\frac{1}{2}(y-a)^T(\sigma^2 \cdot I)^{-1}(y-a)}
\]
\[
= \left( \frac{1}{2\sigma^2 \pi} \right)^{N/2} \cdot e^{-\frac{1}{2\sigma^2}(y-a)^T(y-a)} = \left( \frac{1}{2\sigma^2 \pi} \right)^{N/2} \cdot e^{-\frac{1}{2\sigma^2} \sum_{n=1}^{N}(y_n-a_n)^2} .
\]

As mentioned \( N \) is the number of pulses sent in the signal (and thus the number of information carrying amplitudes \( a_n \) we want to estimate). Looking at the expression we can see that the vector \( a \) that maximizes the expression is the one with smallest Euclidean distance to the sample vector \( y \). Here we are working with an orthogonal basis and thus we can consider the terms \( y_n - a_n \) independently. Thus the vector \( a \) that maximizes the expression is the one that in which each component is closest to \( y \):
\[
x_n = \min_{a_n} (y_n - a_n)^2 \tag{1.31}
\]

Now we strengthen our assumption to regard only a binary alphabet, \( a \) being \( \pm 1 \). In light of (1.26) one can intuitively believe that since the sampled noise is independent and zero-mean the best estimation of the sent data is achieved by just looking at the sign of \( y \). We are indeed going to prove this by observing that for the binary case the exponent-minimizer is the \( a \) with the same sign as \( y \) in each component. We thus end up with the estimator
\[
x = \text{sign}(y) \tag{1.32}
\]

for the case of binary modulated base band PAM sent over an AWGN channel, where the received data \( y \) is the vector given by (1.25). Even if this is well known this derivation will prove useful later.
1.3 Matrix theory

This section will contain some theory about matrices, special structured matrices, and how these can be treated. The term vector will be used in a general sense, it is not necessarily a vector in \( \mathbb{R}^N \) but can be anything defined in a Hilbert space. For a quick definition see [15, pp. 298-300].

Gram matrices

**Definition 13** (Gram Matrix). A Gram matrix is an \( N \times N \) hermitian matrix \( G \) whose elements are all the inner products of a set of vectors \( v_1, v_2, \ldots, v_N \) of a Hilbert space.

\[
G = \begin{pmatrix}
\langle v_1, v_1 \rangle & \langle v_1, v_2 \rangle & \cdots & \langle v_1, v_N \rangle \\
\langle v_2, v_1 \rangle & \langle v_2, v_2 \rangle & \cdots & \langle v_2, v_N \rangle \\
\vdots & \vdots & \ddots & \vdots \\
\langle v_N, v_1 \rangle & \langle v_N, v_2 \rangle & \cdots & \langle v_N, v_N \rangle
\end{pmatrix}
\] (1.33)

The vectors \( v_1, v_2, \ldots, v_N \) are said to define the Gram matrix.

The following two theorems state some important properties of the Gram matrix and how it relates to the vectors defining it [16, 17].

**Theorem 5** (Singular Gram matrices). A Gram matrix is singular if and only if the vectors defining it are linearly dependent.

Once again observe that the term vector is here used in a general sense, they belong to some Hilbert space, for example \( L^2(-\infty, \infty) \).

**Theorem 6** (Positive semidefinite matrices and Gram matrices). A Gram matrix is positive semidefinite, and conversely any positive semidefinite matrix is a Gram matrix from some defining vectors.

This needs not to be a unique representation, but only claims the existence of some defining vectors in some Hilbert space.

Toepplitz matrices

The next type of matrix is the so called Toeplitz matrix.

**Definition 14** (Toeplitz matrix). An \( N \times N \) matrix \( G \) is called Toeplitz if it is constant along the diagonals, namely

\[
G = \begin{pmatrix}
g_0 & g_{-1} & g_{-2} & \cdots & g_{-N+1} \\
g_1 & g_0 & g_{-1} & \cdots & g_{-N+2} \\
g_2 & g_1 & g_0 & \cdots & g_{-N+3} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
g_{N-1} & g_{N-2} & g_{N-3} & \cdots & g_0
\end{pmatrix}
\] (1.34)
1.3. MATRIX THEORY

which is equivalent to \( G_{i,j} = G_{i-j} \).

For Toeplitz matrices there are some theory readily available, like Szegö’s theorem [18–20].

**Theorem 7** (Szegö’s theorem). Let \( G \) be a Toeplitz matrix of size \( N \times N \) and let its eigenvalues \( \lambda_j^N \), \( j = 1, 2, \ldots, N \), be sorted in descending order. \( G \) is then such that \( G_{m,n} = c_{m-n} \) and where \( c_k \) is the \( k \)th Fourier series coefficient of \( f(\theta) \), a real valued function in \( L^\infty(-\pi, \pi) \). Then

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{j=1}^{N} F(\lambda_j^N) = \frac{1}{2\pi} \int_{-\pi}^{\pi} F(f(\theta)) \, d\theta
\]

(1.35)

for any function \( F \) that is continuous on the interval \([\inf f(\theta), \sup f(\theta)]\).

The Fourier series coefficients are the standard exponential coefficients given as \( c_k = 1/2\pi \int_{-\pi}^{\pi} f(\theta) e^{ik\theta} \, d\theta \). Transformations can be done to sin/cos-series. The function \( f(\theta) \) as described in Theorem 7 is going to be called the Szegö function and describes what is usually know as spectral properties of the matrix. This theorem has a very useful corollary [20].

**Corollary 1** (Eigenvalue distribution). Let \( G \), \( \lambda_j^N \) and \( f(\theta) \) be as in Theorem 7, then \( \{\lambda_j^N\}_{j=1}^{N} \) and \( \{f(2\pi j/N)\}_{j=1}^{N} \) are asymptotically equally distributed.

The following theorem is related to Corollary 1 and gives bounds and convergence for the eigenvalues them selves.

**Theorem 8** (Eigenvalues of Toeplitz matrices). Let \( G \), \( \lambda_j^N \), and \( f \) be as in Theorem 7, then for the eigenvalues it holds that for any fixed \( j \geq 1 \),

\[
\inf f(\theta) \leq \lambda_j^N \leq \sup f(\theta) \quad \text{for any fixed } j \geq 1
\]

(1.36)

\[
\lim_{N \to \infty} \lambda_j^N \to \sup f(\theta) \quad \text{for any fixed } j \geq 1
\]

(1.37)

\[
\lim_{N \to \infty} \lambda_{N+1-j}^N \to \inf f(\theta)
\]

(1.38)

**Matrix factorizations**

A matrix factorization/decomposition is a way to write a matrix as a product of other matrices, with special structure. The first factorization to be presented is the singular value decomposition, (SVD). This is a decomposition applicable to all matrices but that show special properties depending on the properties of the matrix. The SVD is described in the next theorem [21].

**Theorem 9** (Singular values decomposition). Let \( A \) be any (rectangular) matrix, that is \( A \in \mathbb{C}^{N \times M} \) then

\[
A = UDV^*
\]

(1.39)
where $U$ and $V$ are unitary square matrices and called the left singular and right singular vectors respectively. $D$ is a real, non-negative, rectangular, diagonal matrix with the singular values on the diagonal. The singular values are the square root of the eigenvalues to $A^*A$.

Moreover the rank of $A$ is the number of non-zero singular values.

**Remark.** If $A$ is a real square and symmetric matrix then the singular values are the absolute values of the eigenvalues, moreover the matrices $U$ and $V$ will be square, real, orthogonal, and related as $U = V$.

Another important factorization is the Cholesky factorization. It is applicable to positive definite matrices and useful in solving linear equation systems efficiently and stable. The following theorem describes the factorization [21,22].

**Theorem 10** (Cholesky factorization). Let $A$ be a symmetric, square, and positive definite matrix (that is has only positive eigenvalues), then

$$A = R^T R$$

where $R$ is an upper triangular matrix. These matrices can be efficiently computed.

**Remark.** The above equation can of course also be written $A = LL^T$ where $L$ is a lower triangular matrix. It is just a matter of transposing the above equation, remembering that $A$ is symmetric. (The same hold when $A$ is Hermitian but then all transposes are changed to transpose-conjugate).

### The square root of a matrix

To take the square root of a matrix is something that will provide useful, but in order to do so we must first define what we mean by it.

**Definition 15** (The square root of a matrix). The square root of a $N \times N$ matrix $A$, denoted either $\sqrt{A}$, or $A^{1/2}$, is a matrix $B$ such that $BB = A$.

First of all observe that the square root of may not exist for all matrices, and for those where it exists it is not uniquely defined. For the case where $A$ is real, square and symmetric we can rely on the SVD and calculate the square root as $G^{1/2} = US^{1/2}U^T$ where $S^{1/2}$ is just the diagonal matrix $S$ but with the square root of all elements on the diagonal. Here we see that regardless if we choose the positive or negative square root of the elements in $S$ we get a square root of $G$. This way of taking the square root hence allows us to compute $2^N$ different square roots of $A$ and furthermore produces symmetric matrices $B$. 
Chapter 2

Motivation and related research

With the preliminaries done we can start focusing on the actual content of this thesis, starting off with a motivation and the background of what has previously been done.

2.1 Idea, motivations, and challenges with FTN

The choice of delay/intermediate time between pulses in today’s communication system is often decided so that the pulses are orthogonal to each other. This is convenient since, as we have seen in Chapter 1, detection and estimation becomes particularly easy at the receiver end. However, combined with the fact that the pulses used today are not spectrally optimal in frequency and that frequency is a limited and expensive resource there seems to be room for improvement. In today’s 3GPP standard for 3G networks the system uses the root-raised-cosine pulse with a roll-off factor $\beta = 0.22$ [12]. This is a 22% frequency leakage since when this sent at the rate at which is orthogonal, it is able to carry the same amount of information that the optimal pulse carries, but at the cost of using 22% more frequency. The mathematically optimal pulse is however not realizable. Faster-than-Nyquist signaling (FTN) is an attempt to pack pulses tighter in time than previously done and increasing the data rates by decreasing the time between the sent bits. The idea is simple, instead of waiting a time $T$ before sending the next pulse one waits $\rho T$ where $\rho < 1$. A mathematical formulation can be found in (2.1) and (2.2) and this approach would give an increase of data carrying pulses in a signal by $(1 - \rho)/\rho$. Realizing this in practice is however not trivial since it comes with problems as inter-symbol interference and there are today no efficient algorithms to counteract it.

There is another gain by this approach namely to increase the speed of communication. It may seem connected to the first but it might not always be that simple. To provide data rates of an average of 1 Mb/s over a minute does not mean that the system can provide $1/60$ Mb/s on average over every second; this is because
existing techniques (crucial for the communication such as block coding) introduces delays in the system. With the FTN approach however an increase in data rates would likewise result an increase in communication speed which is important in time critical applications where delays cannot be tolerated.

2.2 Background, Saltzberg and Mazo

The first paper, to my knowledge, to study faster-than-Nyquist signaling is [23] by B. Saltzberg from 1968. This paper looks at the error probabilities for a linear detector sampling at the sending rate and in the end of this paper Saltzberg is looking at the error probabilities when this is applied to FTN; actually he takes the other approach than usual today, and instead of packing pulses tighter he narrows the bandwidth of the sinc pulse used. The result is discouraging but the study is however not complete since in this case instead of utilizing the structure of the ISI (see Chapter 3, the same structural arguments are possible from Saltzberg’s point of view), the ISI is rather seen as an independent noise itself adding on top of the AWGN noise.

In 1975 Mazo looked at what happens with the the minimum Euclidean distance between pairs of Nyquist trains if these where sent with less than $T$ seconds apart [24]. He was interested in seeing if excess performance in terms of error probabilities could be traded for higher data rates by signaling faster-than-Nyquist while keeping the power of the sender constant. This inevitably introduces inter-symbol interference, but remarkably it seemed like the minimum distance was still given by the pulse energy for some what higher rates than the Nyquist rate. Mazo conjectured that this was the case but didn’t prove it then. It was further studied by Mazo & Landau [25] and Hajela [26] and they independently proved that Mazo’s conjecture was right. This is interesting since the minimum Euclidean distance is a bound on the error rate [8, 24].

Specifically Mazo looked at the sinc signal (1.17) sent as a baseband, PAM modulated pulse train, $a_n \in \{-1, 1\}$

$$s(t) = \sum_{n=N_1}^{N_2} a_n g_T(t - n \cdot T), \quad (2.1)$$

for arbitrary $N_1$ and $N_2 \geq N_1$, and studied what happened when it was sent faster, as

$$s(t) = \sum_{n=N_1}^{N_2} a_n g_T(t - n \cdot \rho T) \quad (2.2)$$

where $\rho < 1$. Then the signals $g_T(t)$ are no longer orthogonal and inter-symbol interference is introduced. However the minimum Euclidean distance, $\min ||s_1(t) - s_2(t)||_2$, is for this kind of pulse train is unaffected (in the sense that it is still given as the pulse energy) for $\rho \in [0.802, 1]$ which was proven in [25, 26]. The minimum
2.2. BACKGROUND, SALTZBERG AND MAZO

\( \rho \) for which the Euclidean distance is unaffected has come to be called the Mazo Limit.

**Derivation of the Mazo limit**

To fully understand the result of the proofs we will now derive the equation for the minimum distance \( [24, \text{see equation (15)}] \) that was used. We start, as Mazo, with two different signals of the type (2.2) with \( a_n = \pm 1 \) and \( g_T(t) \) as the sinc in (1.17).

\[
\begin{align*}
\rho_{\text{min}}^2 &= \min_{s_1(t) \neq s_2(t)} (s_1(t) - s_2(t), s_1(t) - s_2(t)) \\
&= \min_{s_1(t) \neq s_2(t)} \int_{-\infty}^{\infty} |s_1(t) - s_2(t)|^2 \, dt \\
&= \left\{ b_n = a_n^{(1)} - a_n^{(2)} \right\} = \min_{b_n \neq 0} \int_{-\infty}^{\infty} \left| \sum_{n=N_1}^{N_2} b_n g_T(t - n\rho T) \right|^2 \, dt \\
&= \{ \text{Fourier transform, Plancherel's formulae} \} \\
&= \min_{b_n \neq 0} \frac{1}{2\pi} \int_{-\infty}^{\infty} \left| \sum_{n=N_1}^{N_2} b_n G_T(\omega) \cdot e^{-i\omega n\rho T} \right|^2 \, d\omega \\
&= \{ G_T(\omega) = T \cdot \text{rect}[-\pi/T, \pi/T](\omega) \} \\
&= \min_{b_n \neq 0} \frac{1}{2\pi} \int_{-\infty}^{\infty} \left| \sum_{n=N_1}^{N_2} b_n \cdot T \cdot \text{rect}[-\pi/T, \pi/T](\omega) \cdot e^{-i\omega n\rho T} \right|^2 \, d\omega \\
&= \min_{b_n \neq 0} \frac{T^2}{2\pi} \int_{-\pi/T}^{\pi/T} \left| \sum_{n=N_1}^{N_2} b_n \cdot e^{-i\omega n\rho T} \right|^2 \, d\omega = \{ \theta = T\omega \rho \} \\
&= \min_{c_n \neq 0} \frac{2^2 T}{2\pi T} \int_{-\rho\pi}^{\rho\pi} \left| \sum_{n=N_1}^{N_2} c_n \cdot e^{-i\theta n} \right|^2 \, d\theta
\end{align*}
\]

Where the last \( 2^2 \) comes from the transformation \( c_n = b_n/2 \). However from the previous encounter with the sinc function we know that the numerator \( 2^2 T \) is nothing more than \( 4E \), where \( E \) is the energy of \( g_T(t) \). Now let \( l \) be the first index such that \( c_l \neq 0 \) and let \( m \) be the last index such that \( c_m \neq 0 \), then we can without loss of generality assume \( c_m = 1 \) since otherwise we could multiply with \(-1\) and it would make no difference because of the absolute value. We can now also rescale the exponential series by multiplying with \( e^{i\theta m} \) since the absolute value of this is
1. Moreover we let $K = m - l$ and thus we get:

$$\min_{c \neq 0} \frac{4E}{2\pi \rho} \left| \sum_{n=N_1}^{N_2} c_n \cdot e^{-i\theta n} \right|^2 d\theta$$

$$= \min_{K \geq 0} \frac{4E}{2\pi \rho} \left| \sum_{n=0}^{K} c_n \cdot e^{i\theta n} \right|^2 d\theta$$

$$= \min_{K \geq 0} \frac{4E}{2\pi \rho} \left| \sum_{n=0}^{K} c_n \cdot e^{i\theta n} \right|^2 d\theta$$

where the sign in front of the summation character is just a matter of notation. Rearranging and we get the same equation as Mazo [24, equation (15)]

$$\frac{d^2_{\text{min}}}{4E} = \min_{K \geq 0} \frac{1}{2\pi \rho} \left| \sum_{n=1}^{K} c_n \cdot e^{i\theta n} \right|^2 d\theta$$  (2.3)

**Remark.** Note that the energy $E$ in the denominator is the pulse energy, and in order to keep the power at the sender end constant this is an energy actually decreasing linearly with $\rho$. Hence the minimum Euclidean distance will drop when sending FTN subject to constant power, however the reason is due to the normalization of pulse energy needed to compensate for the increased number of pulses and not due to the ISI.

In the proofs of Mazo & Landau and Hajela it is also shown that the signals attaining the minimum distance 1 at $\rho = 0.802$, and then lowering it for $\rho < 0.802$ differ from each other as

$$R(\theta) = 1$$  (2.4)

or

$$R(\theta) = \sum_{n=0}^{7} (-1)^n \cdot e^{i\theta n}.$$  (2.5)

This is remarkable since for $\rho > 0.802$, the signal attaining the minimum distance a way are those who only differs in one bit. With full understanding of the derivation of (2.3) we can understand that this means that the signals a minimum distance apart, at $\rho = 0.802$, are differing in exactly 1 bit or 8 consecutive bits (in a very specific pattern), regardless of the length of the signals. The only criteria on the
length of the regarded signals is the trivial one, $N_2 \geq N_1$, and of course in order to be able to have an 8 bit error pattern it must be at least 8 bits long. Then for some $\epsilon > 0$ and $\rho$ in between $0.802 - \epsilon < \rho < 0.802$ it is (2.5) that defines the difference between a pair of closest signals.¹

2.3 Related research

After the Mazo limit was proven in 1988 [25, 26] the research has taken of in different places and different directions; soon afterwards there were claims of realizable transmission below the Nyquist limit [27]. At the turn of the millennium the research intensified further and one direction was the search for similar results for other types of pulses. The sinc pulse is not practically implementable and thus the family of raised cosine pulses was investigated numerically [28]. At the same time this research took of in Lund [9, 29]. The directions has been different and one particularly interesting is the extension of the Mazo limit into the frequency domain [29], where the signals are also packed tighter in the frequency domain. This introduces interference between previously uncorrelated channels, but this interchannel interference (ICI) does not affect the reliability of the channels and with the use of an optimal detector the error rates should remain the same much like the ISI for time domain packing FTN. The extensive research (foremost in Lund) has lately been summed up in [9] and there seems to be general agreement that FTN is a promising technique [30]. However, the algorithms used for detection and estimation suffers from high complexity, limiting the practical usage of FTN.

¹This follows directly from the definition of $\nu$ and assertion 3) in theorem 1.1 [26]
Chapter 3

Receiving and detecting FTN signals

When sending shift orthogonal pulses at the rate of orthogonality we can use that orthogonality to efficiently receive the data since the bits are in some sense independently transferred, as seen in Chapter 1. When sending faster-than-Nyquist we arrive at the problem that the bits are no longer carried by orthogonal pulses. To start with we are going to consider the normalized sinc pulse, from (1.19). For brevity we denote

\[ g_n^\rho(t) = \sqrt{\rho} \cdot g_T(t - n \cdot \rho T), \]  

(3.1)

and the extra factor \( \sqrt{\rho} \) is described below.

Energy normalization

Since faster-than-Nyquist signaling (FTN) is about packing the pulses \( g_T(t) \) tighter in time, this would require the sender to increase the power. Doing so one could instead choose to increase the alphabet and in order to get a fair comparison of the FTN we normalize all pulses at the sender end with a factor \( \sqrt{\rho} \), thus lowering the pulse energy with a factor \( \rho \) and keeping the power at the send end constant. The corresponding matched filter is also adjusted there after. This might not always be explicitly written (especially not in the calculations) since it is only a linear scaling, but when these factors matters they will be included and this will be followed by a comment.
CHAPTER 3. RECEIVING AND DETECTING FTN SIGNALS

3.1 Premises for Faster-Than-Nyquist Signaling

At the receiver we want a set of sufficient statistics for the estimation. To get this we apply a matched filter, similar to the Nyquist case (1.25), and get

\[ y_n = \langle r(t), g^n_T(t) \rangle = \sum_{m=1}^{N} a_m \cdot \langle g^m_T(t), g^n_T(t) \rangle + \langle w(t), g^n_T(t) \rangle , \quad n = 1, \ldots, N \]

which in turn, looking at a complete block, gives rise to the system

\[ y = Ga + w , \tag{3.2} \]

where

\[ G_{m,n} = \langle g^m_T(t), g^n_T(t) \rangle \]

\[ = \int_{-\infty}^{\infty} g^m_T(t) \cdot \overline{g^n_T(t)} \, dt = \{ \text{Plancherel's formulae} \} \]

\[ = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega m \rho T} G_T(\omega) \cdot \overline{e^{-i\omega n \rho T} G_T(\omega)} \, d\omega \]

\[ = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega (m-n) \rho T} |G_T(\omega)|^2 \, d\omega \]

\[ = \{ \text{Fourier transform is given by (1.20)} \} = \{ \theta = \omega \rho T \} \]

\[ = \frac{1}{2\pi \rho} \int_{-\rho \pi}^{\rho \pi} e^{-i\theta (m-n)} \, d\theta = \frac{1}{\pi \rho (m-n)} \sin (\rho \pi (m-n)) , \]

and thus\(^1\)

\[ G_{m,n} = \rho \text{sinc}(\rho (m-n)) . \tag{3.3} \]

We can see that \(G\) is a Toeplitz, Gram matrix. In the orthogonal Nyquist case \(G\) was a diagonal matrix and the noise was an i.i.d. Gaussian distributed vector; the optimal ML-estimation \(x\) was then derived rather straightforward and the decision rule simple, see (1.32). In the FTN case \(G\) is no longer diagonal and we have introduced ISI since the measurements are no longer only depending on one of the signaled symbols. Regarding the sampled noise we still have that \(w\) is 0 mean

\(^1\)The factor \(\rho\) comes from the energy norming as mentioned above.
Gaussian vector, but looking at the covariance matrix we see that:

\[
\text{Cov}(w_m, w_n) = E[(w_m - E[w_m]) \cdot (w_n - E[w_n])] = E[w_m \cdot w_n]
\]

\[
= E\left[\int_{-\infty}^{\infty} w(t) \cdot g_T^m(t) \, dt \cdot \int_{-\infty}^{\infty} w(s) \cdot g_T^n(s) \, ds\right]
\]

\[
= E\left[\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} w(t) \cdot g_T^m(t) \cdot w(s) \cdot g_T^n(s) \, dt \, ds\right]
\]

\[
= \left\{\text{integrate with respect to } s, \ E[w(t) \cdot w(s)] = \sigma^2 \cdot \delta(s - t)\right\}
\]

\[
= \sigma^2 \cdot \int_{-\infty}^{\infty} g_T^m(t) \cdot g_T^n(t) \, dt
\]

\[
= \sigma^2 \cdot G_{m,n}.
\]

(3.4)

Hence the covariance matrix is \(G\), which is not a diagonal matrix, and the components of the random vector \(w\) are dependent.\(^2\)

In order to continue we have to derive some properties of \(G\), starting off at an important property of the sinc pulses.

**Theorem 11** (Linear independence of sinc pulses). *The set of sinc pulses

\[
\left\{g_T^k(t)\right\}_{k=1}^{N} = \left\{\frac{\sin \left(\frac{\pi}{T} \cdot (t - k \cdot \rho T)\right)}{\frac{\pi}{\sqrt{T}} \cdot (t - k \cdot \rho T)}\right\}_{k=1}^{N}
\]

are linearly independent \(\forall \rho > 0\) and \(T > 0\).

*Proof.* This is a proof by contradiction, and we start by assuming that the sinc pulses are linearly dependent, that is

\[
f(t) := \sum_{k=1}^{N} a_k \cdot g_T^k(t) = 0 \quad \forall t \text{ and some } a_k \neq 0.
\]

\(f(t)\) is thus the zero function and we know from uniqueness that its Fourier transform must be identically zero as well, hence

\[
F(\omega) = \sum_{k=1}^{N} a_k \sqrt{T} \cdot \text{rect}[\omega / T, \pi / T](\omega) \cdot e^{-i\omega k \rho T} = 0 \quad \forall \omega \text{ and some } a_k \neq 0.
\]

\(^2\)Later on this has been found to agree well with what is found in chapter 28 in [10]. Looking at proposition 28.2.1 we also see that this is sufficient statistics.
Simplifying by dividing with $\sqrt{T}$, realizing that this is indeed zero outside the interval $\omega \in [-\pi/T, \pi/T]$, and changing the variable to $\theta = \omega \rho T$ we get the equivalent formulation

$$\sum_{k=1}^{N} a_k \cdot e^{-i\theta k} = 0 \quad \forall \theta \in [-\rho \pi, \rho \pi] \text{ and some } a_k \neq 0 .$$

Now since the complex exponentials are infinitely differentiable we know that this must also hold for all derivatives up to order $N - 1$, yielding the system

$$
\begin{pmatrix}
  e^{-i\theta} & e^{-2i\theta} & \cdots & e^{-Ni\theta} \\
  -i \cdot e^{-i\theta} & -2i \cdot e^{-2i\theta} & \cdots & -Ni \cdot e^{-Ni\theta} \\
  -1 \cdot e^{-i\theta} & -4 \cdot e^{-2i\theta} & \cdots & -N^2 \cdot e^{-Ni\theta} \\
  i \cdot e^{-i\theta} & 8i \cdot e^{-2i\theta} & \cdots & N^3 i \cdot e^{-Ni\theta} \\
  \vdots & \vdots & \ddots & \vdots \\
  (-i)^{N-1} \cdot e^{-i\theta} & (-2i)^{N-1} \cdot e^{-2i\theta} & \cdots & (-Ni)^{N-1} \cdot e^{-Ni\theta}
\end{pmatrix}
\begin{pmatrix}
a_1 \\
a_2 \\
a_3 \\
a_4 \\
\vdots \\
a_N
\end{pmatrix}
= 0 ,
$$

which has a solution $a \neq 0$ according to our assumption of linear dependence. From this we can conclude that the system matrix must be singular which in turn implies that the rows must be linearly dependent. Writing this out explicitly as

$$b_0 \cdot 
\begin{pmatrix}
  1 \cdot e^{-i\theta} \\
  1 \cdot e^{-2i\theta} \\
  1 \cdot e^{-3i\theta} \\
  \vdots \\
  1 \cdot e^{-Ni\theta}
\end{pmatrix}
+ \cdots + b_{N-1} \cdot 
\begin{pmatrix}
  (-i)^{N-1} \cdot e^{-i\theta} \\
  (-2i)^{N-1} \cdot e^{-2i\theta} \\
  (-3i)^{N-1} \cdot e^{-3i\theta} \\
  \vdots \\
  (-Ni)^{N-1} \cdot e^{-Ni\theta}
\end{pmatrix}
= 0 ,
$$

one can note that in row $k$ (in the new system) all entries are multiplied with the factor $e^{-i\theta k}$, in fact all rows are on the form $p(-ki) \cdot e^{-i\theta k} = 0$ where $p(x)$ is a polynomial given by the coefficients for linear dependence $\{b_j\}_{j=0}^{N-1}$, for the second matrix. Since $e^{-i\theta k} \neq 0$ this claims the existence of a polynomial

$$p(x) = \sum_{k=0}^{N-1} b_k \cdot x^k$$

with the $N$ roots $-i, -2i, \ldots, -Ni$. But $p(x)$ is only of degree $N - 1$ and hence this contradicts the fundamental theorem of algebra.

Consequently our system matrix is not singular, thus all coefficients $a$ are required to be 0 contradicting the assumption that some $a_k \neq 0$. Thus, our assumption that $\{g_T^k(t)\}_{k=1}^{N}$ are linearly dependent does not hold.

**Remark.** The above theorem holds for all finite $N \geq 1$.

An important notice is that the same holds even if the information carrying pulses are root raised cosine pulses as in (1.21).

26  

CHAPTER 3. RECEIVING AND DETECTING FTN SIGNALS
Corollary 2 (Linear independence of root raised cosine pulses). The set of root raised cosine pulses $\{g^k_{T,\beta}(t)\}_{k=1}^{N}$ as described in (1.21) are linearly independent $\forall \rho > 0, \beta \in [0,1)$ and $T > 0$.

Proof. One can use the same arguments as in the proof of Theorem 11, the only difference being that the Fourier transform of the root raised cosine pulse is more complicated, see (1.22). That however does not provide any substantial difference since one can instead, after taking the Fourier transform of the zero function as above, look at the special case around $\omega = 0$. Unless $\beta = 1$ we have the transform constant in a continuous region around $\omega = 0$ thus if we can confine ourselves to work in that region the proof is the same as for the sinc. Then for the linear dependence in time domain to be valid it must hold for all $\omega$ and especially for the case $\omega = 0$, thus we can actually confine ourselves to work in a region around $\omega = 0$. This case holds for all $0 < \beta < 1$, but for $\beta = 0$ the root-raised-cosine is actually the sinc so it follows from the theorem above.

This is a reassuring conclusion since it tells us that we have one degree of freedom per pulse we send, and this degree of freedom can be used for transmitting data. Otherwise there would be some pulse that is redundant and we would not be able to use it to send any independent data with that pulse. Having proved this it however quickly leads us to the following conclusion about our matrix $G$.

Corollary 3 (Positive definiteness of $G$). The matrix $G$ given by (3.3) is positive definite (and thus nonsingular) for any finite size $N$.

Proof. We know that $G$ is a Gram matrix, and from Theorem 6 that such a matrix is positive semidefinite. Moreover Theorem 5 states that such matrix is singular if and only if the defining vectors are linearly dependent, but from Theorem 11 we know that the vectors are linearly independent and hence $G$ is nonsingular, so $G$ must be strictly positive definite.

Now we skip ahead of time for a corollary regarding the root-raised-cosine pulses. The proof is identical as the one above but relies on Corollary 2.

Corollary 4 (Positive definiteness of $G$). The matrix $G$ given by (3.20) is positive definite (and thus nonsingular) for any finite size $N$ and $0 \leq \beta < 1$.

This is also a somewhat natural conclusion since that means that the sampled noise $w$ in (3.2), having $G$ as the covariance matrix, is non degenerate. If the matrix would have been singular we would have succeeded in sampling $N$ points in white noise, but getting less than $N$ independent noise samples.

Eigenvalues and trace of the Gram matrix

For the sake of completeness we are now going to derive some properties of the eigenvalues of $G$. We have that $G_{m,n} = \rho \text{sinc}(\rho(n - m))$ and identify this as the
Fourier coefficients of a function \( f(x) \). With the use of a table [15, equation (1) p. 313] we identify the function to be the \( 2\pi \)-periodic step function

\[
f(x) = \begin{cases} 
0 & x \in [-\pi, -\rho\pi) \\
1 & x \in [-\rho\pi, \rho\pi] \\
0 & x \in (\rho\pi, \pi] .
\end{cases}
\]

(3.5)

We can then from Theorem 8 get bound on the eigenvalues. Let \( \lambda_j^N \) be the \( j \):th eigenvalue, sorted in descending order and \( j \) is a fixed number, of the matrix \( G \) of size \( N \times N \), as given by (3.3). The we know that the eigenvalues obey

\[
0 \leq \lambda_j^N \leq 1
\]

(3.6)

\[
\lim_{N \to \infty} \lambda_j^N \to 1
\]

(3.7)

\[
\lim_{N \to \infty} \lambda_{N+1-j}^N \to 0 .
\]

(3.8)

Equation (3.8) reveals a potential problem, the larger the matrix \( G \) gets the closer it gets to singular and hence algorithms manipulating \( G \) must be able to handle that. Moreover trying to solve (3.2) by direct inverse will give an amplification of the noise that increases with \( N \) until it diverges.

We are now going to investigate the trace of \( G \), which is equivalent to the sum of eigenvalues: \( \sum_{j=1}^{N} \lambda_j^N \). In the finite case it is easy to see that

\[
\sum_{j=1}^{N} \lambda_j^N = \rho N .
\]

Using Szegö’s theorem (Theorem 7) and applying \( F \) as the linear function \( F(x) = x \) we can also see that

\[
\lim_{N \to \infty} \frac{1}{N} \sum_{j=1}^{N} \lambda_j^N = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) \, dx = \rho .
\]

We can thus conclude that:

\[
\sum_{j=1}^{N} \lambda_j^N = \rho N \quad \forall \; N \in [1, \infty) .
\]

(3.9)

Looking at Corollary 1 we conclude that as \( N \to \infty \) there is going to be \( \rho N \) eigenvalues that are non-zero and \( (1 - \rho)N \) eigenvalues converging to zero. More formally put we can say that if \( | \cdot | \) denotes the cardinality then \( \forall \; \epsilon \in (0, 1) \) we have that \( \{| \lambda_j^N > \epsilon \}_j^N \!/N \to \rho \) and \( \{| \lambda_j^N < \epsilon \}_j^N \!/N \to (1 - \rho) \) as \( N \to \infty \).

**Remark.** Please note that nowhere in the previous discussion was the actual value of \( \rho \) important, and no assumptions other than \( \rho > 0 \) has been used. The size of the alphabet used for signaling was neither needed to be specified (as long as it is finite and non-empty).
3.2 The difficulty of solving the ML-estimation problem

Now we return to looking at the system (3.2) and knowing that $G$ is invertible we choose to disregard the noise amplification and solve for $a$ to get:

$$Ga + w = y$$
$$a + G^{-1}w = G^{-1}y$$
$$a + v = z,$$

where $z = G^{-1}y$ and $v \in \mathcal{N}(0, \sigma^2 \cdot G^{-1})$. We want to apply maximum likelihood as before (1.30) and thus continue the derivation the same way by noticing that for a given $a$ we have $z \in \mathcal{N}(a, \sigma^2 \cdot G^{-1})$. We can then write the likelihood as [7]

$$P(z \mid a) = \left(\frac{1}{2\sigma^2\pi}\right)^{K/2} \cdot \frac{1}{\sqrt{\det[G^{-1}]}} \cdot e^{-\frac{1}{2\sigma^2} (z-a)^T G (z-a)}.$$

Once again we see that this is decreasing with increasing $(z-a)^T G (z-a)$, but this time it is not all obvious how to solve this. What we want is to find the solution to the binary quadratic program (BQP)

$$\min_a \quad (z-a)^T G (z-a)$$
$$\text{s.t.} \quad a = \{-1, 1\}^N.$$

Simplifying the objective function we get

$$(z-a)^T G (z-a) = z^T G z - a^T G z - z^T G a + a^T G a = z^T G z + a^T G a - 2z^T G a$$
$$= z^T G z + a^T G a - 2(Gz)^T a = z^T G z + a^T G a - 2y^T a.$$

Thus an equivalent formulation is,

$$\min_a \quad a^T G a - 2y^T a$$
$$\text{s.t.} \quad a = \{-1, 1\}^N.$$

(3.10)

Both BQPs differ in objective value by a constant, but the solution vector $a_{\text{opt}}$ is the same and this vector is the ML-estimation, the actual values at an optimal point $a = a_{\text{opt}}$ are less important. Note especially that we never have to compute $z$ in order to do an ML-estimation, which is reassuring since $G$ might be very ill conditioned (and is in fact for the sinc pulse since the eigenvalues goes to 0 as $N \rightarrow \infty$). The problem at hand depends on the received data $y$ and the known system matrix $G$. One should also note that the derivation of (3.10) as equivalent to the minimum error estimation only relied on that $G$ was invertible, not the exact form of the matrix (nor really the fact that we try to send faster-than-Nyquist).

We now extend the previous observation by allowing any finite alphabet and essentially any pulse shape and formulate it as Theorem 12.
\textbf{Theorem 12} (Minimum error estimation using PAM in AWGN channel). Let a communication system send the uncoded multilevel data \( a = \{-\alpha, -(\alpha - 1), \ldots, \alpha - 1, \alpha\}^N \) with equiprobable input, using PAM and the linearly independent pulses \( \{h_k(t)\}_{k=1}^N \). If the channel is an AWGN channel and the receiver is using a matched filter, the optimal do ML-estimation is the solution vector to
\[
\begin{align*}
\min_a & \quad a^T H a - 2y^T a \\
\text{s.t.} & \quad a = \{-\alpha, -(\alpha - 1), \ldots, \alpha - 1, \alpha\}^N ,
\end{align*}
\]
where \( H_{m,n} = \langle h_m(t), h_n(t) \rangle \) is the Gram matrix of the signals used, \( r(t) \) is the received signal and \( y_n = \langle r(t), h_n(t) \rangle \) are the samples from the matched filter.

\textit{Proof.} The proof follows from the derivation of (3.10) which in turn follows from the derivation of (1.32) and goes as follows. At the receiver one gets the equation
\[
Ha + w = y ,
\]
where \( w \in \mathcal{N}(0, \lambda \cdot H) \), which follows from the receiver structure and the AWGN channel. Here we do not bother with the exact form of the SNR and the factor \( \lambda \) will account for any scaling of it. From the assumption of linear independence of the pulses we know according to Theorem 5 that the Gram matrix \( H \) is non singular. Solving the above equation in the same way as before we get
\[
a + v = z ,
\]
where \( v \in \mathcal{N}(0, \lambda \cdot H^{-1}) \) and \( z = H^{-1}y \). With equiprobable input we then know, according to (1.30), that the estimation \( x \) that minimizes the error probability is \( x = \arg \max_a P(z \mid a) \). Combining that with the equation above and non singularity of \( H \) we see that:
\[
x = \arg \max_a P(z \mid a) = \arg \max_a \left( \frac{1}{\lambda 2\pi} \right)^{K/2} \cdot \frac{1}{\sqrt{\det[H^{-1}]}}, e^{-\frac{1}{2\lambda}(z-a)^T H(z-a)} .
\]
This is equivalent to minimizing the exponent (except the minus sign). \( z \) is our (manipulated) measurement and thus a constant with respect to the minimization. Hence we get the problem
\[
\begin{align*}
\min_a & \quad (z-a)^T H(z-a) \\
\text{s.t.} & \quad a = \{-\alpha, -(\alpha - 1), \ldots, \alpha - 1, \alpha\}^N .
\end{align*}
\]
Rearranging the cost function we get
\[
(z-a)^T H(z-a) = z^T Hz + a^T Ha - 2z^T H a = \underbrace{z^T Hz}_{\text{const. w.r.t. the min.}} + a^T Ha - 2y^T a ,
\]
3.2. THE DIFFICULTY OF SOLVING THE ML-ESTIMATION PROBLEM

Since the first term is constant with respect to minimization it can be excluded from the target function but the solution vector will still be the same.

Since all steps are equivalences, there is equivalence in solving the minimum error probability and the optimization problem (3.11) and we are done. 

\[ \text{Remark.} \] This is exactly the same type of mathematical problem encountered in MIMO-systems [31].

This gives us a systematic approach to solve for the minimum error estimation but the idea goes further. You do not have to solve the minimum error estimation using (3.11) but are free to use whatever technique you like, but solving the estimation problem is equivalent of finding the solution to (3.11) since once you have the first you also have the second. From this we can go on with another general result.

**Theorem 13** (Optimal detection in general PAM is NP-hard). To do minimum error detection in PAM over an AWGN channel with a receiver using a matched filter and applying some general linearly independent pulses to send uncoded multilevel data with equiprobal input is NP-hard.

**Proof.** The proof follows directly from Theorem 12 and proposition 1 in [32] and the detailjs are as follows. We look at the problem

\[
\max_a 2y^T a - a^T Ha \\
\text{s.t. } a = \{-\alpha, -(\alpha - 1), \ldots, \alpha - 1, \alpha\}^N,
\]

and from Theorem 12 we know that if we have an algorithm solving the optimal ML-detection detection in PAM, we can solve (3.12) with it. Now since the partition problem is reduced to (3.12) in [32], it is then further reduced to optimal ML-detection in PAM by Theorem 12. Hence by transitivity of polynomial time reductions, an NP-complete problem has been reduced to optimal ML-detection in PAM in polynomial time. This proves the statement.

\[ \text{Remark.} \] This does not prove that sending signals using PAM is a ‘hard’ problem, it is for example easily done using orthogonal pulses. What it does prove is that we must be specific and not rely on efficient general purpose algorithms to give the exact solution at the receiver.

The practical interpretation of these results are that in order to successfully use PAM (without a too complex receiver structure)\(^3\) one must be smart when sending and receiving data. When using orthogonal pulses we have seen that the system described in Theorem 12 reduces to diagonal which is an easily solvable subclass of the problem. If we want to send faster-than-Nyquist we cannot rely on orthogonal pulses.

\(^3\)Too complex here meaning algorithm of exponential complexity and is of course unless \(P = NP\) in which case it might be possible anyhow.
pulses but have to do something else. There are in fact a few cases of (3.10) that are solvable in polynomial time [33], although none of them match our particular structure of $G$ for the FTN case.

One possible way is to rely on approximations, thus allowing for higher error rates. In particular when knowing the structure of $G = \rho \text{sinc}(\rho (m - n))$ we can choose to disregard the full influence of the ISI and truncate $G$ for large enough difference in indexes $|m - n|$, essentially creating a banded Toeplitz matrix. The literature provides some research on this topic, but the algorithm is still exponential in the bandwidth of the matrix and hence the number of ISI’s taken into account, thus actually having the same complexity as the Viterbi algorithm in terms of regarded ISI-length [34].

We end this section with an alternative proof of the minimum error detection being NP-hard, a proof exploiting what is previously done and reduces the BQP-problem to detection using band limited pulses. It shows how a general purpose decoding algorithm could be used to solve NP-hard problems.

Alternative proof of Theorem 13. Assume that we have a general purpose algorithm $\mathcal{A}$ that takes a set of linearly independent pulse shapes $\{g_k(t)\}_{k=1}^N$, a measurement vector $y$ and an alphabet $A$ from which the symbols (amplitudes) are taken and returns the vector $a$ that is a solution to the minimum error detection problem. Then we claim that this algorithm can be used to solve the general instance of the binary quadratic program where $H > 0$, and $H \in \mathbb{R}^{N \times N}$,

$$\min_x \quad x^T H x$$

s.t. \quad $x = \{-1, 1\}^N$

which is known to be NP-hard in general.

From Theorem 6 we know that $H$ can be viewed as a Gram matrix for some set of linearly independent vectors in some Hilbert space. Now if we in polynomial time can construct a set of linearly independent vectors in $L^2(-\infty, \infty)$ whose Gram matrix is $H$ we are done. This in fact can even be done using band limited pulses as follows.

Let our family of pulses be $h_i(t) = \sum_{k=1}^i a_k^{(i)} \cdot \text{sinc}(t - k)$. In order to have the pulses decided we must decide on the coefficients $a_1^{(1)}, a_2^{(1)}, a_2^{(2)}, a_1^{(3)}, \ldots, a_N^{(N)}$. We trivially get $H_{1,1} = \langle h_1(t), h_1(t) \rangle = a_1^{(1)} \cdot a_1^{(1)}$, and hence $a_1^{(1)} = \sqrt{H_{1,1}}$ determining $h_1(t)$. Then if we regard the inner products from top to bottom (only interested in the upper triangular part of $H$ since it is symmetric) we can see that $H_{1,2} = \langle h_1(t), h_2(t) \rangle = a_1^{(1)} \cdot a_1^{(2)}$ where $a_1^{(2)}$ is the only unknown and we can hence solve for it. Then $H_{2,2} = (a_2^{(2)})^2 + (a_2^{(2)})^2$ and here only $a_2^{(2)}$ is unknown making it possible to solve for it. Continuing in this matter (which is actually just forward substitution) we can in $\sum_{k=1}^N k = O(N^2)$ time construct a set of linearly independent defining vectors for the Gram matrix $H$. The linear independence is proved by noticing that every vector we add has, from construction, a component
that is orthogonal to all previous vectors and hence it cannot be linearly dependent on the other vectors and the rest follows from induction.

Now we can feed our algorithm $\mathcal{A}$ with the constructed set of linearly independent vectors $\{h_k(t)\}_{k=1}^N$, the artificial measurement vector $y = 0$ and the alphabet $A = \{-1, 1\}$. Our algorithm returns the optimal estimation $x$ and from Theorem 12 we know that this is equivalent to a solution to the above stated BQP, which we know is NP-hard to solve. Thus if we have a polynomial time algorithm to solve optimal ML-estimation in PAM we can use it solve a general NP-hard problem. This completes the proof. \qed

### 3.3 SVD-precoding

Looking at Theorem 13, at our particular problem in Theorem 12, or the even more restricted (3.10) we realize that in order to do detection efficiently we should seek to do some smart coding of the data. One approach is use a coding that transform the equation to some of the subclasses where we have polynomial time algorithms.

#### Deriving the precoding scheme

In this approach we look at the system (3.2) again, but we rewrite it as

$$Ga + G^{1/2}\eta = y$$

(3.13)

where $\eta$ is zero-mean i.i.d. Gaussian vector with variance $\sigma^2$. To solve this we take a quite different approach and do a singular value decomposition of $G$, see Theorem 9 and the following remark. Inserting this in (3.13) we get:

$$USU^*a + U\sqrt{S}U^*\eta = y.$$  

(3.14)

We observe that if we instead of transmitting the communicated bits $a$ as the amplitudes of the pulses we instead use the amplitudes

$$\hat{a} = Ua$$

(3.15)

the system then becomes particularly easy to solve and we are going to call this SVD-precoding. Solving for the actual communicated symbols removes $U^*$ from the first term and then multiplying with $U^*$ removes the ISI on $a$.

$$USU^*\hat{a} + U\sqrt{S}U^*\eta = y,$$

$$USa + U\sqrt{S}\hat{\eta} = y,$$

$$Sa + \sqrt{S}\hat{\eta} = U^*y.$$  

Here $\hat{\eta}$ is still $\mathcal{N}(0,\sigma^2 \cdot I)$ since $U^*$ is unitary [7]. Completing the derivation we get the random variable $v \in \mathcal{N}(0,\sigma^2 \cdot S)$ still uncorrelated which here is equivalent with independent, $\hat{y} = U^*y$ and the system becomes:

$$Sa + v = \hat{y}.$$  

(3.16)
On this we want to do ML estimation, but this is not more difficult then what we already did in the preliminaries and the formula (1.32) still holds but applied to $\hat{y}$, that is

$$x = \text{sign}(U^*y) \quad (3.17)$$

is the optimal estimation in terms of minimum error probability.

Comments on SVD-precoding

A few notes are in place here, we claim that:

1. This does not prove that the BQP (3.10) in general has the optimal solution $\text{sign}(\hat{y})$.

2. The energy of the sent signal is reasonable, that is the precoder does not alter the expected energy. It is also more evenly distributed when using SVD-precoding since it becomes independent of the actual message transmitted.

3. For small eigenvalues the AWGN-channel model looses its validity. That means, for any SNR the receiver will still make estimation errors due to other noises showing up. Two examples of such noise are noise in the measurement $y$ that arises since we in practice simply cannot compute the inner product $\langle r(t), g^n(t) \rangle$ exactly (it would require infinite time integrals) and quantization noise due to limited numerical precision.

1) To prove the first claim we start by noticing that the whole idea with this approach is to avoid the difficulties with (3.10) by coding the signal as (3.15). The equation (3.10) is for when the amplitudes are binary $a$, but we send the transformed $\hat{a}$ instead which transforms the equation to becoming

$$\min_a a^T S a - 2 \hat{y}^T a$$

s.t. $a \in \{-1, 1\}^N$,

which is certainly easier to solve since we have decoupled the effects of the binary variables and we can read it as $\min_a S_{i,i} - 2 \hat{y}_i \cdot a_i, a_i \in \{-1, 1\}$ for $i = 1, 2, \ldots, N$. This makes the decision rule obvious since the cost function consists of a constant and something with a minus sign. It is of course also possible to compare with the derivation of (1.32) in the preliminaries (although that derivation ended earlier, before rewriting the BQP).
2) Investigating the second claim we look at the energy for a sent PAM signal with arbitrary data \(a\).

\[
\langle s(t), s(t) \rangle = \int_{-\infty}^{\infty} \left( \sum_{m=1}^{N} \hat{a}_m g^n_T(t) \right) \cdot \left( \sum_{n=1}^{N} \hat{a}_n g^n_T(t) \right) \, dt
\]

\[
= \sum_{m=1}^{N} \sum_{n=1}^{N} \hat{a}_m \hat{a}_n \cdot \int_{-\infty}^{\infty} g^n_T(t) g^n_T(t) \, dt = \hat{a}^T G \hat{a} = (a^T U^*) U S U (U a)
\]

\[
= a^T S a = \sum_{k} \lambda^N_k = \{\text{from (3.9)}\} = \rho N
\]

Thus we see that with this coding we are, regardless of what message we are sending, always going to use the same amount of energy for every block, which is the same as for the Nyquist case. This even distribution of energy among all possible messages is a result of our precoding, if one instead look at the uncoded case we have the energy

\[
\langle s(t), s(t) \rangle = a^T G a = \rho N + a^T \hat{G} a,
\]

where \(\hat{G} = G - \rho I\) and thus with zeros on the main diagonal. We can see that the energy is not equally distributed among all possible signals in this case, if it were the BQP in (3.10) would be trivially solved by any vector \(a\) in the case \(y = 0\). In fact the energy is quite uneven. An educated guess is that the energy for a strongly alternating signal, long sequences of \((\ldots, 1, -1, 1, -1, \ldots)\), is quite low. The guess is based on that in this case the ISI will be such that neighboring symbols dampen the main peaks of each other. As an example one can look at the completely alternating signal where the energy becomes \(E_{\text{alt}} = \rho N + \sum_{k=1}^{N} (-1)^k \cdot 2\rho(N - k) \cdot \text{sinc}(\rho \cdot k) \ll \rho N\) and compare it to the completely non-alternating case where the energy is \(E_{\text{nonAlt}} = \rho N + \sum_{k=1}^{N} 2\rho(N - k) \cdot \text{sinc}(\rho \cdot k) > \rho N\). Just how big the difference in energy is can also be seen in the difference between these two signals, \(E_{\text{nonAlt}} - E_{\text{alt}} = \sum_{k=1}^{N} 4\rho(N - k) \cdot \text{sinc}(\rho \cdot k) > \rho N\). This relates to error probability found in simulations where non-precoded detectors seems to perform worse for strongly alternating sequences whereas for SVD-precoding detectors the error rates seems to be approximately the same regardless of message.

As a final comment on the energy in the non-coded case we can conclude that even though there is a huge difference in the actual energy of different messages, the expected energy is still given by the trace of \(G\) and is thus \(\rho N\) as it should be. Denote the (expected) energy \(E\) and the expectation operator \(E[ \cdot ]\). Then

\[
E = E[a^T G a]
\]

but \(E\) is scalar and the trace of a scalar is it self, hence following the standard derivation:

\[
E = \text{trace}[E] = \text{trace} [E[a^T G a]] = E[\text{trace} [a^T G a]]
\]

\[
= E[\text{trace} [G a a^T]] = \text{trace} [G E[a a^T]] = \text{trace} [G \cdot I] = \text{trace} [G] = \rho N,
\]
CHAPTER 3. RECEIVING AND DETECTING FTN SIGNALS

using that expectation and trace are linear and interchangeable operations. The identity matrix $I$ occurs for signaling input that is independent, zero mean, and unit variance.

3) We look at an independent measurement error $\delta y$ occurring in measuring $y$, for any of the two previously stated reasons. We model it as $\delta y \in \mathcal{N}(0, \lambda^2 \cdot I)$. Then (3.13) looks like

$$Ga + G^{1/2} \eta = y + \delta y.$$  

We do exactly the same thing as in the derivation of (3.16) but the result turns out to be

$$Sa + v = \hat{y} + \delta \hat{y}.$$  

Or equivalently

$$Sa + \hat{v} = \hat{y},$$  

(3.18)

where $\hat{v} \in \mathcal{N}(0, \sigma^2 \cdot S + \lambda^2 \cdot I)$ [7]. If we now turn up the SNR to infinity, that is putting $\sigma^2 = 0$ we see that there is still this finite noise covariance $\lambda^2 \cdot I$. This has been verified in numerical experiments in MATLAB where one could observe that, if applying the sinc pulse, all the errors were located at the end (approximately the $(1 - \rho)N$ last positions) of the data vector $a$ and at these positions the error rate was approximately 50%. Another thing one could observe in those simulations was that the observed measurements $\hat{y}$ as described in (3.16) were several orders of magnitude larger than the corresponding eigenvalues for the last $(1 - \rho)N$ positions, indicating that there were indeed noise dominating these measurements and explaining the 50% error probability at these positions. All of this regardless of the SNR (SNR = $\infty$ was simulated by simply not adding any noise to the signal).

3.4 Root-raised-Cosine pulses in FTN

The previous approach with sinc pulses seems to suffer from the fact that a significant proportion of the eigenvalues are converging to zero rather quickly. We will therefore change our approach by changing the pulses to root-raised-cosine (1.21). Fortunately most of what we have discussed so far has only been dependent on that the pulses applied was linearly independent, which was also shown for the root-raised-cosine pulses, for finite number; see Corollary 2.

Gram matrix for root-raised-cosine

The Gram matrix proved to be an important tool in the previous sections, therefore we start this investigation by deriving it for the root-raised-cosine (rtrc). As before
3.4. ROOT-RAISED-COSINE PULSES IN FTN

this is done by looking at the inner product between two pulses, one shifted $m$ time steps and the other shifted $n$ steps.

\[ \hat{G}_{m,n} = \langle g_{T,\beta}(t - m \cdot \rho T), g_{T,\beta}(t - n \cdot \rho T) \rangle = \int_{-\infty}^{\infty} g_{T,\beta}(t - m \cdot \rho T) \cdot \overline{g_{T,\beta}(t - n \cdot \rho T)} \, dt \]

\[ = \{ \text{Plancherel’s formulae} \} = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega m \rho T} G_{T,\beta}(\omega) \cdot e^{-i\omega n \rho T} \overline{G_{T,\beta}(\omega)} \, d\omega \]

\[ = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-i\omega(m-n) \rho T} |G_{T,\beta}(\omega)|^2 \, d\omega \]

\[ = \{ \text{Fourier transform is given by (1.22)} \} = \{ \text{Split integral} \} \]

\[ = \{ \text{Change variable, } \theta = \omega \rho T \} = \frac{1}{2\pi} \int_{-(1-\beta)\rho \pi}^{(1-\beta)\rho \pi} e^{-i\theta(m-n)} \, d\theta \]

\[ + \frac{1}{4\pi \rho} \int_{-(1-\beta)\rho \pi}^{(1+\beta)\rho \pi} e^{-i\theta(m-n)} \cdot \left[ 1 - \sin \left( \frac{1}{2\beta} \left( \frac{\theta}{\rho} - \pi \right) \right) \right] \, d\theta \]

\[ + \frac{1}{4\pi \rho} \int_{-(1+\beta)\rho \pi}^{-(1-\beta)\rho \pi} e^{-i\theta(m-n)} \cdot \left[ 1 - \sin \left( \frac{1}{2\beta} \left( -\frac{\theta}{\rho} - \pi \right) \right) \right] \, d\theta \]

\[ = \{ \text{In the last integral, change } \gamma = -\theta, \text{ and swap boundaries} \} \]

\[ = \{ \text{Solve the first integral} \} = \{ \text{merge the two remaining integrals} \} \]

\[ = (1-\beta) \text{sinc}((1-\beta)\rho(m-n)) \]

\[ + \frac{1}{4\pi \rho} \int_{-(1-\beta)\rho \pi}^{(1+\beta)\rho \pi} e^{-i\theta(m-n)} \cdot \left[ 1 - \sin \left( \frac{1}{2\beta} \left( \frac{\theta}{\rho} - \pi \right) \right) \right] \, d\theta \]

\[ = (1-\beta) \text{sinc}((1-\beta)\rho(m-n)) \]

\[ + \frac{1}{2\pi \rho} \int_{(1-\beta)\rho \pi}^{(1+\beta)\rho \pi} \cos(\theta(m-n)) \cdot \left[ 1 - \sin \left( \frac{1}{2\beta} \left( \frac{\theta}{\rho} - \pi \right) \right) \right] \, d\theta \]

\[ = \frac{1}{2} \text{sinc}((1+\beta)\rho(m-n)) + \frac{1}{2} \text{sinc}((1-\beta)\rho(m-n)) \]

\[ - \frac{1}{2\pi \rho} \int_{(1-\beta)\rho \pi}^{(1+\beta)\rho \pi} \cos(\theta(m-n)) \cdot \sin \left( \frac{1}{2\beta} \left( \frac{\theta}{\rho} - \pi \right) \right) \, d\theta \]

Now with the aid of some symbolic software [35] we can finally solve the last integral and arrive at the desired result, a closed form expression of the Gram matrix for
the root-raised-cosine pulses of roll-off parameter $\beta$ and FTN parameter $\rho$.

\[ \hat{G}_{m,n} = \frac{(1 + \beta)}{2} \cdot \frac{\rho}{1 - 4\beta^2 \rho^2 (m - n)^2} \cdot \text{sinc}((1 + \beta)\rho(m - n)) \]

\[ + \frac{(1 - \beta)}{2} \cdot \frac{\rho}{1 - 4\beta^2 \rho^2 (m - n)^2} \cdot \text{sinc}((1 - \beta)\rho(m - n)) \]

\[ - \frac{4\beta^2 \rho}{\pi(4\beta^2 \rho^2 (m - n)^2 - 1)} (m - n) \cdot \sin(\pi \rho(m - n)) \cdot \cos(\pi \rho \beta(m - n)) \]

(3.19)

A few observations are in place here. Note that for $\beta = 0$ this reduces to the Gram matrix for the sinc pulses (3.3) as it should since the root-raised-cosine pulse itself reduces to the sinc pulse. Moreover for $\rho = 1$ the matrix reduces to the identity matrix; the last term disappears from $\sin(\pi (m - n)) = 0$ and the two other terms cancel each other out except for $n = m$ when they sum to 1, corresponding to the main diagonal.\footnote{This can be seen by expanding sinc using definition from (1.17), remember that $T = 1$ when not explicitly written, and then use the argument sum rule of sinus on the terms $(m-n)\pm\beta(m-n)$.}

This is also expected and desired since the root-raised-cosine pulses are shift orthogonal for the parameter $T$. A third observation is that $G_{m,n}$ is an even function of the argument $(m - n)$ implying that the matrix is symmetric and Toeplitz, the same nice properties as for the Gram matrix of the sinc pulses.

Now as the derivation is almost done we include the factor $\rho$ to keep constant power at the sender as mentioned in the beginning of this chapter, moreover one can rearrange the last term to reach the more compact result

\[ G_{m,n} = \frac{(1 + \beta)}{2} \cdot \frac{\rho}{1 - 4\beta^2 \rho^2 (m - n)^2} \cdot \text{sinc}((1 + \beta)\rho(m - n)) \]

\[ + \frac{(1 - \beta)}{2} \cdot \frac{\rho}{1 - 4\beta^2 \rho^2 (m - n)^2} \cdot \text{sinc}((1 - \beta)\rho(m - n)) . \]

(3.20)

It might seem as if the Gram matrix above has two possible poles. The denominators can be factorized as $(1 + 2\beta \rho(m - n))(1 - 2\beta \rho(m - n))$ and hence for suitable choices of $\beta$ and $\rho$ there would be some integer $\pm(m - n)$ where the result would be infinite. This is however not the case since the numerator also goes to 0 for these values. A somewhat tedious application of l’Hospital’s rule \cite[pp.133-134]{15} reveals that the limit is in fact

\[ \lim_{x \to \pm 1/2} \frac{\rho}{1 - 4x^2} \left( \frac{1 + \beta}{2} \cdot \text{sinc}(\pi(1/\beta + 1) \cdot x) + \frac{1 - \beta}{2} \cdot \text{sinc}(\pi(1/\beta - 1) \cdot x) \right) \]

\[ = \frac{\beta \cdot \rho}{2} \sin \left( \frac{\pi}{2\beta} \right) . \]

(3.21)
Investigating the Eigenvalues

From Corollary 2 we know that this matrix, (3.20), is nonsingular in the finite case. We are however going to investigate the Eigenvalues of $G$, again using Theorem 8 since it revealed some problems with the sinc pulse. We assume that $\rho \leq 1$ since this is FTN. Here the calculations are more tedious, and therefore found in Appendix A. Instead we directly present the Szegö function for root-raised-cosine:

$$f(x) = \begin{cases} 
0 & x \in [-\pi, -(1 + \beta)\rho \pi] \\
\frac{1}{2} \left(1 + \sin \left(\frac{x + \rho \pi}{2 \beta \rho}\right)\right) & x \in [-(1 + \beta)\rho \pi, -(1 - \beta)\rho \pi] \\
1 & x \in [-(1 - \beta)\rho \pi, (1 - \beta)\rho \pi] \\
\frac{1}{2} \left(1 - \sin \left(\frac{x - \rho \pi}{2 \beta \rho}\right)\right) & x \in [(1 - \beta)\rho \pi, (1 + \beta)\rho \pi] \\
0 & x \in [(1 + \beta)\rho \pi, \pi]
\end{cases}$$

If $(1 + \beta)\rho \leq 1$

$$f(x) = \begin{cases} 
\frac{1}{2} \left(1 + \sin \left(\frac{x + \rho \pi}{2 \beta \rho}\right)\right) & x \in [-\pi, -(2 - (1 + \beta)\rho)\pi] \\
1 - \sin \left(\frac{\pi (1 - \rho)}{2 \beta \rho}\right) \cos \left(\frac{x + \pi}{2 \beta \rho}\right) & x \in [-(2 - (1 + \beta)\rho)\pi, -(1 - \beta)\rho \pi] \\
1 & x \in [-(1 - \beta)\rho \pi, (1 - \beta)\rho \pi] \\
\frac{1}{2} \left(1 - \sin \left(\frac{x - \rho \pi}{2 \beta \rho}\right)\right) & x \in [(1 - \beta)\rho \pi, (2 - (1 + \beta)\rho)\pi] \\
1 - \sin \left(\frac{\pi (1 - \rho)}{2 \beta \rho}\right) \cos \left(\frac{x - \pi}{2 \beta \rho}\right) & x \in [(2 - (1 + \beta)\rho)\pi, \pi]
\end{cases}$$

(3.22)

If $(1 + \beta)\rho \geq 1$

An example of how the function $f(x)$ looks like for the case $(1 + \beta)\rho \geq 1$ can be found in Figure 5.1, around page 47.

In our investigation of the eigenvalues we are now ready to apply Theorem 8 and conclude that

---

This is equivalent with $(1 - \beta)\rho \geq 2 - (1 + \beta)\rho$, and together with $\beta \in [0, 1]$ it assures that $(1 + \beta)\rho \leq 2$. This is of technical nature to the derivation and the points where the function changes. The case when this does not hold is not investigated.
For \((1 + \beta)\rho \leq 1:\) 
\[
0 \leq \lambda_j^N \leq 1
\]
\[
\lim_{N \to \infty} \lambda_j^N \to 1
\]
\[
\lim_{N \to \infty} \lambda_{N+1-j}^N \to 0
\]

For \((1 + \beta)\rho \geq 1:\) 
\[
1 - \sin \left(\frac{\pi(1 - \rho)}{2\beta\rho}\right) \leq \lambda_j^N \leq 1
\]
\[
\lim_{N \to \infty} \lambda_j^N \to 1
\]
\[
\lim_{N \to \infty} \lambda_{N+1-j}^N \to 1 - \sin \left(\frac{\pi(1 - \rho)}{2\beta\rho}\right).
\]

### The trace if the Gram matrix

We investigated the trace of the Gram matrix for the sinc case and for the sake of completeness we do so also for the root-raised-cosine pulse (this is also known to be the same as the sum of the eigenvalues). For the finite case we can just rely on the structure of the Gram matrix (3.20) and see that it is \(\text{trace}(G) = \sum_j \lambda_j^N = \sum_j G_{j,j} = \rho N\). For the limit when \(N \to \infty\) we apply Szegö’s theorem (Theorem 7) with a linear function to get:
\[
\lim_{N \to \infty} \frac{1}{N} \sum_{j=1}^{N} \lambda_j^N = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) \, dx = \rho,
\]
when solved since the integral corresponds to the zero:th Fourier coefficient of \(f(x)\) which we know is the diagonal element of \(G\) and thus \(\rho\). Hence regardless of the factor \((1 + \beta)\rho\) we can again conclude the following to be true for all sizes of the matrix,
\[
\sum_{j=1}^{N} \lambda_j^N = \rho N \quad \forall \ N \in [1, \infty)
\]

### On the choice of \(\rho\)

In the derivations found in Appendix A, more specifically in (A.3), the parameter constellation \((1 + \beta)\rho\) shows up and changes the nature of the function. It then propagates through the derivation to the final function, (3.22). We can see that for \((1 + \beta)\rho < 1\) the same structure shows up as for the sinc, with eigenvalues going to zero, but for \((1 + \beta)\rho > 1\) this does not happen. It might seem discouraging that the same type of degeneration shows up for the two most commonly studied pulses, but whereas for the sinc pulse this was a fact that we were unable to get rid of for any FTN, the root-raised-cosine pulse do give us some parameter constellations that are in some sense better.
Theorem 14. Apply FTN to PAM signaling with time shifted root-raised-cosine (1.21), with roll-off parameter $\beta$ and FTN parameter $\rho$, over an infinite time horizon. Then in order to not have a singular system matrix one should keep

$$(1 + \beta)\rho \geq 1.$$ (3.27)

Proof. The proof follows from the fact that the receiver has the structure (3.2), the structure of the Gram matrix (3.20), and the Szegő function (3.22). Then from [20, theorem 5.2] we have that if $(1 + \beta)\rho \geq 1$ then $G$ is nonsingular and the system has a unique solution. \hfill $\square$

Remark. The absolute summability of the Fourier series as required by theorem 5.2 follows from the fact that (3.20) decays as $1/n^3$.

This might be important in practical considerations to since the matrix becomes ill-conditioned and numerically indefinite otherwise. This has been found to happen even for smaller matrix sizes and around $(1 + \beta)\rho = 1$.

An interesting special case: $\beta = 1$, $\rho = 0.5$

With a closed form expression of the Gram matrix one can easily find this interesting special case for the parameter combination $\rho = 0.5$ and $\beta = 1$. Basically we can rewrite the expression of (3.20) to be

$$G_{m,n} = \frac{1}{2\pi(m-n)} \cdot \frac{1}{1 - 4\beta^2\rho^2(m-n)^2} \cdot 2\sin(\pi\rho(m-n)) \cdot \cos(\pi\rho\beta(m-n))$$

$$= \{\text{For the case } \rho = 0.5 \text{ and } \beta = 1\}$$

$$= \frac{1}{\pi(m-n)} \cdot \frac{1}{1 - (m-n)^2} \cdot \sin\left(\frac{\pi}{2}(m-n)\right) \cdot \cos\left(\frac{\pi}{2}(m-n)\right).$$

Here we see that for all values $(m-n) \in \mathbb{N}\{0, 1\}$ the matrix is zero since the denominators are non-zero and $\cos(\pi/2 \cdot [\text{integer } \neq 0]) = 0$. Then for $(m-n) = 0$ we see that the limit value is $\rho = 0.5$ since $\lim_{x \to 0} \sin(0.5\pi x)/0.5\pi x = 1$ and for $(m-n) = 1$ we rely on the limit (3.21), and the value is $1 \cdot 0.5/2 \cdot 1 = 1/4$. Thus the matrix becomes tridiagonal and hence the estimation problem described as a BQP in (3.10) in solvable in polynomial time [36].

Although for practical purposes this algorithm might be to slow (look at the simulation results in the reference [36]). Moreover this parameter setting is most likely sub-optimal in terms of capacity since one would like the power spectral density (square of the Fourier transform) to be as even as possible. The sinc pulse, having a constant transform inside the bandwidth, is optimal in that sense but with $\beta = 1$ the root-raised-cosine has very little in common with that since the constant part of that spectrum is completely removed. Never the less it is an interesting special case and shows the value of having a closed form expression.
Chapter 4

Linear Precoding

In Chapter 3 we investigated the SVD-precoding and found it useful for countering the ISI caused by the faster-than-Nyquist signaling. It exploits some of the properties of FTN and provide good features, also giving good insights in the degeneration of the matrices as the pulses approached linear dependence for small enough $\rho$. For practical purposes it is however still not optimal since it does not distribute the energy evenly among the communicated bits; the bits corresponding to small singular values are getting less energy than the others which is a suboptimal approach. This is potentially avoided by using water filling, but this however falls outside the scope of this thesis. Here instead we present some other precoding techniques, and since the modulation and receiver is linear we will focus on linear codes.

4.1 GTMH-precoding

GTMH-precoding is short for G-to-minus-half-precoding and as the name indicates it will use the inverse of the square root of the Gram matrix for precoding purposes. Hence one will use the amplitudes:

$$\hat{a} = \sqrt{\rho}G^{-1/2}a.$$  \hspace{1cm} (4.1)

At the receiver the system (3.13) then reduces to

$$y = G^{1/2}(\sqrt{\rho}a + \eta)$$  \hspace{1cm} (4.2)

where $\eta$ is zero mean Gaussian with covariance $I \cdot \sigma^2$, and hence uncorrelated. Thus one can freely invert the square root of $G$ once more and then apply the normal ML-estimation to

$$\hat{y} = G^{-1/2}y = \sqrt{\rho}a + \eta$$  \hspace{1cm} (4.3)

The GTMH-precoding has some of the nice properties of SVD-precoding, such as making the estimation at the receiver end easy by removing both the ISI and
the noise correlation. Another good property is the energy, this precoding scheme
is averaging the energy making the energy per sent block independent of the actual
sent message, the same as claim 2 in SVD-precoding. The sender power is also kept
constant with this precoding, \( \langle s(t), s(t) \rangle = \sqrt{\rho a^T (G^{-1/2})^T G G^{-1/2} a} \sqrt{\rho} = \rho N \).

In calculating the matrix square root we know that it is not unique, but we can
calculate it using SVD and then we can require that we use the positive root of
every element. By using this technique to calculate the square root of \( G \) we may
directly calculate the GTMH-matrix, \( G^{-1/2} \), by simply inverting all the elements in
the diagonal matrix \( S \), from the SVD, at the same time as we compute the square
root of them.

**Complexity of the coding scheme**

Since the non-coded problem is NP-hard and the best known algorithm to deal with
that is the Viterbi algorithm with its exponential complexity, \( O(N 2^K) \) where \( K \) is
the length of the ISI.\(^1\) Therefore it is important to comment on the complexity of
this proposed percodng/decoding. The calculation of \( G^{-1/2} \) can be done in \( O(N^3) \)
flops and is a preparatory one time computation, after that the decoding is just
a matrix vector multiplication which can be done in \( O(N^2) \) time. Hence once the
system is set up the decoding and estimation is \( (N^2) \), a significant improvement.

**Cholesky factorization, on the fly decoding**

Another way to calculate a root is to do Cholesky factorization; this is not exactly
a square root as in Definition 15 since it is \( G = LL^T \). It is however efficient and
will work as long as \( G \) is positive definite, which for root-raised-cosine is as long as
\( (1 + \beta) \rho \geq 1 \). To see the usefulness we once again consider system (3.2), but this
time we apply the Cholesky factorization turning it into
\[
y = LL^T \hat{a} + L\eta .
\]

Where \( L\eta \) is valid since that will produce a covariance matrix \( LL^T = G \) if we let it
affect our Gaussian vector. Using the precoding
\[
\hat{a} = \sqrt{\rho} (L^T)^{-1} a \tag{4.4}
\]
the problem reduces to
\[
y = L(\sqrt{\rho} a + \eta) .
\]

This is a major improvement since \( L \) is lower triangular. Hence \( L^{-1} \) is also lower
triangular and the decoding can be made on the fly. This since
\[
\hat{y} = L^{-1} y = \sqrt{\rho} a + \eta ,
\]

\(^1\)For most cases \( K = N \) and setting \( K < N \) could be used as an approximation. However
there are exceptions, e.g. the special case in the end of section 3.4
and thus to know the first element of $\hat{y}$ (the measurement we want to apply our ML-estimation to) we only need to have measured the first element of $y$. Hence to start our ML-estimation we need not to wait for all the measurements but can start as soon as the first bit arrives. On the other hand $L^T$ is an upper triangular matrix and thus $(L^T)^{-1}$ is also an upper triangular matrix, making it necessary for the transmitter to wait for all the needed bits before generating the signal. This however can be defended by the same claims that made the model causal in the first place, that in order to get a nice continuous signal the sender would anyhow wait for all the bits before generating a signal to send, thus no extra delay is introduced in the system. In practical situations there will also be techniques such as block coding and interleaving that has to be applied to the full block, and hence no significant delay will be introduced by the precoding.

Another thing that is useful to point out for practical purposes is the fact that $L^{-1}$ does not need to be computed. The system

$$L\hat{y} = y$$

can be solved just as easily by forward substitution which is the normal way of solving this type of equation. Thus one can save computation time as well as memory and memory accesses in the computation.

### 4.2 Some words on general linear precoding

Linear precoding corresponds to a matrix-vector multiplication at the sender end;

$$\hat{a} = Aa$$

where $\hat{a}$ are the amplitudes, $a$ the communicated bits, and $A$ the precoding matrix. At the receiver there is then another matrix-vector multiplication to produce the decoded samples.

$$\hat{y} = By.$$  

A diagram explaining how a linear precoder is attached to the AWGN-channel model is found in Figure 4.1, this can be compared to the diagram in Figure 1.1. It is worth noticing that whatever other preprocessing done in the system affecting $a$ can still be done, the linear coding only forms an intermediate layer countering the ISI in the channel. The same goes for the samples, what ever post-processing and estimation one did with the samples $y$ in Figure 1.1 one can do to the decoded samples $\hat{y}$ in Figure 4.1. An example that will come up in Chapter 5 is turbo coding, which is easily implemented together with both SVD- and GTMH-precoding.

### The goal with precoding

Essentially any type of matrix transformation and/or factorization of $G$ might lead to a suitable precoding; it seems however like the tricky part is to make the noise
uncorrelated at the same time as the ISI is removed. Both the precoding schemes presented earlier in this thesis have successfully decorrelated the noise. Other factorizations that can be thought of are for example LU decomposition or QR decomposition, but their use remains questionable. As long as the precoding/decoding does not manage to diagonalize the noise finding the solution will still be NP-hard and any polynomial time solution will be an approximation.\(^2\) Strictly speaking the noise must not be diagonalized but reduced to something solvable when the ISI is removed, as long as \(G\) is positive definite it is enough to get the system described in Theorem 12 on a solvable form.

**Symmetry in the positive definite case**

Note that the precoding/decoding matrices \(A\) and \(B\) as presented here could possibly be 'uncorrelated' and also allowed to be the identity matrix of no change is desired at respective end.\(^3\) However by investigating the expression in Theorem 12 we see that in the target function the entity \(y^T a\) shows up in the non-precoded case and thus \(y^T \hat{a}\) in the precoded, indicating that the estimation should be applied to \(A^T y\). Thus suggesting that \(B = A^T\) should correspond to some kind of symmetric, optimal precoding-decoding scheme; at least as long as the matrix \(G\) is invertible. This fits well with the two precoding schemes presented earlier in this thesis and would explain why LU and QR decompositions have not proved good for creating precoders.

\(^2\)Unless \(\text{P} = \text{NP}\) in which case exact polynomial time solutions may be possible anyhow.

\(^3\)Although letting \(A = I\) is actually not worth mentioning as precoding and having a non-full rank matrix seems like a bad idea.
Chapter 5

Numerical results

Error rates

The error rates presented later in this chapter are closely related to the error probabilities in Definition 7, from a frequentist perspective they are to be seen as numerical measurements of these. For example the bit error probability:

\[ p_b = \frac{1}{K} \sum_{i=1}^{K} P(\hat{b}_i \neq b_i) \approx \frac{1}{K} \sum_{i=1}^{K} \frac{\text{number of times } \hat{b}_i \neq b_i}{\text{number of iterations}} \]
\[ = \frac{\text{total number of incorrectly decoded bits}}{\text{data bits/block} \cdot \text{blocks sent} \cdot \text{iterations}}. \]

5.1 Numerical investigation of the Eigenvalues

To show the validity of (3.22) being the corresponding Szegő function to the Gram matrix for the root-raised-cosine, (3.20), and that it indeed represents the distribution of the eigenvalues Figure 5.1 was generated. It shows the similarity between the distribution of the eigenvalues and the Szegő function according to Corollary 1, and is a comparison of numerically calculated eigenvalues with the direct sum of the corresponding Fourier series with coefficients given by (3.20) and as well as the closed form expression of the function from (3.22). It proves nothing new, rather it gives validation and intuition to what is already known. It was generated for the parameters \( \beta = 0.22 \) and \( \rho = 0.82 \), a matrix size of 1700x1700, and summing all the coefficients of the Fourier series from index \(-6000\) to \(6000\). The calculation of the eigenvalues was actually used by applying SVD (which is mathematically equivalent in this case). It is also worth commenting that the matrix for the root-raised-cosine actually becomes numerically indefinite in the region \((1 + \beta)\rho < 1\) which makes it hard to work with and efficiently removes the possibility of on-the-fly decoding using Cholesky factorization.
Figure 5.1: a) The Eigenvalues of the matrix (3.20) for size $N = 1700$ in decreasing order. b) The function approximation from summing the corresponding Fourier series from $n = -6000$ to $n = 6000$. c) The analytic function described in (3.22). Calculations were done for $\rho = 0.82$ and $\beta = 0.22$.

A similar plot is found in Figure 5.2, it was generated with matrix size of only 100x100 and here the eigenvalues are spread equidistant over the interval $[0, \pi]$ and plotted in the same graph as Szegö function. One can then see that the eigenvalues, as expected, seems to converge to the same distribution as the Szegö function.

A third numerical investigation was done for the, in limit, smallest eigenvalue for the Gram matrix in the non-singular case. From the Szegö function in (3.22) as well as the illustration in Figure 5.1 we can see that the smallest eigenvalue occurs for $x = \pm \pi$ and is then given by $1 - \sin\left(\frac{\pi(1-\rho)}{2\beta}\right)$. The contours of this, as a function of $\beta$ and $\rho$, are plotted in Figure 5.3 and could provide a good insights in how the weakest of the diagonal channels created by FTN perform. This is although more important if SVD-precoding is applied since this is equivalent to sending data across independent channels with these powers.

5.2 Simulation 1; description and benchmark simulation

This section describes the setting of the simulations as well as a benchmark simulation that can be used to compare the result against.
5.2. SIMULATION 1; DESCRIPTION AND BENCHMARK SIMULATION

Figure 5.2: The Eigenvalues of the matrix (3.20) for size $N = 100$ in decreasing order and spread equidistant over the interval $[0, \pi]$ together with the corresponding Szegö function (3.22). For $\rho = 0.82$ and $\beta = 0.22$.

Figure 5.3: Contour plot of $1 - \sin\left(\frac{\pi(1-\rho)}{2\beta^2}\right)$, which is the expression for the smallest eigenvalue of the Gram matrix for the root-raised-cosine. The zero-line is an approximation of the line $(1 + \beta)\rho = 1$ and the dashed horizontal line is $\beta = 0.22$. 
CHAPTER 5. NUMERICAL RESULTS

Simulation setup

All simulations were carried out in MATLAB where binary input alphabet PAM signals\(^1\) were modeled over an AWGN-channel. The input distribution of the data bits where random and independent with \(P(1) = 0.5\) and \(P(-1) = 0.5\) and the pulse shape applied was a truncated root-raised-cosine with \(\beta = 0.22\); it was truncated after \(14T_{send} = 14\rho T\). The data was sent in three blocks with a variable size (although in one specific simulation all blocks were of the same size). The blocks were separated by a guard interval of \(4T_{send} = 4\rho T\) meaning that in the ISI case the blocks actually interfered with each other at the ends, although none of the precoding schemes took care of that. The simulation was repeated a number of times, normally 500, in order to get statistical data since the input and noise are random.

The simulation was built on the formulation in (3.13) and the amplitudes where shifted using \(G\), which was numerically computed for the truncated case.

Remark. Observe that when using truncated pulses the Gram matrix is not a truncated version of (3.20). A direct computation of the corresponding inner products shows that it is indeed different. However no closed form expression was derived for this since it amounted to too much work for little gain.\(^2\)

The noise was generated as an i.i.d. Gaussian vector with 0 mean and variance equal to 1. This vector was then right-multiplied with \(G^{1/2} \cdot \sigma\) and transposed in order to generate the desired distribution.\(^3\) The SNR was measured in dB, see section 2 in Chapter 1 for further explanation of how it links to \(\sigma\).

There was also the possibility of using WCDMA-turbo-coding. This is a forward error correcting code applied so that the receiver should be able to decode the message even if some bits are corrupted, this is normally applied in a system since the noise can corrupt bits and instead of having to resend the message due to error in a few bits this allows the receiver to correct some bits. This is a block code as described in Definition 6 with a variable code rate, a longer explanation of turbo codes can be found in [4] but these were not studied in the scope of this thesis. If no turbo codes were used the estimation was done according to the ML-estimation rule (1.32) derived in Chapter 1.

To the description of the simulation it has to be added that a randomized channel interleaver/burst interleaver was used, essentially mixing the bits between the three blocks before the sending. This is normally used in real channels where the conditions vary and may be so severe that almost complete blocks/bursts can

---

\(^1\)Not necessary binary amplitudes since precoding might change that.

\(^2\)In time domain calculations the first out of three integrals were solved partially using WolframAlpha (see the references) and it turned out to be a sum with 20 terms. In frequency domain the Fourier transform of the truncated pulse contained less terms but they contained a sum of the exponential integral which would then be squared and integrated again.

\(^3\)Observe the right multiplication. This: 1) seems to be the standard way of generating correlated Gaussian noise numerically. 2) was in order to not make the GTMH-decoding trivially decoupling the noise in MATLAB, since it makes a left multiplication with \(G^{-1/2}\).
be lost. This interleaving then helps the error correcting code to be able to restore that data since the actual data is spread over many blocks. In the AWGN-channel this interleaving should have no effect, but for SVD-precoding it seemed to have.

**Remark.** To interpret the figures correct it is important to understand the effect of FTN in these simulations. The SNR, it is always given as defined in (1.28). That means that even if the pulse energy is scaled by $\rho$, and there occurs a loss in $E_b/N_0$ in the constant power FTN case, it does not affect the SNR as shown here. This is instead taken care of in $G$. The SNR is thus a standardized SNR as it would have been in the Nyquist case.

**Benchmark simulation**

The specific simulation in this section is done without FTN and hence no precoding was used. There is almost no ISI, only extremely small ISI due to the truncation of the pulse shape. There was 6000 bits per block and the whole simulation was repeated 500 times per SNR-value. WCDMA-turbo codes was used with a code rate $= 2/3$, meaning that 4000 bits were data from our input distribution and 2000 bits were redundancy bits added by the turbo code. The result, error rate plotted against SNR, can be found in Figure 5.4.

![Figure 5.4: The benchmark simulation. Binary PAM over and AWGN-channel with WCDMA-turbo codes with code rate $= 2/3$ applied. When the line disappears it means that the receiver performed error free.](image-url)
5.3 Simulation 2; comparing GTMH, SVD, and no precoding in FTN

This simulations is aimed at being as real as possible, although an AWGN-channel is a strong simplification. Here every block of data consisted of 6000 bits and WCDMA-turbo code was applied with code rate = 2/3, thus 4000 bits were data and 2000 bits were redundancy. \( \rho \) ranged from 0.65 to 0.9 with a step size of 0.025 and the SNR was in the range from 1.5 to 7 dB with a step size of 0.25. Each such simulation were repeated 500 times, so that for every combination of \( \rho \), SNR, and precoding there were a total of 1500 blocks simulated, each block consisting of 6000 bits.

About the interleaving; it seemed not to have any affect on the GTMH-precoding nor the no-precode cases, but it seemed to improve the performance of the SVD-precoding. This probably because then the low energy bits is then not a sequence of consecutive bits in the actual message but rather random. Hence if they were lost the turbo code could correct that. The improvement was mostly in the block error rate which supports that claim. The interleaving was used in this simulation.

A comparison of the performance for the SVD-precoding, the GTMH-precoding and the system without precoding is found in Figure 5.5.

Remark. Note that for the following error-rate-plots the bottom of the y-axis represent the minimum possible error rate (i.e. error in only one bit in all iterations of simulation 2 corresponds to \( \log_{10}(1/(4000\times3\times500)) \approx -6.78 \)) and when the curves disappear it means that the performance is error free.

From the same simulation we also get the following parameter study for the GTMH-precoding case. Here the block and bit error rates are plotted versus the SNR, but for all different values of \( \rho \) studied; this plot can be found in Figure 5.6. In this plot we can see that a lower \( \rho \) seems to need a higher SNR to perform good. The data is a bit inconclusive since the lines \( \rho = 0.75 \) and 0.725 are not following this rule.

5.4 Simulation 3; non-turbo-coded comparison of the precoding schemes

While the previous simulation was aimed at being as realistic as possible for mobile applications, applying the WCDMA turbo coding makes the comparison between the precodings less transparent. Hence for the sake of completeness a similar simulation was done, this time with block length of only 200 bits and with no turbo codes. This calls for higher SNR since the performance even in the non-FTN case is also greatly improved by turbo codes.

The result can be seen in Figures 5.7 and 5.8, these can then be compared with Figures 5.5 and 5.6 respectively.
5.4. SIMULATION 3; NON-TURBO-CODED COMPARISON OF THE PRECODING SCHEMES

Figure 5.5: Simulation 2; a performance comparison between two precodings and a non-precoded case. FTN is applied together with WCDMA-turbo-codes over an AWGN channel; the blocks consists of 6000 bits, but only 4000 bits input data.

Figure 5.6: Parameter study, how different ρ affect the error rates when using GTMH-precoding and WCDMA turbo codes. log₁₀ of the error rate plotted against the SNR. Lower ρ seems to implicate higher error rate for the same SNR.
Figure 5.7: Simulation 3; a performance comparison between two precodings and a non-precoded case. FTN is applied over an AWGN channel; the blocks consists of 200 bits. It seems like the non-precoded case is marginally better than the SVD-precoded case here.

Figure 5.8: Parameter study, how different $\rho$ affect the error rates when using GTMH-precoding. Here we can see that for $\rho \geq 0.8$ the curves seems to cluster a bit and between $\rho = 0.8$ and $\rho = 0.75$ there seems to be a larger jump in error rate.
From these plots it seems like without turbo codes it is better to not use any precoding rather than using SVD-precoding. This is however not the complete picture since for \( \rho \geq 0.9 \) the SVD-precoding is clearly better than no precoding. A reason for the sub-optimality of the SVD-precoding is probably due to uneven allocation of energy per symbol, and a water-filling algorithm might help to enhance the performance, this however falls beyond the scope of this investigation. Still the GTMH-precoding outperforms the other alternatives, and are regarded as the best.

5.5 Simulation 4; application to low latency communication, improving short code performance

In some applications the amount of data needed to be transmitted is small but the delay tolerated is also small. In this case it might be preferable to transfer small blocks of data since the receiver application cannot wait for larger sets of data to aggregate at the sender. Moreover, a large redundancy in coding might neither be tolerated because of time constraints. For these applications there is still a need for shorter codes with good performance [37]. The existing codes are asymptotically good, but for some applications this asymptote is not reached. FTN is not a short code alternative, it is not an error correcting code at all. Still it might provide an improvement to this situation and this simulation will show such a scenario.

In the following simulation all different schemes will send 400 bits of data from our input distribution, but the total number of bits transmitted is different from different schemes. It is still the WCDMA turbo codes applied, but the coding rates are varied along with the amount of FTN applied and what is kept constant in all transmissions is the total time and the number of data bits drawn from the input distribution. This means that a higher amount of FTN will allow for lower code rate. In the following setup the Nyquist case is starting on a coding rate of 2/3 and transmits a total of 600 bits. The result can be found in Figure 5.9. Here we can see that the error performance is better and better the more FTN and lower code rate we use. It is constant power FTN and GTMH-precoding that is used.

In one view this simulation shows the completely non-greedy way of FTN, the data rates were kept the same even though up to twice as many bits were transmitted. This can be compared with simulation 2 that uses the greedy way of FTN, where the code rate was kept the same and applying FTN essentially increased the amount of data transferred with a factor \( 1/\rho \).
Nyquist sending ($\rho = 1$), with high code rate (2/3)

FTN sending ($\rho = 6/7$), with GTMH–precoding and lower code rate (4/7)

FTN sending ($\rho = 0.75$), with GTMH–precoding and lower code rate (1/2)

FTN sending ($\rho = 0.5$), with GTMH–precoding and low code rate (1/3)

Figure 5.9: Simulation 4; FTN providing an improvement in the low latency case by instead of increasing the data rate one instead can decrease the code rate. The total amount of bits transmitted are 600, 700, 800, and 1200 respectively, all having the same total transmission time.
Here we derive the Szegö function for the root-raised-cosine. We start from the gram matrix (3.20),

\[
G_{m,n} = \frac{(1 + \beta)}{2} \cdot \frac{\rho}{1 - 4\beta^2 \rho^2 (m - n)^2} \cdot \text{sinc}((1 + \beta)\rho (m - n)) + \frac{(1 - \beta)}{2} \cdot \frac{\rho}{1 - 4\beta^2 \rho^2 (m - n)^2} \cdot \text{sinc}((1 - \beta)\rho (m - n)) ,
\]

and for technical reasons we assume \( \rho \leq 1 \).

With the use of Parseval's identities, the one regarding periodic convolution, as well as the formulas (1) and (10) from [15, pp. 311-314] we identify the sought Szegö function \( f(x) \) as the convolution of the two functions \( g_1(x) \) and \( g_2(x) \) with the corresponding Fourier series coefficients:

\[
c_n^1 = \frac{(1 + \beta)\rho}{2} \cdot \text{sinc}((1 + \beta)\rho n) + \frac{(1 - \beta)\rho}{2} \cdot \text{sinc}((1 - \beta)\rho n) \quad (A.1)
\]

and

\[
c_n^2 = \frac{-1}{4\beta^2 \rho^2 n^2 - 1} . \quad (A.2)
\]

Using the newly cited formula (1) and working the same way as for the sinc pulses we can conclude that \( g_1 \) is a sum of rectangular functions, periodic on the interval

\footnote{This is the case in FTN and also equivalent with \((1 - \beta)\rho \geq 2 - (1 + \beta)\rho \). Then with \( \beta \in [0, 1] \) as for root-raised-cosine, this also implies \((1 + \beta)\rho \leq 2 \).}
[−π, π] and looks like:

If (1 + β)ρ < 1

\[
\begin{align*}
g_1(x) &= \frac{1}{2} \left( \text{rect}[−(1 + β)ρπ, (1 + β)ρπ](x) + \text{rect}[−(1 − β)ρπ, (1 − β)ρπ](x) \right) \\
&= \text{rect}[−(1 − β)ρπ, (1 − β)ρπ](x) \\
&\quad + \frac{1}{2} \left( \text{rect}[−(1 + β)ρπ, −(1 − β)ρπ](x) + \text{rect}[(1 − β)ρπ, (1 + β)ρπ](x) \right)
\end{align*}
\]

If (1 + β)ρ ≥ 1

\[
\begin{align*}
g_1(x) &= \frac{1}{2} \left( 1 + \text{rect}[−π, −(2 − (1 + β)ρ)π](x) + \text{rect}[−(1 − β)ρπ, (1 − β)ρπ](x) \\
&\quad + \text{rect}[(2 − (1 + β)ρ)π, π](x) \right) \\
&= 1 - \frac{1}{2} \left( \text{rect}[−(2 − (1 + β)ρ)π, −(1 − β)ρπ](x) \\
&\quad + \text{rect}[(1 − β)ρπ, (2 − (1 + β)ρ)π](x) \right)
\end{align*}
\]

(A.3)

and is hence different depending on the value of the parameter combination (1 + β)ρ.

Then for \( g_2(x) \) we work in the light of the above cited formula (10) and try to find the Fourier series of the following function

\[
g_2^*(x) = a \cdot \sin \left( \frac{|x| − d}{2b} \right) \quad \text{on the interval } [−π, π], \quad (A.4)
\]

and find that it has the following coefficients:

\[
c_n^{2*} = \frac{2ab}{\pi} \cos \left( \frac{d}{2b} \right) \cdot \frac{−1}{4b^2n^2 − 1} + \frac{2ab(-1)^n}{\pi} \cdot \frac{−1}{4b^2n^2 − 1} \cos \left( \frac{π − d}{2b} \right). \quad (A.5)
\]

Comparing \( c_n^{2*} \) with \( c_n^2 \), that is (A.5) with (A.2), one can quickly identify\(^2\) that in our case we have \( b = βρ \), \( d = (1 − βρ)π \), and \( a = \frac{π}{2βρ} \cdot \frac{1}{\cos((1 − βρ)π/(2βρ))} \). Making some last rearrangements we can conclude that

\[
g_2(x) = \frac{π}{2βρ} \cdot \frac{1}{\sin(\frac{π}{2βρ})} \cdot \cos \left( \frac{|t| − π}{2βρ} \right) \quad (A.6)
\]

Now the searched function for applying Szegő’s theorem is found as

\[
f(x) = \frac{1}{2π} \int_{−π}^{π} g_1(x − t)g_2(t) \, dt.
\]

\(^2\)In \( c_n^{2*} \) the second term must disappear and all coefficients of the first term must be = 1.
This is solved by interchanging $g_1$ and $g_2$ due to the symmetry of the convolution and then solve the following integral for the three cases $x \geq b$, $x \in (a, b)$, and $x \leq a$ respectively, since $g_1(x)$ is constant over intervals.

$$\int_a^b \cos \left( \frac{|x-t|-\pi}{2\beta \rho} \right) \, dt =$$

$$\begin{cases} 
2\beta \rho \left( -\sin \left( \frac{x-\pi-b}{2\beta \rho} \right) + \sin \left( \frac{x-\pi-a}{2\beta \rho} \right) \right) & \text{for } x \geq b \\
2\beta \rho \left( \sin \left( \frac{-x+\pi+b}{2\beta \rho} \right) + \sin \left( \frac{x-\pi-a}{2\beta \rho} \right) + 2 \sin \left( \frac{\pi}{2\beta \rho} \right) \right) & \text{for } x \in (a, b) \\
2\beta \rho \left( \sin \left( \frac{-x+\pi+b}{2\beta \rho} \right) - \sin \left( \frac{-x-\pi+a}{2\beta \rho} \right) \right) & \text{for } x \leq a 
\end{cases}$$

(A.7)

The rest is merely to check the different cases $(1 + \beta)\rho \leq 1$ and $(1 + \beta)\rho \geq 1$. Combining the results (A.3), (A.6) and (A.7) gives, after quite some calculations and a lot of usage of [15, p 127], the sought function as:

If $(1 + \beta)\rho \leq 1$

$$f(x) = \begin{cases} 
0 & x \in [-\pi, -(1 + \beta)\rho \pi] \\
\frac{1}{2} \left( 1 + \sin \left( \frac{x + \rho \pi}{2\beta \rho} \right) \right) & x \in [-(1 + \beta)\rho \pi, -(1 - \beta)\rho \pi] \\
1 & x \in [-(1 - \beta)\rho \pi, (1 - \beta)\rho \pi] \\
\frac{1}{2} \left( 1 - \sin \left( \frac{x - \rho \pi}{2\beta \rho} \right) \right) & x \in [(1 - \beta)\rho \pi, (1 + \beta)\rho \pi] \\
0 & x \in [(1 + \beta)\rho \pi, \pi]
\end{cases}$$

If $(1 + \beta)\rho \geq 1$

$$f(x) = \begin{cases} 
1 - \sin \left( \frac{\pi(1 - \rho)}{2\beta \rho} \right) \cos \left( \frac{x + \pi}{2\beta \rho} \right) & x \in [-\pi, -(2 - (1 + \beta)\rho)\pi] \\
\frac{1}{2} \left( 1 + \sin \left( \frac{x + \rho \pi}{2\beta \rho} \right) \right) & x \in [-(2 - (1 + \beta)\rho)\pi, -(1 - \beta)\rho \pi] \\
1 & x \in [-(1 - \beta)\rho \pi, (1 - \beta)\rho \pi] \\
\frac{1}{2} \left( 1 - \sin \left( \frac{x - \rho \pi}{2\beta \rho} \right) \right) & x \in [(1 - \beta)\rho \pi, (2 - (1 + \beta)\rho)\pi] \\
1 - \sin \left( \frac{\pi(1 - \rho)}{2\beta \rho} \right) \cos \left( \frac{x - \pi}{2\beta \rho} \right) & x \in [(2 - (1 + \beta)\rho)\pi, \pi]
\end{cases}$$

(A.8)

This is the Szegö function to (3.20) as given in (3.22).
A few notes about this function are in place here. Although $f$ might seem discontinuous it is not. For all the endpoints of intervals on which it is defined, the left and right limits are the same, but the derivative is of course not continuous in these points. The same holds for the parameter combination $(1 + \beta)\rho$. In the transition at $(1 + \beta)\rho = 1$ the top most definition reduces to being zero at the point $x = \pm\pi$ (once again regardless if one uses the interval $x = \pm\pi$ or $x \in [\pm\pi, \pm(1 + \beta)\rho\pi]$) and the lower definition of the function does the same. Hence $f$ is continuous in all variables and parameters.
Appendix B

Eigenvalue spectra of the Gram matrix and frequency spectra of the pulse

The Gram matrix and especially the Szegő function has proven tedious to derive for some pulses, look for example at Appendix A, therefore the following theorem is useful. It was pointed out to me by Johan Karlsson.

**Theorem 15** (Szegő function from Frequency spectra). Let \( \{ h_T^n(t) \}_{n=1}^N \) be some set of pulses generated from an energy limited and \( T \)-orthogonal pulse shape that is time shifted integer times \( \rho T \). Let \( h(t) \) have the Fourier transform \( H(\omega) \) and let it have finite support. Moreover let \( G \) be their Gram matrix and \( f(\theta) \) the corresponding Szegő function. If this set of pulses is energy normalized so that \( \forall \rho \in (0, \infty) \) the expected energy per block is the same as in the Nyquist case (that is \( \text{trace}[G] = \rho N \)). Then

\[
 f(\theta) = \frac{1}{T} \left| H \left( \frac{\theta}{\rho T} \right) \right|^2 , \tag{B.1}
\]

where \( f(\theta) \) is periodic on the interval \([-\pi, \pi]\) and the Fourier transform is folded around \( \pi/\rho T \).

**Proof.** First of all we establish that \( G \) is a symmetric Toeplitz matrix. But this is easily done by observing that \( G_{m,n} = \langle h_T(t - m\rho T), h(t - n\rho T) \rangle = \langle h_T(t - (m - n)\rho T), h(t) \rangle = \langle h_T(t), h(t - (n - m)\rho T) \rangle \) and the entries in \( G \) are only dependent on the absolute difference between \( m \) and \( n \).

Now we know that \( f(\theta) \) is the function generating the Fourier series, whose
coefficients form the row of $G$. Looking at a Fourier coefficient we see that:

$$c_k = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(\gamma) e^{-ik\gamma} \, d\gamma$$

$$= \{ \text{choose } n, m \in \mathbb{Z} \text{ such that } k = m - n \} = G_{m,n}$$

$$= \langle h^m_T, h^n_T \rangle = \int_{-\infty}^{\infty} \sqrt{\rho} h(t - m\rho T) \cdot \sqrt{\rho} h(t - n\rho T) \, dt$$

$$= \frac{\rho}{2\pi} \int_{-\infty}^{\infty} H(\omega) e^{-i\omega m\rho T} \cdot H(\omega) e^{-i\omega n\rho T} \, d\omega = \frac{\rho}{2\pi} \int_{-\infty}^{\infty} |H(\omega)|^2 e^{-i\omega(m-n)\rho T} \, d\omega$$

$$= \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{1}{T} \left| H \left( \frac{\theta}{\rho T} \right) \right|^2 e^{-i\theta(m-n)} \, d\theta .$$

Hence we see that the function generating the Fourier series $1/T \cdot |H(\theta/\rho T)|^2$ and from uniqueness of Fourier series we know that $f(\theta) = 1/T \cdot |H(\theta/\rho T)|^2$, except possibly at a set of measure zero. The folding part comes from the fact that $e^{i\theta(n-m)}$ is $2\pi$-periodic and hence if $H(\theta)$ has support outside $[-\pi/\rho T, \pi/\rho T]$ it is going to be folded.
Appendix C

Saltzberg formulation, the FTN dual of Mazo formulation

In this thesis the focus has been on packing the data pulses tighter in time. That is the normal approach to faster-than-Nyquist signaling and how Mazo presented it in [24]. Here I would like to present a different view of FTN, that is although mathematically equivalent, and it is the view first presented on the topic by Saltzberg in [23]. Instead of packing tighter in time to increase data rates one could keep the same data rates and decrease the frequency band. When the whole frequency spectrum (Fourier transform) domain is scaled with \( \rho \) the mathematical problem would still be the same, but the practical aspects would be different. Instead of potentially increasing data rates in existing channels one would decrease the frequency spectrum usage, which is an expensive resource.

Another difference is the energy normalization, since the Saltzberg formulation is not trying to send more data in a given time the power in the sender remains unchanged and hence no \( \sqrt{\rho} \) is needed in the pulses. Unless of course the extra spectrum is used to create a new frequency channel, then an energy normalization at the base station level would be needed to keep the transmitted power constant. This approach could be used to make room for more channels and in the end still result in an increase in data rate for the end user, but the implementation would be different.

The theoretical problems of ISI are equivalent but in practical realizations they might be different. With the Saltzberg formulation one is freeing spectrum rather than using existing spectrum more efficient. What is done with the freed spectrum is then another, although related, question. Observe however that applications like the one in simulation 4 are in Mazo formulation, since they make use of the increased bit density in the transmission. There could however be other benefits of Saltzberg formulation, with the creation of a new independent channel in frequency domain more users could be served simultaneously.
Bibliography


