Tomlinson Harashima Precoding for Multi-Gigabit short-haul Transmission over Plastical Optical Fibers

Laura Caballero Nadales

September 7, 2010
Acknowledgments

Thanks to my family and the people who has been with me in the worst moments of my life, for you.
Resumen

En este proyecto, el sistema de Precodificación Tomlinson-Harashima es estudiado e implementado mediante un programa en Matlab con el objetivo de eliminar la ISI en nuestro canal. Nuestro canal es característico de las POF (Plastic Optical Fiber) y tiene ciertas características por las cuales hay que tener muy en cuenta a la hora de aplicar el THP. El uso del THP es perfectamente justificable si se tienen en cuenta las grandes ventajas que ofrece en contraposición a otras técnicas como el DFE. El THP puede trabajar en conjunción con el Reed Solomon code para mejorar así la BER. Más características del RS són presentadas a lo largo de la memoria y finalmente en la última parte, las simulaciones del THP y RS són mostradas y posteriormente razonadas.
Abstract

In this dissertation, the Tomlinson-Harashima is presented and implemented by a Matlab program with the objective to eliminate the ISI in our channel. Our channel is characteristic by the use of Plastical Optical Fibers (POF) and the channel has certain features that should be taken into account for the THP application. The use of THP is justified because of the great advantages that it presents over other techniques such as DFE, THP can work in conjunction with the Reed Solomon to further decrease the Bit Error Ratio. More characteristics of RS are presented along the dissertation and in the final part RS is combined with THP and the performance results are shown.
Contents

1 Backgrounds of Tomlinson Harashima ......................................................... 9
   1.1 Introduction .................................................................................... 9
   1.2 Decision-Feedback Equalization ....................................................... 9
   1.3 Precoding Schemes ........................................................................ 12
   1.4 Tomlinson Harashima Precoding ..................................................... 13
      1.4.1 History .................................................................................. 13
      1.4.2 Definition .............................................................................. 13
      1.4.3 Mathematical concept ............................................................ 14
      1.4.4 Losses of Tomlinson-Harashima Precoding ............................. 19
      1.4.5 THP under different criterions ................................................. 20

2 System ....................................................................................................... 22
   2.1 Channel .......................................................................................... 23
      2.1.1 Polymer Optical fibers ......................................................... 24

3 Simulation results for THP ...................................................................... 27
   3.1 Comparison .................................................................................... 31
   3.2 Reasons .......................................................................................... 31
      3.2.1 Losses .................................................................................. 31
      3.2.2 Channel typology ................................................................. 32

4 Backgrounds Reed Solomon .................................................................. 35
   4.1 Introduction .................................................................................... 35
## Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BER</td>
<td>Bit Error Ratio</td>
</tr>
<tr>
<td>DFE</td>
<td>Decision Feedback Equalizer</td>
</tr>
<tr>
<td>EDS</td>
<td>Effective Data Sequence</td>
</tr>
<tr>
<td>FEC</td>
<td>Forward Error Correction</td>
</tr>
<tr>
<td>ISI</td>
<td>Intersymbol Interference</td>
</tr>
<tr>
<td>MMSE</td>
<td>Minimum Mean Square Error</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multiple Input Multiple Output</td>
</tr>
<tr>
<td>PAM</td>
<td>Pulse Amplitud Modulation</td>
</tr>
<tr>
<td>POF</td>
<td>Polymer Optical Fiber</td>
</tr>
<tr>
<td>RS</td>
<td>Reed-Solomon</td>
</tr>
<tr>
<td>TH</td>
<td>Tomlinson-Harashima</td>
</tr>
<tr>
<td>SER</td>
<td>Symbol Error Ratio</td>
</tr>
<tr>
<td>SISO</td>
<td>Single Input Single Output</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise-Ratio</td>
</tr>
<tr>
<td>ZF</td>
<td>ZeroForcing</td>
</tr>
</tbody>
</table>
Chapter 1

Backgrounds of Tomlinson Harashima

1.1 Introduction

To overcome the multipath effect (ISI-Intersymbol Interference-) and achieve high throughput transmission, channel equalization or precoding techniques can be used. However, in linear equalization there are some limitations, for example in case of severe ISI the performance of the linear equalizer is limited and suffers from noise enhancement. Thus, this dissertation will show how to improve the system performance by nonlinear equalization. The most commonly used nonlinear equalizer is the Decision-Feedback Equalizer, but it has some drawbacks, thus Tomlinson-Harashima precoding technique is introduced in this dissertation as the main option to overcome the drawbacks of DFE.

1.2 Decision-Feedback Equalization

The Decision Feedback Equalizer (DFE) is the improved equalizer according to the linear equalizer by the introduced nonlinearity \[ \Pi \]. The DFE has a noise reduction
A disadvantage of DFE is that erroneous decisions lead to residual postcursive ISI and this can possibly result in further decision errors, a phenomenon known as error propagation (EP) in its feedback loop. Error propagation is exacerbated when DFE is combined with coded modulation techniques using dense constellations. As the constellation distances become smaller, symbol-by-symbol decisions become less accurate. In our case, MPAM is considered, whose constellation distances decreases when M increases, causing a severer error propagation.

DFE is composed of two filters: The feedforward filter and feedback filter. Both are implemented at, in general, symbol rate. In the DFE, the input from the channel is passed through a feedforward filter. The output of the feedback filter is then subtracted from the output of the feedforward filter.

The block diagram of the DFE is given in Figure 1.1:

![Block diagram of DFE](image)

**Figure 1.1: Block diagram of DFE**
The feedforward filter removes some of the ISI from the received signal, but leaves some of the postcursor ISI on the signal (generally, all postcursor ISI). The feedback filter estimates the residual ISI from the past decisions and subtracts it from the feedforward filter output. The DFE solution is better than linear equalizer with a low-complexity solution.

The low noise enhancement of the DFE arises from the fact that, by assuming no decision errors, the decision device removes all the noise present in the signal \[^1\]. So, if the inputs of the feedback filter have no noise then the outputs of the feedback filter have no noise. Also, the feedforward filter has the less complex problem of removing only precursor coefficients. This results in a better performance for the feedforward filter comparing to the linear equalizer. But, the assumption of correct symbol decisions at the output of the decision device may not work in practical cases, so error propagation occurs and the performance of the equalizer degrades.

To cope with these problems exposed before, the precoding technique is introduced.
1.3 Precoding Schemes

The original principle of precoding is that if transmitter knows the channel information, we can design the transmit signal so that the ISI in the receiver side is greatly mitigated. For instance, the use of Tomlinson-Harashima Precoding can be regarded as moving the feedback filter of DFE to the transmit side to avoid error propagation problem.

In this section precoding techniques are discussed, by employing precoding, the disadvantages of DFE can be avoided: Coding techniques can be applied in the same way as for channels without ISI, and no error propagation occurs.

These techniques are always apply if known in advance by the transmitter, the channel transfer function. Although the channel is not fully known by the transmitter, the use of precoding is still correct but with a set of compromises, such as removing the residual intersymbol interference (caused by estimation errors) through the linear adaptive equalization in the receiver.

Basically, two precoding techniques are known: Tomlinson-Harashima precoding (THP), which was proposed almost 30 years ago, and flexible precoding (FLP)[5], developed recently during the standards activities for the international telephone line modem standard ITU V.34.

In contrast to THP which is derived from linear preequalization at the transmitter, flexible precoding resembles linear equalization at the receiver side. The disadvantage of linear equalization is that the channel noise is filtered with $\frac{1}{H(z)}$, too, and thus the desired prediction gain is lost.
1.4 Tomlinson Harashima Precoding

1.4.1 History

Tomlinson-Harashima precoding (THP) was invented independently and more or less simultaneously (1968-1969) in theses by Tomlinson in the United Kingdom and Harashima in Japan. They applied THP to a SISO system without adaptive receive filter in order to suppress the ISI caused by the frequency selectivity of the channel, since the recursive filter necessary to equalize the channel can be unstable.

Later, Ginis and Fischer proposed Spatial THP without ordering for flat fading MIMO channels. Whereas Ginis included a feedforward filter in the transmitter and assumed a receive filter which is a diagonal matrix, Fischer investigated a system with the feedforward filter at the receiver. Further Fischer and Simeone were the only ones to investigate the THP with partial channel state information at the transmitter.

Finally, Joham presented the necessary optimizations for THP with FIR feedforward and feedback filters for frequency selective MIMO channels and Fischer designed THP for frequency selective MIMO channels with IIR feedforward filter by applying a spectral factorization of the channel transfer function.

1.4.2 Definition

Tomlinson-Harashima precoding is a transmitter equalization technique where equalization is performed at the transmitter side, and has been widely used in many applications, such as DSL systems, voice band and cable modems. It can eliminate error propagation by moving the FBF of DFE to the transmitter and allow us to use current capacity-achieving channel codes, such as low-density parity-check (LDPC) codes in a natural way. Recently, TH precoding has been proposed to be used in 10 Gigabit Ethernet over copper (10 GBASE-T)[2].
1.4.3 Mathematical concept

Tomlinson-Harashima precoding was originally proposed for use with an M-point one-dimensional PAM signal set \( A = \pm 1, \pm 2, \pm 3, \ldots, \pm (M - 1) \). For this constellation THP is almost identical to the inverse channel filter \( \frac{1}{H(z)} \), except that an offset-free (symmetrical about the origin) modulo-2M adder is used instead of the conventional adder.

![Figure 1.2: Output vs Input for the modulo adder.][12]

Figure 1.2 shows an example plot of the function employed by the modulo adder for \( M = 4 \). The modulo adder over a fixed interval of, say, \([-M, +M)\) implements the following algorithm:

- If the result of the summation, \( x(k) \) is greater than \( M \), \( 2M \) is deducted from it until it is less than \( M \).
- If the result of the summation, \( x(k) \) is less than \(-M \), \( 2M \) is added to it until it is greater than or equal to \(-M \).

The block diagram of the Tomlinson-Harashima precoder using a sawtooth non-
1. Backgrounds of Tomlinson Harashima

Linearity for modulo reduction is sketched in Figure 1.3 [4].

![Figure 1.3: Tomlinson-Harashima precoder and linearized description](image)

As we see that the TH structure could be conceived in two ways. The first way is: in the left part of the figure there is a sequence $a[k]$ that is deducted by $f[k]$. The $f[k]$ is the result of filtering the output sequence $x[k]$ with the channel impulse response subtracted by one unit in the first element. At this point the result of the addition is subjected by the explained modulo adder and, thus $x[k]$ can be obtained.

The second manner to understand the TH structure is as follows: unique sequence $d[k] \in 2MZ$ (considering $Z$ as an integer number) is added to the data sequence $a[k]$ in order to create an effective data sequence (EDS) $v[k]$, with $v[k] = a[k] + d[k]$, $v[k]$ is then filtered with the inverse of $H(z)$.

Below shows the mathematical relationship between the sequences:

$$x[k] = a[k] + d[k] - \sum_{K=1}^{P} h[K] \cdot x[k - K] = v[k] - f[k]$$

The values $d[k]$ are implicitly selected symbol-by-symbol by the memoryless modulo operation, which reduces $x[k]$ to the interval $[-M, +M]$.
The principle of modulo precoding can be understood as a multiple-symbol representation based on congruent signal levels. The congruent signal points are generated by extending the signal set $A$ periodically to the set $[A]$. 

$$V \equiv A + 2MZ = a + d \mid a \in A, \ d \in 2MZ$$  \hspace{1cm} (1.2) 

Next figure shows the extended signal set $V$:

![Extended signal set V](image)

Figure 1.4: Extended signal set $V$ with $M = 4$.

The THP uses the signal $V$ set instead of $A$, hence THP is an extension of linear preequalization. The current effective data symbol is selected in THP, which is congruent to the current $a[k]$, and minimizes the magnitude of the corresponding channel symbol $x[k]$. It is noteworthy that THP not only has to be matched to the channel, but also is closely tied to the actual signal constellation $A$. 

---

Institut für Nachrichtenübertragung  
Universität Stuttgart
Next Figure 1.5 shows the complete scheme for a transmission with THP:

Figure 1.5: Complete scheme for a transmission with THP.

Following we can conclude that the receiver output sequence is given by:

\[
    r[k] = v'[k] = \sum_{K=0}^{P} h[K] \cdot x[k - K] + n[k] = x[k] + \sum_{K=1}^{P} h[K] \cdot x[k - K] + n[k] = x[k] + f[k] + n[k] = v[k] + f[k] + n[k] = v[k] + n[k] \quad (1.3)
\]

where \( n[k] \) is again the white Gaussian noise sequence.

It can be easily deduced, that in absence of noise, \( v[k] \) can be recovered directly before the entry of the last Mod 2M, then the output \( v'[k] \) would be reduced to the
range $R=[-M, M)$ by the modulo reductor without any ISI.

**Theoretical SER of THP**

The probability of a correct decision of $a'[k]$ can be written as:

$$P_c = \frac{1}{M} \sum_{p \in A} \sum_{s=-\infty}^{\infty} Pr \{z[k] = s | a[k] = p\} \cdot Pr \{a'[k] = p | a[k] = p, z[k] = s\} \quad (1.4)$$

where $z[k]$ is the integer element that restrict $x[k]$ within the interval $[-M, M)$, and $p$ is a particular symbol in signal set $A$. Finally, $s$ is considered as an integer number.

The decision variable $y[k]$ is obtained by continuously folding the sample $(a[k] + 2Mz[k] + f[k] + n[k])$ in the interval $[-M, M)$. It that suggest that

$$Pr = \{a'[k] = p | a[k] = p, z[k] = s\} = Pr \{p - 1 < y[k] > p + 1 | a[k] = p, z[k] = s\} =$$

$$\sum_{m=-\infty}^{\infty} Pr \{2(m - s)M - 1 < f[k] + n[k] \leq 2(m - s)M + 1 | a[k] = p, z[k] = s\} =$$

$$\sum_{m=-\infty}^{\infty} Pr \{2mM - 1 < f[k] + n[k] < 2mM + 1\} \quad (1.5)$$

The final step in the above equation is obtained by using the following facts: the summation is made from $m = -\infty$ to $\infty$ so that the integer $s$ has no effect on the resultant value and the inequality $2(m - s)M - 1 < f[k] + n[k] \leq 2(m - s)M + 1$ is free from the condition of $a[k] = p$ and $z[k] = s$. 
Now substituting, it yields:

\[ P_c = \sum_{m=-\infty}^{\infty} Pr \{ 2(m-s)M - 1 < f[k] + n[k] \leq 2(m-s)M + 1 \} \]

\[ \cdot \left[ \frac{1}{M} \sum_{p \in A} \sum_{s=-\infty}^{\infty} Pr \{ z[k] = s | a[k] = p \} \right] \quad (1.6) \]

Note that the last term in brackets is equal to 1, then the SER of THP is:

\[ P_s = 1 - P_c = 1 - \sum_{m=-\infty}^{\infty} \int_{2mM-1}^{2mM+1} f_\epsilon(x) dx \quad (1.7) \]

where \( m \) is an integer, and \( f_\epsilon(x) \) is the probability density function of the disturbance \( \epsilon[k] = f[k] + n[k] \).

### 1.4.4 Losses of Tomlinson-Harashima Precoding

There are some losses limiting the performance of THP:

- **Modulo loss:**

  To bind the transmission power, a modulo operation is used at the transmitter. Due to the modulo operation at the receiver will be modulo errors because when the noise is large this modulo may flip signals to the wrong side of the signal constellation and causes errors. These errors are referred to as modulo errors.

- **Shaping loss:**
1. Backgrounds of Tomlinson Harashima

The output is uniform rather than Gaussian. This corresponds to a loss of
\[ \log_2 \left( \frac{2\pi e}{12} \right) \approx 1.53 dB \] at high SNR, and a loss of more than 4dB at low SNR.

- Power loss:

Compared to signaling using the data symbols \( a[k] \), Tomlinson-Harashima
precoding slightly increases average transmit power. This precoding loss is
only relevant for “small” signal sets and vanishes completely as the number \( M \)
of signal points goes to infinity. Additionally, the number of nearest neighbor
points increases slightly due to the periodic extension of the constellation.

1.4.5 THP under different criterions

The precoding processing, can be optimized to satisfy a ZF or a MMSE criterion.
This is the structure of THP based on ZF and MMSE criterion:

\[ B(D) \text{ and } F(D) \text{ correspond to the feedback and feedforward filters of THP de-} \]
\[ \text{signed to minimize either ZF or MMSE optimaly criteria. } B(D) \text{ is closely related} \]
\[ \text{to the feedback filter of a DFE. } F(D) \text{ must to be linear and time-invariant and} \]

\[ \text{Figure 1.6: Communication system using THP with ZF and DFE} \]
corresponds to the combination of the sampled matched filter and feedforward filter in a DFE [13].

In the context of THP, zero-forcing implies forcing $y_k = 0$ for all $k$. Spectral factorization techniques satisfy this criterion by producing an allpass $F(D)$ such that $H(D)F(D)$ is causal, monic and minimum phase (minimum delay). $B(D)$ is chosen to equal $H(D)F(D)$ [13].

The MMSE-THP is obtained by choosing $F(D)$ and $B(D)$ to minimize $\text{VAR}(\hat{n}_k + y_k)$ under the constraints that $B(D)$ and $H(D)F(D)$ are monic. As with the ZF-THP, spectral factorization techniques provide the desired filters, which are exactly the values of $B(D)$ and $F(D)$ used in MMSE-DFE. Unlike the ZF-THP, $H(D)F(D)$ need not be causal or minimum phase, and $F(D)$ need not be allpass [13].
Chapter 2

System

Our system is formed by: first of all a Reed Solomon encoder, the following part consist of a modulation block (PAM modulation), then this modulated signal goes through a Tomlinson-Harashima Precoding block, afterwards the output’s signal is filtered by the channel and then the demodulation and RS decodification are applied. This figure allows the understanding of the system:

![System scheme](image)

Figure 2.1: System scheme
2.1 Channel

The channel is not monic and obtained from the use of Plastic Optical Fiber in our system, it has this mathematical expression:

\[ h = \frac{A}{\sqrt{2\pi}\sigma} \cdot e^{-\frac{(t-\tau \cdot Lpof)^2}{2\sigma^2}} \]  \hspace{1cm} (2.1)

Here, some parameters are defined:

\( \sigma = 0.132/B_{3dB} \rightarrow \) fiber dispersion

\( \tau = 4.97 \cdot 10^{-9} \text{s/m} \rightarrow \) group delay

\( Lpof = 10/15 \text{m} \rightarrow \) POF length in meter

and it is represented as follows:

![Figure 2.2: Channel impulse response](image)

The most important feature of this channel is the fact that it varies little with time, this could be related with the requirement of Tomlinson-Harashima Pre-
2. System coding concerning always is applied if the channel is known in advance by the transmitter. Hence, if the channel-impulse response is not so variable, THP is useful in this situation because it is supposed that the channel transfer function is the 'same' all the time.

Because of this channel is directly related with Polymer Optical fibers, it would be appropriate to introduce them in the theoretical way to get a rough idea of the environment in which we work.

2.1.1 Polymer Optical fibers

In the last decade, in optical communication networks, Polymer optical fibers (POF) are become of a great interest. They are based on the theory of generalized impulse response of multimode channels. There are two types of POF depending if they have been manufactured with step-index structure (SI-POF) (used in our system) or gradual index (GI-POF).

The core material of a POF is known as PMMA (acrylic), and fluorinated polymers are the cladding material, but since the late 1990s however, much higher-performance POF based on perfluorinated polymers (mainly polyperfluorobutenylvinylether) has begun to appear in the marketplace.

In large-diameter fibers, 96% of the cross section is the core that allows the transmission of light. Similar to traditional glass fiber, POF transmits light (or data) through the core of the fiber. The core size of POF is in some cases 100 times larger than glass fiber.

Because the links, connectors, and installation of POF is inexpensive, is given the name of the "consumer" of optical fiber. PMMA system is used in environments of

Institut für Nachrichtenübertragung
Universität Stuttgart
low-speed transmission and short distances, especially in digital home devices, home networks, industrial networks and car networks. In contrast, the perfluorinated polymer fibers are commonly used for much higher-speed data center applications such as LAN wiring and building wiring.

In recent years, POF has begun to consider as a possible option for high-speed home networking to the next generation of Gigabit/s. So European projects such as POF-ALL and POF-PLUS have been initiated to further investigation on this type of fiber.

The appearance of mPOF (microstructured POF), a type of a photonic crystal fiber, is a great development in polymer fibers field.

These are some characteristics of POF in short:[10]

- PMMA and Polystyrene are used as fiber core, with refractive indices of 1.49 and 1.59 respectively.
- Generally, fiber cladding is made of silicone resin (refractive index 1.46).
- High refractive index difference is maintained between core and cladding.
- POF have high numerical aperture.
- Have high mechanical flexibility and low cost.

Modal dispersion

Modal dispersion is a distortion mechanism occurring in multimode fibers and other waveguides, in which the signal is spread in time because the propagation velocity of the optical signal is not the same for all modes. Dispersion is sometimes called chromatic dispersion to emphasize its wavelength-dependent nature, or group-velocity
2. System

dispersion (GVD) to emphasize the role of the group velocity. This modal dispersion creates the limiting factor in POF systems, and this is usually the bandwidth of the fiber itself.

This kind of dispersion is also considered as a type of intersymbol interference (ISI). In the ray optics analogy, modal dispersion in a step-index optical fiber may be compared to multipath propagation of a radio signal. Rays of light enter the fiber with different angles to the fiber axis, up to the fiber’s acceptance angle. Rays that enter with a shallower angle travel by a more direct path, and arrive sooner than rays that enter at a steeper angle. The arrival of different components of the signal at different times distorts the shape.
Chapter 3

Simulation results for THP

In this chapter, I have written a program developed in MATLAB (see Appendix B) where the Tomlinson-Harashima Precoding is used. The aim of this program is to eliminate the ISI in our channel, and then, to improve the general system performance.

First of all, it is good to get an idea of the schematic structure of the program before to show the simulation results. In the first lines, the channel features and some simulation parameters are defined. The most featured simulation parameters are the $R_b$ that is the bit rate and $M$ which corresponds to the order of modulation. Afterwards, the channel impulse response seen in the last system chapter is defined.

Then we go into the main part of the program, where a bit sequence is created and subjected to a PAM modulation. In this case the $M = 8 = 2^3$ is chosen, the modulation is used because due to every symbol carries $n$ bit, the required bandwidth is reduced by a factor of $1/n$ and thus the noise as well. A lower noise level is required because the many signal levels are much closer spaced. The advantages of PAM are its flexibility and adaptability to the actual SNR. Higher the SNR is available, the more bits per symbol can be transmitted to raise the transmission capacity.
After modulation, the modulated signal crosses through the Tomlinson-Harashima Precoding block, and when it is recovered, the demodulation is the next step.

Next diagram represents the blocks in the THP program developed:

![THP Program Structure](image)

Figure 3.1: THP program structure

Finally the recovered sequence is compared with the initial one, and we can obtain the $BER$ (Bit error rate), depending on the $SNR$ (Signal-to-Noise ratio).

In the Tomlinson-harashima Precoding block, in our case, it is not possible to directly apply the theoretical scheme found in the bibliography because the channel is not monic and then the first element $chnl(0)$ is not equal to 1. Only if the channel is normalized by $chnl(0)$, the original scheme of THP could be used.

Now, the graphs resulting from the simulation of Tomlinson-Harashima program with different bit rates are shown below:
3. Simulation results for THP

Figure 3.2: BER performance for different bit rates with THP.

As it is seen the BER performance with 3Gbit/s and in 2.5Gbit/s is worse than in the case of 2Gbit/s. When the bit rate increases, the performance gets worse.

The next three graphs show the performance of DFE [11], they are useful to compare with our final results:

Figure 3.3: BER performance of 8 PAM with DFE for 2Gbit/s [11]
3. Simulation results for THP

Figure 3.4: BER performance of 8 PAM with DFE for 2.5 Gbit/s [11]

Figure 3.5: BER performance of 8 PAM with DFE for 3 Gbit/s [11]
3. Simulation results for THP

3.1 Comparison

At first these results seem contradictory if we consider the theory exposed in the preceding chapters, where for instance, the use of Tomlinson-Harashima Precoding can be regarded as moving the feedback filter of DFE to the transmit side to avoid error propagation problem. Therefore, THP performance is supposed to have better BER performance than the DFE. As it can be seen, this is not true in our case, the DFE performance is better than THP. For example for the BER value of $10^{-2}$ the DFE needs only 19.7dB to reach it, however with THP, 35dB of SNR is needed. In both cases it is possible to observe one thing in common, when the bit rate increases both techniques perform worse than in lower bit rate.

3.2 Reasons

Instead of the theoretical results expected based on THP and DFE performance, there are possible reasons why these assumptions are not true in this dissertation:

3.2.1 Losses

One of these reasons could be the different types of losses seen in the Backgrounds of THP: shaping loss, power loss and modulo loss.

In the particular case of the shaping loss, as we have explained in the Backgrounds of THP is caused by the fact that the channel's output is uniform rather than gaussian.

The power loss can be neglected because: since what TH changed is the transmit signal power, so if we consider the SNR at the transmitter side, TH would cause additional lost. However, we consider here the SNR at receiver, which means, we

Institut für Nachrichtenübertragung Universität Stuttgart
3. Simulation results for THP

generated the same SNR for both schemes (THP & DFE) at the receiver, since we measured the received signal power, then add noise according to the SNR at receiver. It doesn’t take the transmit power into account.

In case of modulo loss, is caused by the errors that appear after the modulo reduction which are the consequence when the noise is large. Related with these modulo errors we could introduce the following section for justify the performance achieved.

3.2.2 Channel typology

In order to understand why these modulo errors have importance in our BER performance in the case of the THP use, we introduce a monic channel with the first tap as a maximum:

\[ \text{chnl} = [1.0000 0.3011] \]

Now we can get the simulation results after the use of this monic channel:

![Figure 3.6: BER performance for 8 PAM with TH for new monolic channel](image)

Institut für Nachrichtenübertragung Universität Stuttgart
In fact, the performance becomes better when the first element of the monic channel is maximum and of course the performance would be much better if chnl(0) is far more bigger than the other taps.

Comparing the DFE at 2Gbit/s with the response of our THP with the monic channel, it is possible to compare some specific values:

<table>
<thead>
<tr>
<th>BER</th>
<th>10^{-2}</th>
<th>10^{-3}</th>
<th>10^{-4}</th>
<th>10^{-5}</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFE</td>
<td>21,3dB</td>
<td>23,5dB</td>
<td>25,7dB</td>
<td>30,7dB</td>
</tr>
<tr>
<td>THP</td>
<td>21,7dB</td>
<td>25,8dB</td>
<td>28,4dB</td>
<td>30,1dB</td>
</tr>
</tbody>
</table>

As it is seen the BER performances with both techniques are similar, but maybe in the lower SNR the BER performance of DFE is a little better because of the losses exposed before (see Losses in Backgrounds of THP), but from 30dB approximately the THP performance is better than DFE.

The reason why THP performs better with this monic channel, is related with the fact that when the $x$ precursors ($x[k - K]$) are filtered in THP block to form the $\bar{f}$ signal, if the samples of the channel are bigger than the first sample, the ISI increases ($\bar{f}$), and $\bar{x}$ is going to be reduced into the interval $[-M, M)$ by the modulo 2M reduction much more times ($\bar{d}$ increases). Then, in filter’s output, despite that $\bar{y}$ has no ISI, $\bar{y}$ is equivalent to the $\bar{a}$ but affected by the signal that has reduced the $\bar{x}$ (in the first modulo reduction), which that corresponds to $\bar{d}$. If this number of reductions is high, the noise added is proportional depending on the output $\bar{y}$ and the SNR is as follows:

$$SNR = \frac{E[\bar{a} + \bar{d}]}{E[\bar{n}]}$$  \hspace{1cm} (3.1)

The next diagram allows to understand in a better way what is happens:
3. Simulation results for THP

\[ f[k] = \sum_{k=1}^{p} h[K] x[k-K] \]

If channel samples bigger than first channel sample (case of POF channel)

\[ \bar{f} \uparrow \rightarrow x[k] = a[k] + d[k] - f[k] \]

Noise added after the output is bigger in case of hpoF channel because:

\[ y[k] = a[k] + d[k] \leftarrow \bar{d} \uparrow \]

\[ \bar{d}_{hpoF} > \bar{d}_{monic} \]

\[ SNR = \frac{E[\bar{a} + \bar{d}]}{E[n]} \]

The noise is proportional to the output

Figure 3.7: Diagram for explanation

hence, when the second modulo reduction is done after the noise addition, it will be more errors due to the explained Modulo errors and the BER performance becomes worse.

One solution for our channel (known as hpoF channel) is to make a post equalization (for example Zero Forcing equalization) to achieve that our channel is monic and has the maximum sample of the channel in the first position, because we have seen before the THP could work better under these conditions and this supposes an improvement on the BER performance.
Chapter 4

Backgrounds Reed Solomon

4.1 Introduction

In this chapter the Reed Solomon encoding is introduced in the theoretical way, explaining different reasons why it is useful by adding it with THP to further improve the system performance. The THP precoding removes inter-symbol interferences, and RS codes are often used to reduce bit errors to a tolerating level.

The RS code is a nonbinary cyclic code, subtype of FEC(also called channel coding), which is a type of digital signal processing that improves data reliability by introducing a known structure(redundant data) into a data sequence prior to transmission. Then, thanks to this structure, the receiver can detect and correct errors without the need to ask the sender for additional data. Some of the advantages of FEC are that back-channel is not required and the retransmission of data can often be avoided. One of the applications of FEC is that these codes are used in data storage systems.

RS describes a systematic way of building codes that could detect and correct multiple random symbol errors. One of the reasons why Reed Solomon works properly against the exposure to noise is when a decoder corrects a byte, it replaces the
incorrect byte with the correct one, whether the error bit was being caused by one or all eight bits corrupted being corrupted. This gives an RS code a tremendous burst-noise advantage over binary codes, even allowing for the interleaving of binary codes.

In order to fight against noise effectively, the noise duration has to be relatively small percentage of the codeword. To represent that ensure this happens most of the time, the receiver should be long average noise over a period of time, reducing the effect of a freak streak of bad luck. Hence, error-correcting codes become more efficient (error performance improved) as the code block size increases.

As the redundancy of an RS code increases (lower code rate), its implementation grows in complexity. Also, the bandwidth expansion must grow for any real-time communications application. However, the benefit of increased redundancy, just like the benefit of increased symbol size, is the improvement in bit-error performance.

4.2 Mathematical concept

Reed-Solomon codes are nonbinary cyclic codes with symbols made up of m-bit sequences, where \( m \) is any positive integer having a value greater than 2. RS \((n,k)\) codes on m-bit symbols exist for all \( n \) and \( k \) for which

\[
0 < k < n < 2^m + 2
\] (4.1)

where \( k \) is the number of data symbols being encoded, and \( n \) is the total number of code symbols in the encoded block. For the most conventional RS \((n,k)\) code,
(n, k) = (2^m - 1, 2^m - 1 - 2t) \quad (4.2)

where \( t \) is the symbol-error correcting capability of the code, and \( n - k = 2t \) is the number of parity symbols. An extended RS code can be made up with \( n = 2^m \) or \( n = 2^m + 1 \), but not any further.\[\text{8}\]

Reed-Solomon codes achieve the largest possible code minimum distance for any linear code with the same encoder input and output block lengths. For nonbinary codes, the distance between two codewords is defined (analogous to Hamming distance) as the number of symbols in which the sequences differ. For Reed-Solomon codes, the code minimum distance is given by \[\text{7}\]:

\[
d_{\text{min}} = n - k + 1
\quad (4.3)
\]

The code is capable of correcting any combination of \( t \) or fewer errors, where \( t \) can be expressed as:

\[
t = \left\lfloor \frac{d_{\text{min}} - 1}{2} \right\rfloor = \left\lfloor \frac{n - k}{2} \right\rfloor \quad (4.4)
\]

where \( \lfloor x \rfloor \) means the largest integer not to exceed \( x \). Equation 4.4 illustrates that for the case of RS codes, correcting \( t \) symbol errors requires no more than \( 2t \) parity symbols. The equation lends itself to the following intuitive reasoning. One can say that the decoder has \( n - k \) redundant symbols to "spend", which is twice the amount of correctable errors. For each error, one redundant symbol is used to
locate the error, and another redundant symbol is used to find its correct value.

In our particular case, we have used a RS code with the following values:

\[ m = 8, \text{ then } n = 255 \text{ and } k = 223 \]  

(4.5)

with these values, the code is capable of correcting any combination of \( t \) or fewer errors, in this case \( t \) is equal to 16. Correcting 16 symbol errors requires no more than 32 parity symbols.

4.2.1 Galois Finite Fields

It is necessary to know the Galois finite fields (GF) to understand the encoding and decoding of nonbinary codes. For any prime number, \( p \), there exists a finite field denoted \( GF(p) \) that contains \( p \) elements. It is possible to extend \( GF(p) \) to a field of \( p^m \) elements, called an extension field of \( GF(p) \), and denoted by \( GF(p^m) \), where \( m \) is a nonzero positive integer. Symbols from the extension field \( GF(2^m) \) are used in the construction of Reed-Solomon (RS) codes [8].
Chapter 5

Simulations results of RS

This chapter consists of developing a RS code for use in conjunction with the THP (see Appendix B).

There are some reasons for using this nonbinary cyclic code: as it has been explained in the last chapter, RS describes a systematic way of building codes that could detect and correct multiple random symbol errors. Another reason is that error performance is improved the code block size increases, making RS codes an attractive choice whenever long block lengths are desired.

The RS program wrote contains a modulation part after encoding the original bit sequence, in particular PAM modulation. In the encoding part, the bit sequence is transformed to a Galois array Message (belongs to Galois finite field) and then is converted to a Reed Solomon code word.

In the simulation, the RS(255,223) with m=8 is chosen (which is a commonly used code for DVB). Since up to 16 symbols/bytes can be corrected, the code is able to decrease BER from $10^{-4}$ to $10^{-9}$, where errorless transmission for digital communication is achieved \[\text{[14]}\].
5. Simulations results of RS

Next theoretical graph shows the RS performance (in Eb/No terms):

![Error performances for RS codes](image)

Figure 5.1: Error performances for RS codes [9].

If we observed the values and the trend of the curve, coincide with the explanation in the last paragraph, then these results should be expected in the simulation results from RS program.
Now, we can get the BER before RS code and BER after RS from RS program’s simulation with different modulations:

Figure 5.2: BER performance 2 PAM with RS

Figure 5.3: BER performance 4 PAM with RS
5. Simulations results of RS

The figures show the reduction of BER after the use of the RS code. In case of 2 PAM, the reduction occurs approximately with 7dB of SNR, only with the increase of 1dB till 8dB it is possible to achieve a reduction from $1.3 \cdot 10^{-2}$ to $7 \cdot 10^{-4}$. In 4 PAM in the range of 14-15dB, for example, the reduction is more prominent from $7 \cdot 10^{-3}$ till $1.8 \cdot 10^{-4}$ and finally in 8 PAM the reduction starts with 20dB of SNR, with SNR equal to 21dB the system can reach a value of $7 \cdot 10^{-5}$.

In the Figures 5.1, 5.2 and 5.3 is only shown a part of the BER performance, it is caused by the fact that the results shown above are dependent of the number of simulations, in our case this value is $10^{-6}$, if we would want to get more values to see that RS works as we expected, it is needed a bigger number of simulations, but this implies a lot of time getting the results, may be we should need a computer with higher processing. Instead of this, we can observe the trend of the BER performance with the use of RS prolonging it (in 8 PAM case, for example), we see that it satisfies the theoretically expected (when the BER before RS is $10^{-4}$, BER after RS reaches $10^{-9}$), while BER before RS needs 23.8dB(approximately) of SNR to reach the benchmark of $2 \cdot 10^{-4}$, in the case of RS performance with this value of SNR, reaches $10^{-9}$.
Now the imaginary prolongation in case of more simulations is done and represented as follows:

Figure 5.5: 8 PAM RS with prolongation.

Thus it demonstrates the great use of RS, which reduces the error performance in a substantial way for any value of modulation. An important observation is that when M increases, a higher SNR is needed to observe the RS effect in the BER performance.
Chapter 6

Simulations results of RS+TH

In this final chapter, the purpose is to introduce a program (see Appendix C) that includes both RS code and THP. So in this way improvement is consist of eliminating the ISI resulting from the use of THP and reducing the error performance thanks to the properties of RS code.

In TH program, the initial parameters and our channel are first defined. Next, inside the main loop the original bit sequence is encoded by RS and afterwards is modulated. The next step of this program consists of adding the THP part as it has been created before, then the noise is added and in this point starts again the same process described lines before but in reverse. In the final lines the BER after RS and before RS are obtained to be displayed after.

The next figure shows the BER performance achieved by the use of RS+THP together with 8 PAM modulation, it is possible to see that in case of 2Gb/s the BER after the use of RS+TH has a free error transmission from 37dB (with $10^{-6}$ simulations), where BER before RS decoder reaches $6 \cdot 10^{-5}$. With the other bit rates the BER achieved doesn’t supposed any improvement regarding the TH program without RS.
To summarize, if we observe the results with RS+TH, the BER performance in our system is better with the use of RS, from $10^{-2}$ this BER performance experiments a great reduction comparing the BER before RS, thanks to the capacity of detection and correction errors offered by RS.
Chapter 7

Conclusion

Indeed the use of TH precoding, is an improvement on the elimination of the ISI of our channel but there are some limitations. Our channel is not monic, therefore we have to normalize it to apply THP block in a correct way. Another point to consider, is the fact that the rest of the channel samples have values bigger than the initial sample, thus to reduce $\bar{x}$ we need a large number of reductions as we have seen, affecting the output of channel ($\bar{y}$). Then, in filter’s output, despite that $\bar{y}$ has no ISI, $\bar{y}$ is equivalent to the $\bar{a}$ (modulated signal) but affected by the signal that has reduced the $\bar{x}$ (in the first modulo reduction), which that corresponds to $\bar{d}$. If this number of reductions is high, the noise added is proportional depending on the output, this causes that after the second modulo reduction will be more errors (called modulo errors), then the BER performance becomes worsen. Therefore, the application of THP will be better if the channel is monic and the remaining samples have small values. If it is not the case, we need to consider which could be considered as a solution and obviously should be taken into account as future work to improve our system, the use of an equalizer would be one of the options to be considered to achieve the above exposed. Finally we can say that with the use of Reed Solomon is possible to achieve a great improvement in the BER performance and it is easily combined with THP.
Bibliography


[10] Jurgen Klauser and Olaf Ziemann Peter E. Zamzow Werner Daum


[12] Adnan Blind equalization for THP precoded systems

   Precoding IEEE paper

[14] Yixuan Wang Powerpoint presentation
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Block diagram of DFE</td>
<td>10</td>
</tr>
<tr>
<td>1.2</td>
<td>Output vs Input for the modulo adder</td>
<td>14</td>
</tr>
<tr>
<td>1.3</td>
<td>Tomlinson-Harashima precoder and linearized description</td>
<td>15</td>
</tr>
<tr>
<td>1.4</td>
<td>Extended signal set $V$ with $M = 4$</td>
<td>16</td>
</tr>
<tr>
<td>1.5</td>
<td>Complete scheme for a transmission with THP</td>
<td>17</td>
</tr>
<tr>
<td>1.6</td>
<td>Communication system using THP with ZF and DFE</td>
<td>20</td>
</tr>
<tr>
<td>2.1</td>
<td>System scheme</td>
<td>22</td>
</tr>
<tr>
<td>2.2</td>
<td>Channel impulse response</td>
<td>23</td>
</tr>
<tr>
<td>3.1</td>
<td>THP program structure</td>
<td>28</td>
</tr>
<tr>
<td>3.2</td>
<td>BER performance for different bit rates with THP</td>
<td>29</td>
</tr>
<tr>
<td>3.3</td>
<td>BER performance of 8 PAM with DFE for 2Gbit/s</td>
<td>29</td>
</tr>
<tr>
<td>3.4</td>
<td>BER performance of 8 PAM with DFE for 2.5Gbit/s</td>
<td>30</td>
</tr>
<tr>
<td>3.5</td>
<td>BER performance of 8 PAM with DFE for 3Gbit/s</td>
<td>30</td>
</tr>
<tr>
<td>3.6</td>
<td>BER performance for 8 PAM with TH for new monolithic channel</td>
<td>32</td>
</tr>
<tr>
<td>3.7</td>
<td>Diagram for explanation</td>
<td>34</td>
</tr>
<tr>
<td>5.1</td>
<td>Error performances for RS codes</td>
<td>40</td>
</tr>
<tr>
<td>5.2</td>
<td>BER performance 2 PAM with RS</td>
<td>41</td>
</tr>
<tr>
<td>5.3</td>
<td>BER performance 4 PAM with RS</td>
<td>41</td>
</tr>
<tr>
<td>5.4</td>
<td>BER performance 8 PAM with RS</td>
<td>42</td>
</tr>
<tr>
<td>5.5</td>
<td>8 PAM RS with prolongation</td>
<td>43</td>
</tr>
</tbody>
</table>
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>BER performance for 2Gbit/s with RS and THP.</td>
<td>45</td>
</tr>
</tbody>
</table>
Appendix A

Reed Solomon CODE

SNR = 15:30;
BERratiobefore=zeros(size(SNR));
BERratioafter=zeros(size(SNR));

for e = 1:length(SNR)
ja=0;jb=0;BERa=0;BERb=0;
    while jb <= 10^6

        m=8;k=223;M=8;n=255;
        bitseq1 = randint(m*k*log2(M),1); % generating the sequence
        ja=ja+m*k*log2(M);
        jb=jb+m*n*log2(M);
        x = reshape(bitseq1,m,k*log2(M));
        xx1 = bit2de(x');
        xx2 = reshape(xx1,k*log2(M))';
        msg = gf(xx2,m); % message in galois field

        % encode
        c = rsenc(msg,n,k)';
        cc = double(de2bi(c.x)');
        cc2 = reshape(cc,log2(M),n*m);

        % modulation
        h = modem.pammod('M',M,'SymbolOrder', 'gray','InputType', 'bit');
        y = modulate(h,cc2);
        y = real(y);

        % noise added
        yw = awgn(y,SNR(e), 'measured');

    end

    BERa(e) = ja/M*n/k/log2(M);
    BERb(e) = jb/M*n/k/log2(M);
end
h2=modem.pamdemod(h);
z=demodulate(h2,yw);
z3=reshape(z,m*n*log2(M),1);
z4=reshape(z,m,n*log2(M));
xx3=b12de(z4');
xx4=reshape(xx3,n,log2(M))';

%decode

g2=gf(xx4,m);
q=rsdec(g2,n,k);

H=double(q.x)';
H2=de2bi(H)';
H3=reshape(H2,m*k*log2(M),1);

[NUMb,RATIO]=biterr(z3,cc(:));
[NUMa,RATIO2]=biterr(H3,bitsq1);
BERb = NUMb-BERb;
BERa = NUMa-BERA;

end

BERratiobefore(e)=BERb/|b|;
BERratioafter(e)=BERa/|a|;

end

disp('BER ratio before RS decoder is:')
disp(BERratiobefore);
disp('BER ratio after RS decoder is:')
disp(BERratioafter);

semilogy(SNR,BERratiobefore,SNR,BERratioafter)
legend('BERratiobeforeRS','BERratioafterRS')
Appendix B

Tomlinson Harashima CODE

% POF parameters
AN = 0.5; % numerical aperture
Lpof = 10; % POF length in meter
tau = 4.97e-3; % unit us/m
alpha = 0; % fiber loss in dB/m;
A = 10^(-alpha*Lpof/10); % attenuation over the fiber
AN_On = false; % if numerical aperture effect

if AN_On
   Bo = 428.07/(AN^2) - 1127.2/AN + 1466.3; % unit MHz
   q = -0.0383/(AN^2) + 0.2156/AN + 0.5195;
   B_3dB = Bo*Lpof^(-q); % unit MHz
else
   B_3dB = 1009*Lpof^(-0.8747); % unit MHz
end
sigma = 0.1325/B_3dB;

% Simulation parameters
M = 8; % order of modulation
Rb = 2000; % bit rate in Mbit/s
Rs = Rb/log2(M); % symbol rate
fs = Rs; % sampling frequency
NFFT = 2048;
tt = (0:NFFT-1)/fs;
f = fs/NFFT*(1:NFFT-1);

% channel impulse response
hpoft = A/sqrt(2*pi)/sigma*exp(-(tt-tau*Lpof).^2/2/sigma^2);
\begin{verbatim}

n1 = find(abs(hpof1)>max(hpof1)*0.01);
chnl = hpof1(n1(1):n1(end));
chnl = chnl/sqrt(sum(chnl.^2));
figure, stem(tt(1:length(chnl)),chnl,'filled','MarkerSize',2);
legend('Sampled by Rs')
xlabel('Time [us]')
ylabel('hpof(t)')
grid on
[numfil,numcol]=size(n1);
chnl=chnl/chnl(1);

SNR=20:45;
BERratio13=zeros(size(SNR));
SERratio13=zeros(size(SNR));
for e = 1:length(SNR)
j=0;BERb13=0;SERb13=0;
while j<=10^3
m=3000;M=8;
bitseq1 = randint(m*log2(M),1);
j=j+m*log2(M);
bitseq2 = reshape(bitseq1,log2(M),m);

%MODULATION

g=modem.pammod('M',M,'SymbolOrder','gray','InputType','bit');
a=modulate(g,bitseq2);
a=real(a);

%TH PRECODE

x=zeros(1,m);
f=zeros(1,m);
y=zeros(1,m);
yn=zeros(1,m);
an=zeros(1,m);
q=zeros(1,m);
v=zeros(1,m);
b=zeros(1,m);
d=zeros(1,m);

for k=1:m
    n=2;
    if k>=2
        while n<k&n<=numcol
            f(k)=f(k)+chnl(n)*x(k-(n-1));
        end
    end
end

\end{verbatim}
\[ n = n + 1; \]
\[ \text{end} \]
\[ \text{end} \]
\[ q(k) = a(k) - f(k); \]

\text{% MOD-2M}

\[ \text{if} \ (q(k) \gt M) \]
\[ \text{while} \ (q(k) \gt M) \]
\[ q(k) = q(k) - (2 \times M); \]
\[ b(k) = b(k) + (2 \times M); \]
\[ d(k) = -b(k); \]
\[ \text{end} \]
\[ \text{elseif} \ (q(k) \leq (-M)) \]
\[ \text{while} \ (q(k) \leq (-M)) \]
\[ q(k) = q(k) + (2 \times M); \]
\[ d(k) = d(k) + (2 \times M); \]
\[ \text{end} \]
\[ \text{end} \]

\[ v(k) = a(k) + d(k); \]
\[ x(k) = v(k) - f(k); \]

\text{% FILTER}
\[ y = \text{conv} \ (x, \text{chnl}); \]

\text{% ADDING NOISE}
\[ y_n(k) = \text{awgn} \ (y(k), \text{SNR(e)}, ' \text{measured}'); \]
\[ a_n(k) = y_n(k); \]

\text{% MOD 2M-arithmetic}
\[ \text{if} \ (a_n(k) > M) \]
\[ \text{while} \ (a_n(k) > M) \]
\[ a_n(k) = a_n(k) - (2 \times M); \]
\[ \text{end} \]
\[ \text{elseif} \ (a_n(k) \leq (-M)) \]
\[ \text{while} \ (a_n(k) \leq (-M)) \]
\[ a_n(k) = a_n(k) + (2 \times M); \]
\[ \text{end} \]
\[ \text{end} \]
\[ \text{end} \]
% DEMOD

h2 = modem.pamdemod(g);
z = demodulate(h2, an);
zs = modulate(g, z);
zb = reshape(z, m*log2(M), 1);

[NUMs, RATIOS] = symerr(a, zs);
[NUM, RATIO] = biterr(zb, bitseq1);

SERb = NUMs / SERb;
BERb = NUM / BERb;

end

BER ratiobefore (e) = BERb / j;
SER ratiobefore (e) = SERb / j * log2(M);

end

figure; semilogy(SNR, BER ratiobefore, SNR, SER ratiobefore)
legend(['BER ratiobefore', 'SER ratiobefore'])
Appendix C

Tomlinson Harashima and Reed Solomon CODE

% POF parameters
AN = 0.5;  % numerical aperture
Lpof = 10;  % POF length in meter
tau = 4.97e-3;  % unit us/m
alpha = 0;  % fiber loss in dB/m;
A = 10^(-alpha*Lpof/10);  % attenuation over the fiber
AN_On = false;  % if numerical aperture effect

if AN_On
  Bo=428.07/(AN^2)-1127.2/AN+1466.3;  % unit MHz
  q=-0.0383/(AN^2)+0.2456/AN+0.5195;
  B_3dB=Bo*Lpof^(-q);  % unit MHz
else
  B_3dB=1009*Lpof^(-0.8747);  % unit MHz
end

sigma=0.1325/B_3dB;

% Simulation parameters
M = 8;  % order of modulation
Rb = 2000;  % bit rate in Mbit/s
Rs = Rb/log2(M);  % symbol
fs = Rs;  % sampling frequency
NFFT = 2048;
tt = (0:NFFT-1)/fs;
f = fs/NFFT*(1:NFFT-1);

% channel impulse response
\[ h_{\text{pof}} = A / \sqrt{2 \pi} \exp \left( - \frac{(t - \tau \ast L_{\text{pof}})^2}{2 \sigma^2} \right); \]

\[ n_1 = \text{find} \left( \text{abs} \left( h_{\text{pof}} \right) > \max(h_{\text{pof}}) \ast 0.01 \right); \]

\[ c_{\text{hl}} = h_{\text{pof}}(n_1(1):n_1(\text{end})); \]

\[ c_{\text{hl}} = c_{\text{hl}} / \sqrt{\text{sum}(c_{\text{hl}}.^2)}; \]

\[ \text{figure}, \text{stem}(tt(1:length(c_{\text{hl}})),c_{\text{hl}},'filled','MarkerSize',2); \]

\[ \text{legend}('\text{Sampled by Rs}') \]

\[ \text{xlabel('Time [us]')}; \]

\[ \text{ylabel('h_{pof}(t)')}; \]

\[ \text{grid on}; \]

\[ [\text{numfil},\text{numcol}] = \text{size}(n_1); \]

\[ c_{\text{hl}} = c_{\text{hl}} / c_{\text{hl}}(1); \text{because the channel is not monic, it is necessary for THP applying} \]

SNR = 20:45;

\[ \text{BERratiobefore} = \text{zeros} \left( \text{size}(\text{SNR}) \right); \]

\[ \text{SERratiobefore} = \text{zeros} \left( \text{size}(\text{SNR}) \right); \]

\[ \text{BERratioafter} = \text{zeros} \left( \text{size}(\text{SNR}) \right); \]

\[ \text{for } e = 1:length(\text{SNR}) \]

\[ j_b = 0; j_a = 0; \text{BER}_b = 0; \text{SER}_b = 0; \text{BER}_a = 0; \text{NUM} = 0; \text{NUM}2 = 0; \text{NUMs} = 0; \]

\[ \text{while } j_b < 10^3 \]

\[ m = 8; k = 223; M = 8; n = 255; \]

\[ \text{bitseq1} = \text{randint}(m \ast k \ast \log2(M),1); \text{generate seq} \]

\[ j_a = j_a + m \ast k \ast \log2(M); \]

\[ j_b = j_b + m \ast n \ast \log2(M); \]

\[ x = \text{reshape}(\text{bitseq1}, m \ast k \ast \log2(M)); \text{reorder the structure} \]

\[ xx1 = \text{bi2de}(x'); \text{binary to decimal} \]

\[ xx2 = \text{reshape}(xx1, k, \log2(M)); \]

\[ \text{msg} = \text{gf}(xx2, m); \text{Message is a Galois array.} \]

\[ \text{% encode} \]

\[ c = \text{rsenc}(\text{msg}, n, k)'; \]

\[ \text{cc} = \text{double} \left( \text{de2bi}(c.x) \right)'; \]

\[ \text{cc2} = \text{reshape}(\text{cc}, \log2(M), n \ast m); \]

\[ \text{% modulation} \]

\[ h = \text{modem.pammmod('M',M,'SymbolOrder', 'gray','InputType', 'bit');} \]

\[ a = \text{modulate}(h, \text{cc2}); \]

\[ a = \text{real}(a); \]

\[ \text{% TH PRECODE} \]

\[ x = \text{zeros}(1,m \ast n); \]

\[ f = \text{zeros}(1,m \ast n); \]

\[ y = \text{zeros}(1,m \ast n); \]

\[ y_n = \text{zeros}(1,m \ast n); \]

\[ a_n = \text{zeros}(1,m \ast n); \]
v = zeros(1, m*n);
d = zeros(1, m*n);
b = zeros(1, m*n);
q = zeros(1, m*n);

for s = 1:(m*n)
    w = 2;
    if s >= 2
        while w <= s && w <= numcol
            f(s) = f(s) + chnl(w) * x(s-(w-1));
            w = w + 1;
        end
    end
    q(s) = a(s) - f(s);

% MOD-2M
    if (q(s) >= M)
        while (q(s) >= M)
            q(s) = q(s) - (2*M);
            b(s) = b(s) + (2*M);
            d(s) = -b(s);
        end
    elseif (q(s) < (-M))
        while (q(s) < (-M))
            q(s) = q(s) + (2*M);
            d(s) = d(s) + (2*M);
        end
    end

v(s) = a(s) + d(s);
x(s) = v(s) - f(s);

% FILTER
    y = conv(x, chnl);

% ADDING NOISE
    yn(s) = awgn(y(s), SNR(e), 'measured');
    yn(s) = u(s) + n(s);

% MOD 2M-arithmetic
    if (an(s) >= M)
        while (an(s) >= M)
            an(s) = an(s) - (2*M);
        end
else if (an(s)<(-M))
    while (an(s)<(-M))
        an(s)=an(s)+(2*M);
    end
end

end

% demodulation
h2=modem.pandemod(h);
z=demodulate(h2,an);
zs=modulate(h,z);
z3=reshape(z,m*n*log2(M),1);
z4=reshape(z,m,n*log2(M));
xx3 =bi2de(z4');
xx4=reshape(xx3,n,log2(M)');
g2=gf(xx4,m); % Message is a Galois array.

% decode
q=rsdec(g2,n,k);

H=double(q.x); % change type of data
H2 = de2bi(H)';
H3=reshape(H2,m*k*log2(M),1);

[NUM,RATIO]=biterr(z3,cc( : ));
[NUM2,RATIO2]=biterr(H3,bitseq1);
[NUMs,RATIOs]=symerr(a,zs);

BERb=NUM-BERb;
BERa=NUM2-BERA;
SERb=NUMs-SERb;

end

BERratiobefore(e)=BERb/\|b\|;
BERratioafter(e)=BERa/\|a\|;
SERratiobefore(e)=SERb/\|b\|*log2(M);
end

disp('BER ratio before RS decoder is:')
disp(BERratiobefore);
disp('BER ratio after RS decoder is:')
disp(BERratioafter);

semilogy(SNR,BERratiobefore,SNR,BERratioafter,SNR,SERratiobefore)
legend ('BER ratio before RS', 'BER ratio after RS', 'SER ratio before RS');