

Bachelor's Thesis



**Czech
Technical
University
in Prague**

F3

Faculty of Electrical Engineering
Department of Radio Engineering

Wireless Physical Layer Network Coding

Jozef Lukáč
Open Electronic Systems

May 2017
Supervisor: prof. Ing. Jan Sýkora, CSc.

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Radioelectronics

BACHELOR PROJECT ASSIGNMENT

Student: **Jozef Lukáč**

Study programme: Open Electronic Systems

Title of Bachelor Project: **WPNC coding in simple radio networks**

Guidelines:

Student will get acquainted with fundamentals of Wireless Physical Layer Network Coding (WPNC) in simple scenarios including two-way relay channel and butterfly network. Student should focus on the simple source components with layered (or uncoded) Network Coded Modulation (NCM) and simple relay processing strategies (e.g. hierarchical decode and forward) and on simple destination decoding strategies with perfect hierarchical side information (or some simple scenarios with imperfect one). Student should investigate the impact of the transmitted NCM and H-MAC channel parameterisation on the H-constellation and the overall end-to-end performance (solvability and error rate properties). The work target is to develop an implementation and simulation platform in Matlab that would demonstrate functionality and the performance of the selected schemes.

Bibliography/Sources:

[1] J. Sykora, A. Burr: Wireless Physical Layer Network Coding, tutorial 2016

Bachelor Project Supervisor: prof. Jan Sýkora Ing., CSc.

Valid until the end of the summer semester of academic year 2017/2018

L.S.

doc. Mgr. Petr Páta, Ph.D.
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, January 27, 2017

Acknowledgement / Declaration

I would like to thank to prof. Ing. Jan Sýkora, CSc. for his guidance and for providing study materials to the WPNC issue.

I declare, that I have worked up the submitted work independently and that I have mentioned all the sources used in accordance with methodical instruction about adhering ethical principles when preparing university final works.

In Prague, 26. 5. 2017.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 26. 5. 2017

.....

Abstrakt / Abstract

Tenhle dokument stručně vysvětluje hlavní myšlenky WPNC (Kódování na fyzické vrstvě sítí), koncept obousměrného kanálu s jedním přeposílacím uzlem (2WRC) a „motýlí topologie“ sítě. Cílem této práce je přezkoumat závislost relativního počtu chyb v 2WRC na SNR (poměru výkonu signálu, k výkonu šumu).

Klíčová slova: Bezdrátové kódování na fyzické vrstvě sítí; obousměrní kanál s jedním přeposílacím uzlem, „Motýlí topologie“ sítě.

Překlad titulu: Bezdrátové kódování na fyzické vrstvě sítí

This document briefly explains main ideas of WPNC, concept of a Two Way Relay Channel (2WRC) and butterfly network. The aim of the work is to examine error rate dependencies in 2WRC on signal to noise ratio.

Keywords: Wireless Physical Layer Network Coding; Two Way Relay Channel; Butterfly network.

/ Contents

1 Introduction	1
1.1 Definitions of Basic Structures ..	1
1.1.1 Field	1
1.1.2 Vector Space	2
1.2 Modulation fundamentals	4
1.2.1 Modulation Pulse	4
1.2.2 PSK	5
1.3 Communication Channel Model.....	5
1.4 Demodulation and Detection Fundamentals	6
1.5 Wireless Physical Layer Network Coding (WPNC)	7
1.6 Two Way Relay Channel with BPSK	9
1.6.1 Demodulation and Decoding in 2WRC with BPSK.....	10
1.7 Two Way Relay Channel with QPSK	10
2 Error performance in 2WRC and Matlab implementation	13
2.1 Discrete data modulation.....	13
2.2 Data flows in 2WRC and error rate	14
2.3 A note on a detector.....	17
2.4 Conclusion and results.....	20
References	23
A Attached files	25

Tables / Figures

<p>1.1. XOR function as an example of relay function9</p> <p>1.2. BPSK relay mapping in 2WRC 10</p> <p>1.3. QPSK relay mapping in 2WRC 11</p> <p>1.4. Hierarchical symbols labeling for QPSK. 11</p>	<p>1.1. Demodulation procedure.....7</p> <p>1.2. Two way relay channel network ..8</p> <p>1.3. Butterfly network8</p> <p>1.4. H-points for BPSK, $h = 1$9</p> <p>1.5. H-points for BPSK, $h = 0.9e^{j25\pi/180}$9</p> <p>1.6. H-points for QPSK, $h = 1$ 10</p> <p>1.7. H-points for QPSK, $h = 0.9e^{j5\pi/180}$ 11</p> <p>2.1. Non-causal and causal pulse... 14</p> <p>2.2. An example of a hierarchical data impulse train and received filtered signal 15</p> <p>2.3. A BPSK decision region for $h = 0.9 \exp(j25\pi/180)$..... 18</p> <p>2.4. A BPSK decision region for $h = 0.9 \exp(j100\pi/180)$..... 18</p> <p>2.5. A QPSK decision region for $h = 0.7 \exp(j5\pi/180)$ 19</p> <p>2.6. A QPSK decision region for $h = 0.7 \exp(j15\pi/180)$..... 19</p> <p>2.7. A QPSK decision region for $h = 0.7 \exp(j50\pi/180)$..... 19</p> <p>2.8. A QPSK decision region for $h = 0.7 \exp(j150\pi/180)$..... 20</p> <p>2.9. The error rate in 2WRC using BPSK; for unknown h 20</p> <p>2.10. The error rate in 2WRC using BPSK; for known h 21</p> <p>2.11. The error rate in 2WRC using QPSK; for unknown h 21</p> <p>2.12. The error rate in 2WRC using QPSK; for known h 21</p>
---	--

Chapter 1

Introduction

In the chapter first some elementary concepts are introduced. Then some radio communication parts are mentioned and finally the 2WRC is introduced and BPSK and QPSK modulation are studied.

1.1 Definitions of Basic Structures

The section shortly refreshes general knowledge about structures (sometimes implicitly) used through the work.

1.1.1 Field

We can represent a signal as a very strange object, depending on a situation of use. We do it mainly because it simplifies the original treatment or we can show some dependencies easily. Anyway we need to work with such a strange objects – it means we need to define operations with them.

Definition 1.1. (Field) [1, pg. 296] *A Field is an algebraic structure $\mathbb{F} = (\mathcal{A}, \square, \circ)$ defining two operation \square, \circ and the set of elements \mathcal{A} with the properties given below:*

(closure)	$\forall a, b \in \mathcal{A} :$	$(a \square b) \in \mathcal{A},$	(1)
(associativity)	$\forall a, b, c \in \mathcal{A} :$	$(a \square b) \square c = a \square (b \square c),$	(2)
(neutral element)	$\exists e \in \mathcal{A} : \forall a \in \mathcal{A}$	$a \square e = e \square a,$	(3)
(inverse)	$\forall a \in \mathcal{A} : \exists \bar{a} \in \mathcal{A} :$	$a \square \bar{a} = \bar{a} \square a = e,$	(4)
(commutativity)	$\forall a, b \in \mathcal{A} :$	$a \square b = b \square a.$	(5)

The neutral element w.r.t. \square is usually called “zero” and also denoted such $e = 0$. The field w.r.t \circ has an inversion for all non-zero elements.

(closure)	$\forall a, b \in \mathcal{A} :$	$(a \circ b) \in \mathcal{A},$	(6)
(associativity)	$\forall a, b, c \in \mathcal{A} :$	$(a \circ b) \circ c = a \circ (b \circ c),$	(7)
(neutral element)	$\exists i \in \mathcal{A} : \forall a \in \mathcal{A}$	$a \circ i = i \circ a,$	(8)
(inverse)	$\forall a \in \mathcal{A} \setminus \{0\} : \exists \bar{a} \in \mathcal{A} :$	$a \circ \bar{a} = \bar{a} \circ a = i,$	(9)
(commutativity)	$\forall a, b \in \mathcal{A} :$	$a \circ b = b \circ a.$	(10)

The neutral element w.r.t. \circ is usually called “unity” and also denoted such $i = 1$. The \circ and \square operations are bind together by a distributive law

$$\text{(distributive law)} \quad \forall a, b, c \in \mathcal{A} : a \circ (b \square c) = (a \circ b) \square (a \circ c). \quad (11)$$

If the cardinality of the elements set is finite, we call it finite field of Galois field (GF). The finite field exists only if its size is p^m where p is a prime number and $n \in \mathbb{N}$. If $m = 1$, we can define \square and \circ as a modulo p addition and multiplication.

1.1.2 Vector Space

A common way to represent a signal is as a function $s(t)$ of a time or in a discrete time as $s[n]$ sequence both with values generally in \mathbb{C} . We often use operation with signals such as sum and scaling of signals. For that it is useful a vector space structure V .

Definition 1.2. (Vector space) *A set $V = \{v_i\}$ is called the vector space if the following holds: $\forall \vec{x}, \vec{y}, \vec{z} \in V$ and any scalars $r, s \in \mathbb{F}$ (we will use the field \mathbb{C}):*

$$\text{(Closure under vector addition)} \quad \vec{x} \square \vec{y} \in V, \quad (12)$$

$$\text{(Closure under scalar multiplication)} \quad r \circ \vec{x} \in V, \quad (13)$$

$$\text{(Comutativity)} \quad \vec{x} \square \vec{y} = \vec{y} \square \vec{x}, \quad (14)$$

$$\text{(Associativity of vector addition)} \quad (\vec{x} \square \vec{y}) \square \vec{z} = \vec{x} \square (\vec{y} \square \vec{z}), \quad (15)$$

$$\text{(Existence of a null vector)} \quad \vec{0} \square \vec{x} = \vec{x} \square \vec{0} = \vec{x}, \quad (16)$$

$$\text{(Existence of additive inverse)} \quad \exists(-\vec{x}) : \vec{x} \square (-\vec{x}) = \vec{0}, \quad (17)$$

$$\text{(Associativity of scalar multiplication)} \quad r \circ (s \circ \vec{x}) = (r \cdot s) \circ \vec{x}, \quad (18)$$

$$\text{(Distributivity of scallar sums)} \quad (r + s) \circ \vec{x} = r \circ \vec{x} \square s \circ \vec{x}, \quad (19)$$

$$\text{(Distributivity of vector sums)} \quad r \circ (\vec{x} \square \vec{y}) = r \circ \vec{x} \square r \circ \vec{y}, \quad (20)$$

$$\text{(Scalar multiplication identity)} \quad 1 \circ \vec{x} = \vec{x}, \quad (21)$$

The elements of vector space are called vectors. \square – element-wise addition of vectors, \circ – scalar multiplication (both operations defined on the field).[[2]]

On the vector space we define so called *inner product*: $\langle \cdot, \cdot \rangle : V \times V \rightarrow \mathbb{C}$ – an arbitrary function satisfying $\forall a, b, c \in V$:

$$\langle a + b, c \rangle = \langle a, c \rangle + \langle b, c \rangle \quad (22)$$

$$\langle \alpha a, b \rangle = \alpha \langle a, b \rangle \quad (23)$$

$$\langle a, b \rangle = \langle b, a \rangle^* \quad (24)$$

$$\langle a, a \rangle \geq 0, \text{ it is 0 only if } v = 0 \quad (25)$$

A norm on a vector space (a “size” of a vector) is frequently defined by the inner product such as: $\|s\| = \sqrt{\langle s, s \rangle}$.

We will work with vector space of functions \mathcal{L}^2 , a vector space of sequences ℓ^2 and a vector space of n-tuples \mathbb{C}^n . We will consider all these spaces to be defined over the field of \mathbb{C} . (i.e. the scalars used for multiplication of vectors are from \mathbb{C} , the summation of vectors are done element-wise). The inner product is defined as:

$$(\mathcal{L}^2) : \langle f(t), g(t) \rangle = \int_{-\infty}^{\infty} f(t)g^*(t)dt,$$

$$(\ell^2) : \langle f[n], g[n] \rangle = \sum_{k=-\infty}^{\infty} f[k]g^*[k],$$

$$(\mathbb{C}^n) : \langle \vec{f}, \vec{g} \rangle = \sum_{k=1}^n f_k g_k^*.$$

The norms on these vector spaces are:

$$\|f\| = \|f\|_2 = \sqrt{\langle f, f \rangle}.$$

Every vector in a vector space must satisfy $\|f\| < \infty$, otherwise the element doesn't belong to the vector space.

A set of vectors \mathcal{B} , that enables every vector to be written as a weighted sum of vectors from $\mathcal{B} = \{b_i | i = 1, 2, \dots\}$ i.e.

$$\forall v \in V : v = \sum_{k=1}^{|\mathcal{B}|} \alpha_k b_k, \quad \alpha_k \in \mathbb{C}, b_k \in \mathcal{B}.$$

and the cardinality of (number of elements in) \mathcal{B} ($|\mathcal{B}|$) is the lowest possible is called a *basis* of the vector space. There is a lot of bases for one vector space. One of them is *orthogonal basis*. The orthogonal basis consists of the vectors that satisfy:

$$\forall v, w \in \mathcal{B}_o : \langle v, w \rangle = 0.$$

We say the vectors in \mathcal{B}_o are orthogonal. An orthogonal basis with vectors fulfilling $\|v_i\| = 1, \forall v_i \in \mathcal{B}_o$ is called *orthonormal basis*. Vectors are called orthonormal, explicitly written as:

$$\langle b_i, b_j \rangle = \delta_{i,j}, \text{ i.e. } 0 \text{ for } i \neq j \text{ and } 1 \text{ for } i = j. \quad (26)$$

Sometimes we need to examine a subset of a vector space. It's required the set to be also a vector space. Such a vector space is called *subspace* of the original vector space.

In the communication it is used not many different signals, therefore we can describe these signals in a communication channel generally as signals from a vector subspace of \mathcal{L}_2 (the norm of these signals is finite, because the signals are finite – non-zero only in an finite length interval and doesn't took on infinite values). The cardinality of the basis is called *dimension*. Now in the subspace with dimension n , we will show an equivalence of the inner product between \mathcal{L}_2 and \mathbb{C}^n . Let's choose an orthonormal basis of the n -dimensional vector space: $\mathcal{B} = \{\zeta_i(t) | i = 1, 2, \dots, n\}$. Select 2 arbitrary vectors:

$$\begin{aligned} x(t) &= \sum_{k=1}^n x_k \zeta_k(t), \\ y(t) &= \sum_{k=1}^n y_k \zeta_k(t). \end{aligned}$$

Coefficients x_i are determined as:

$$\langle x(t), \zeta_i(t) \rangle = \left\langle \sum_{k=1}^n x_k \zeta_k(t), \zeta_i(t) \right\rangle \stackrel{(22)}{=} \sum_{k=1}^n x_k \langle \zeta_k(t), \zeta_i(t) \rangle \stackrel{(26)}{=} \sum_{k=1}^n x_k \delta_{ki} = x_i. \quad (27)$$

We have used the linearity property of the inner product and the orthonormal property of the $\zeta_i(t)$ vectors. Similarly for y_i . Then using complex conjugate linearity in the second component of the inner product and the orthonormal basis $\zeta_i(t)$:

$$\begin{aligned} \langle x(t), y(t) \rangle &= \left\langle \sum_{k=1}^n x_k \zeta_k(t), \sum_{l=1}^n y_l \zeta_l(t) \right\rangle = \sum_{k=1}^n \sum_{l=1}^n x_k y_l^* \langle \zeta_k(t), \zeta_l(t) \rangle, \\ &= \sum_{k=1}^n \sum_{l=1}^n x_k y_l \delta_{kl} = \sum_{k=1}^n x_k y_k^* = \langle \vec{x}, \vec{y} \rangle. \end{aligned} \quad (28)$$

Where $\vec{x} = [x_1, x_2, \dots, x_n]$, similarly for \vec{y} , $\vec{x}, \vec{y} \in \mathbb{C}^n$ are n -tuples of complex numbers – points in \mathbb{C}^n space. That enable us to represent signal by it's coefficients w.r.t. given orthonormal basis – a point in the \mathbb{C}^n space called *constellation space* in the communication branch. In the signal processing the inner product is called *correlation* between signals, the autocorrelation of a signal is named an *energy* of the signal.

1.2 Modulation fundamentals

In communication branch we are dealing with the issue how to encode data to a signal to enable receiver reliably decode them. An elementary part of a signal is a pulse, we will use labelling $g(t)$ for a pulse (it is a function of time, discrete $g[n]$ or continuous $g(t)$). Data will be a discrete vector $\vec{d} \in \mathbb{C}^n$ with elements $d_i \in \mathbb{C}$ called *symbols*. For a better properties we can broadcast some function of symbols: $q_i = q_i(d_i, \sigma_i)$ called *channel symbols*, that depend on actual data and sometimes on a function of past data σ_i . Also it can be an identity. The modulation is a modulation/modification of the pulse by channel symbol. The pulse then carry an information about our channel symbol, explicitly written as: $g_i(t) = g_i(q_i(d_i, \sigma_i), t)$. We often use only one modulation pulse. The pulses $g_i(t)$ are formed from $g(t)$ by the time shift: $g_i(q_i, t) = g(q_i, t - iT_s)$, where T_s states for *symbol period*, the time allocated to one symbol to broadcast. There is two ways how to construct signal from a data: by the sum of modulation pulses or by the product of them. We will use the sum. So the transmitted signal is:

$$s(t) = \sum_{i=1}^n g(q_i(d_i, \sigma_i, t - iT_s)). \quad (29)$$

In this work we will examine two linear modulations. A linear modulation is characterized by linearity of g w.r.t. q_i , i.e.: $g(q_i, t) = q_i g(t)$.

1.2.1 Modulation Pulse

A very important attribute of pulses is their correlation, also the correlation of the shifted pulses. We can construct a correlation function of them:

$$R_{g_1, g_2}(\tau) = \int_{-\infty}^{\infty} g_1(t + \tau) g_2^*(t) dt. \quad (30)$$

The correlation function says how much is one signal influenced by the other for a given time shift τ . We are mainly interested in time shifts of an integer multiple of T_s - symbol period. The pulses which are unaffected each other w.r.t. an integer multiple of symbol period we call Nyquist pulses.

Definition 1.3. (Nyquist pulses) *For the two pulses (not necessary different) we define a correlation function:*

$$R_{g_1, g_2}[n] = \langle g_1(t - nT_s), g_2(t) \rangle.$$

If the correlation function is equal 0 for all $n \in \mathbb{Z} \setminus \{0\}$ the pulses are called Nyquist to each other. In case of $g_1 = g_2$ we speak about Nyquist pulse.

The most common modulation pulses are *rectangular* pulse (REC) and *root raised cosine spectrum* pulse (RRC). They are defined as:

$$\text{(REC):} \quad g_{\text{rec } n}(t) = \frac{1}{\sqrt{\tau}} (1(t) - 1(t - \tau)), \tau = nT_s. \quad (31)$$

$$\text{(RRC):} \quad G_{\text{rrc}}(f) = \begin{cases} \sqrt{T_s} & , \text{ for } |f| < \frac{1-\alpha}{2T_s}, \\ \sqrt{T_s} \cos\left(\frac{\pi T_s}{2\alpha} \left(|f| - \frac{1-\alpha}{2T_s}\right)\right) & , \text{ for } \frac{1-\alpha}{2T_s} \leq |f| < \frac{1+\alpha}{2T_s}, \\ 0 & , \text{ for } \frac{1+\alpha}{2T_s} \leq |f|. \end{cases} \quad (32)$$

$1(t)$ is a unit step function: zero for $t < 0$ and one for $t \geq 0$) The RRC pulse is defined in frequency domain. In time domain it is $g_{\text{rec}}(t) = \mathcal{F}^{-1}\{G_{\text{rrc}}(f)\}(t)$, (\mathcal{F}^{-1} means inverse Fourier Transform):

$$g_{\text{rrc}}(t) = \frac{1}{\sqrt{T_s}(1 - 16\alpha^2 t^2/T_s^2)} \left[\frac{\sin((1 - \alpha)\pi t/T_s)}{\pi t/T_s} + \frac{4\alpha \cos((1 + \alpha)\pi t/T_s)}{\pi} \right]. \quad (33)$$

The RRC pulse is defined so to be Nyquist pulse. The REC pulse is important because it's the simplest one. Next we will consider pulses with unit energy: $\langle g, g \rangle = 1$.

1.2.2 PSK

Full name – Phase Shift Keying is a linear modulation type. The data are encoded to the signal by a shift of the constellation data point angle. Data are of the form:

$$q_n \in \{e^{j2\pi i/M_q}\}_{i=0}^{M_q-1}, \quad (34)$$

where M_q is a number of the channel symbols. In our case we will assume the data points to be identical to the channel symbols, i.e. $d_n = q_n$ and the memoryless modulation – $q_n(d_n, \sigma_n) = q_n(d_n) = d_n$. We define symbol energy as an energy of the pulse modified by the channel symbol (eventually also by the memory):

$$\mathcal{E}(d_n, \sigma_n) = \frac{1}{2} \langle g(d_n, \sigma_n, t), g(d_n, \sigma_n, t) \rangle, \quad (35)$$

The $1/2$ is there because we work with so called *convex envelopes* of all signals/pulses, but we would like to relate the energy to real signals energy. It can be shown that if we multiply the energy acquired from signals/pulses in convex envelopes by $1/2$ we get the energy of real signals. I would like to emphasize that the $\langle g, g \rangle$ term called an energy here doesn't have to relate to real/physical energy of the signal. (often it is needed to scale that by a constant, but generally in signal theory there is no relation). For our channel symbols the symbol energy is:

$$\mathcal{E}(q_n) = \frac{1}{2} \langle q_n g(t), q_n g(t) \rangle = \frac{1}{2} q_n q_n^* \langle g(t), g(t) \rangle = \frac{1}{2} |q_n|^2 = \frac{1}{2}. \quad (36)$$

We will use two types of the PSK: BPSK and QPSK.

- BPSK: Binary PSK – binary data $M_q = 2$. $q_n \in \{\pm 1\}$.
- QPSK: Quaternary PSK – $M_q = 4$. $q_n \in \{e^{j\pi/4} \exp(j\pi i/2)\}_{i=0}^{M_q-1}$.

1.3 Communication Channel Model

By the channel concept we are describing the propagation and changes in a broadcast signal by the environment. The description of a channel is probabilistic, it is described by a conditional probability $p(y|x)$ – probability of an output y provided that we have broadcast signal s . The probability $p(y|x)$ depends also on channel parameters θ : $p(y|x) = p(y|x, \theta)$.

In the work we will use only classical *Additive White Gaussian Noise Channel* (AWGN Channel). Input-output relation is: $y(t) = x(t) + w(t)$. (w is the noise – parameter of the channel, $w = \theta$). The noise w – random process, is:

- Gaussian (complex) – Gaussian bell shape of the probability density function (PDF)
- stationary – second moments depends only on a time shift
- white – power spectral density (PSD) is constant over *all* frequencies
- rotationally symmetric – preserves stochastic properties under arbitrary rotation: $w' = e^{j\phi} w$, $p_w(\xi) = p_{w'}(\xi)$.

The noise a priori probability density function (PDF) $p_{\mathbf{w}}(\mathbf{w})$ is:

$$p_{\mathbf{w}}(\mathbf{w}) = c e^{-\|\mathbf{w}\|^2/(2N_0)}. \quad (37)$$

$c > 0$ is a normative constant, N_0 is a spectral power density of AWGN.

1.4 Demodulation and Detection Fundamentals

In demodulation we have available the broadcast signal changed by the channel: $u = f(x, \theta)$ – useful signal. The demodulator makes a mapping $x \rightarrow \rho(x, \hat{d})$. The decoder then makes decision based on the metric ρ . The AWGN channel is assumed: $x(t) = u(t) + w(t)$. We are interested in the $p(x|u)$ probability. The following holds:

$$p(x|u) = \sum_w p(x, u|w)p(w) = \sum_w \delta((x - u) - w)p(w) = p_w(x - u). \quad (38)$$

Using that the only random part is w , so $p(x, u|w)$, i.e. joint probability of x and u , provided w is non-random. Mathematically written as a Dirac delta function centered in $x - u = w$.

We assume uniformly distributed data, so *maximum likelihood* (ML) detector will be used. It is a procedure of maximizing likelihood function of the channel through possible useful signals. Sometimes, it is handy to maximize some function of the likelihood function, that preserves the extrema points. Such a function is e.g. logarithm \log_a , $a > 1$. We are used to call the new function *log-likelihood function*. In our case the likelihood function of the channel is:

$$p(x|u) = p_w(x - u) = c' \exp(-\|x - u\|^2/(2N_0)). \quad (39)$$

The log-likelihood function is:

$$\Lambda(u) = \ln(p(x|u)) = c(-\|x - u\|^2/2), \quad (40)$$

Where c' is $c/(N_0) > 0$. The monotone of the maximized function and the extrema points will not be changed if the function will be multiplied by a positive constant. We can unwind $\Lambda(u) := \Lambda(u)/c'$ as:

$$\begin{aligned} \Lambda(u) &= -\frac{1}{2} \langle x - u, x - u \rangle = -\frac{1}{2} (\|x\|^2 - \langle x, u \rangle - \langle u, x \rangle + \|u\|^2), \\ &\stackrel{(24)}{=} -\frac{1}{2} (\|x\|^2 - \langle x, u \rangle - \langle x, u \rangle^* + \|u\|^2) = -(\|x\|^2/2 - \operatorname{Re}\{\langle x, u \rangle\} + \|u\|^2/2). \end{aligned} \quad (41)$$

The useful signal is $u(t) = s(t)$. Let's assume linear modulation with a Nyquist pulse. The reminder ($E_g := \frac{1}{2} \langle g, g \rangle$, energy of the pulse):

$$\begin{aligned} u(t) = s(t) &= \sum_k q_k g(t - kT_s), \\ R_g[n] &= \langle g(t - nT_s), g(t) \rangle = 2E_g \delta_{n0}, \\ \|u\|^2 &= \left\langle \sum_k q_k g(t - kT_s), \sum_l q_l g(t - lT_s) \right\rangle \\ &= \sum_k \sum_l q_k q_l^* 2E_g \delta_{kl} = 2E_g \sum_k |q_k|^2, \\ \langle x, u \rangle &= \sum_k q_k^* \langle x(t), g(t - kT_s) \rangle, \\ z_k &:= q_k^* \langle x(t), g(t - kT_s) \rangle. \end{aligned} \quad (42)$$

In demodulation we suppose x to be received yet. The extrema points of a function $\Lambda(u)$ are independent w.r.t. constant shift in values. Let's assign $\Lambda(u) = 2\operatorname{Re}\{\langle x, u \rangle\} - \|u\|^2$.

Because of $u(t) = \sum_k q_k g(t - kT_s)$ we can consider the Λ as a function of $\vec{q} = (q_k)_{k=0}^{N-1}$. That enables us to write $\Lambda(\vec{q})$ as a sum of symbol log-likelihood functions $\Lambda_k(q_k)$:

$$\begin{aligned}\Lambda(\vec{q}) &= \sum_k \Lambda_k(q_k) = \sum_k [\text{Re}\{q_k^* \langle x, g(t - kT_s) \rangle\} - E_g |q_k|^2], \\ \Lambda_k(q_k) &= \text{Re}\{z_k\} - E_g |q_k|^2.\end{aligned}\quad (43)$$

In order to maximize $\Lambda(\vec{q})$, we have to maximize $\Lambda_k(q_k)$, i.e. symbol-wise maximization. The formula for z_k can be arranged as:

$$\begin{aligned}z_k &= q_k^* \int_{-\infty}^{\infty} x(t) g^*(-(kT_s - t)) dt = q_k^* x(t) * g^*(-t)|_{kT_s} \quad (\text{continuous time}), \\ &= q_k^* \sum_{l=-\infty}^{\infty} x[l] g^*[-(kN_s - l)] = q_k^* x[k] * g^*[-k]|_{kN_s}. \quad (\text{discrete time}).\end{aligned}\quad (44)$$

(N_s - number of samples per symbol, $*$ - convolution)

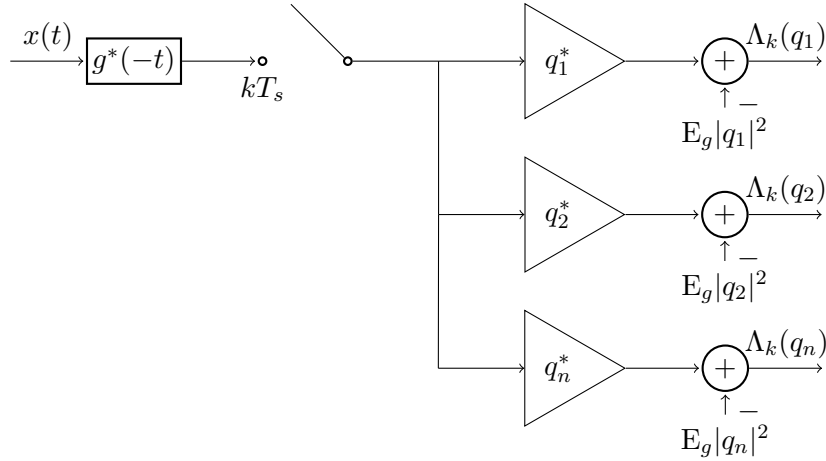


Figure 1.1. Illustrated processing of the signal x .

Then for every discrete-time step k we will choose the maximum value of $\{\Lambda_k(q_i)\}_i$. The estimated symbol is :

$$\hat{q}_k = \arg \max_{q^{(i)}} \Lambda_k(q^{(i)}). \quad (45)$$

($\{q^{(i)}\}_{i=0}^{N-1}$ is a set of all channel symbols)

1.5 Wireless Physical Layer Network Coding (WPNC)

The name states for a method used in a wireless channel. In the channel there is a certain amount of source nodes (which would like to send their messages) and an amount of destination nodes (which receives these messages). Also there are some mediators called *relays*, that forward messages. In the work we will examine one basic node structure called *2WRC* (Two Way Relay Channel) (1.2), that is a special case of *Butterfly Network* (1.3) namely the Butterfly Network with so called *perfect hierarchical information* (HI). The arrows with the same style symbolize data flows that are done together (with the other flows marked the same style arrows).



Figure 1.2. Two way relay channel network.

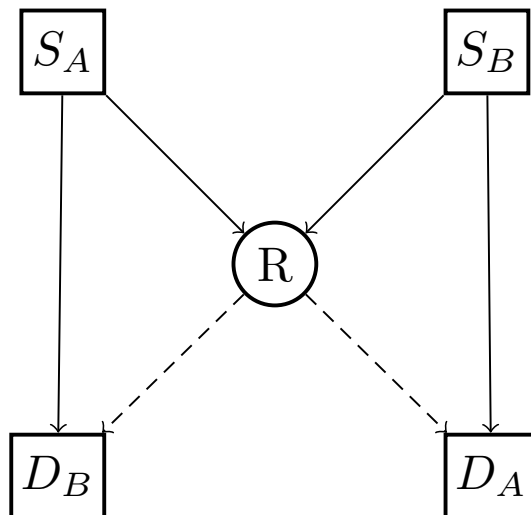


Figure 1.3. Butterfly network (S_A – source A, D_A – destination A).

Let's assume binary data and BPSK modulation. We distinguish 3 approaches of communication strategies:

- *Conventional method.* It organizes relay so to enable one way transfer of the data from one source node to another destination node, then after some time slots it switches to enable opposite data flow. In the 2WRC there are four stages/data flows: $A \rightarrow R$, $R \rightarrow B$, $B \rightarrow R$ and $R \rightarrow A$.
- *Network-Layer Network Coding.* The method organizes relay so to receive data from A, then receive data from B, computes a function of data – in our case XOR (exclusive OR (\oplus) on binary data) and broadcast the value to the terminals. For each terminal it is enough information to decode data from the opposite terminal – $a \oplus (a \oplus b) = b$ and similarly for the b . (a , resp. b are binary data attributable to source A, resp. B).
- *Wireless Physical Layer Network Coding.* The approach improves the latter one by using the wireless property of the channel.

“WPNC applies to wireless *networks* not to individual point-to-point links.” [1, pg. 16] It uses network structure – it is aware of that. All processing is being realized on a physical layer. There is two basic phases (2 phase protocol): 1) Multiple Access Channel (MAC) – sources transmit their messages to relay, 2) Broadcast Channel (BC) – the relay broadcasts a function (marked χ) of data to destinations. The data in WPNC can be seen from at least two points of view: 1) discrete data symbols (we will mark them as b_A , resp. b_B), 2) constellation points symbols (we will mark them as $\mathcal{C}(b_A)$, resp. $\mathcal{C}(b_B)$).

Now we describe relay operation modes :

1. *Amplify and Forward (AF).* The relay only resend the received signal, that is at first amplified. It amplifies also a noise. From the point of view of decodability it is not the best way but it is interesting for the simplicity of the relay operation.
2. *Join Decode and Forward (JDF).* The relay decodes separately each source. We assume orthogonal signals, i.e.:

- time division multiple access (TDMA). Resp. above mentioned Network-Layer Network Coding.
 - frequency division multiple access (FDMA). Using different frequencies.
 - others
3. *Hierarchical Decode and Forward (HDF)*. The relay decodes *hierarchical symbol* – a network coded function of data (A and B): $\mathcal{C}(b_A, b_B) = \mathcal{C}(b_A) + \mathcal{C}(b_B)$. $\mathcal{C}(b_A)$, $\mathcal{C}(b_B)$ are constellation points of ordinary modulation, $\mathcal{C}(b_A, b_B)$ is a hierarchical (constellation) symbol. The relay doesn't have to be able to decode data separately, which reduces it's processing. Then at destination nodes to unambiguously decode data there is necessary a hierarchical information (HI) (in butterfly network 1.3 it is a line $S_A \rightarrow D_B$ and $S_B \rightarrow D_A$).

1.6 Two Way Relay Channel with BPSK

We assume HDF relay operation mode. The aim is to find such a function $\chi(b_A, b_B) = b_{AB}$ of data (applied by relay), that together with hierarchical information – data b_A , (resp. b_B at destination A) enables unambiguously decode data b_B . To define χ we will use a table. It has to be a function with values not repeating in any row, nor column. There are only 2 possibilities: XOR function or its inverse.

$b_A \setminus b_B$	0	1
0	0	1
1	1	0

Table 1.1. XOR – an example of the hierarchical function χ .

From the constellation point of view the signal received by relay is: $x[k] = h_A s_A[k] + h_B s_B[k] + w[k] = h_A(s_A[k] + h s_B[k]) + w[k]$. The discrete – sample-wise approach is used. The h_A, h_B represent attenuations of channel. The second part of the equation expresses the attenuation in two components: 1) common – h_A , linearly acts on both symbols; 2) relative – $h = h_B/h_A$, non-linear, deforms hierarchical symbol; The received signal can be modelled as a modulation of const. space symbols in form $\mathcal{C}(b_A, b_B) = \mathcal{C}(b_A) + h\mathcal{C}(b_B)$. We will study the relative attenuation. There are only two such h , that the hierarchical constellation symbols are only 3. In all other cases there are 4 H-symbols.

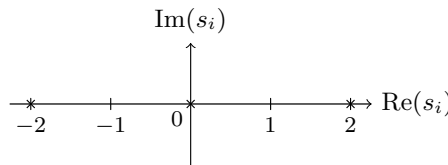


Figure 1.4. Hierarchical const. points for BPSK, $h = 1$.

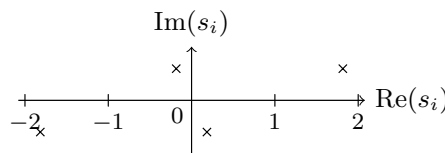


Figure 1.5. Hierarchical const. points for BPSK, $h = 0.9e^{j25\pi/180}$.

b_A, b_B	$\mathcal{C}(b_A), \mathcal{C}(b_B)$	$\mathcal{C}(b_A) + h\mathcal{C}(b_B) \rightarrow b_A \oplus b_B$	BPSK($b_A \oplus b_B$)
0, 0	-1, -1	-2 \rightarrow 0	-1
0, 1	-1, 1	0 \rightarrow 1	1
1, 0	1, -1	0 \rightarrow 1	1
1, 1	1, 1	2 \rightarrow 0	-1

Table 1.2. Mapping of hierarchical constellation points ($\mathcal{C}(b_A) + h\mathcal{C}(b_B)$) received at relay to BPSK symbols (for $h = 1$).

In the case $h = \pm 1$ there is a clash in zero point. The WPNC to operate correctly needs to assign these points in a clash to the same network coded symbol. If fulfilled we say the clash is *resolved*. In our case the mapping is (using XOR function) in the table: 1.2.

1.6.1 Demodulation and Decoding in 2WRC with BPSK

We assume AWGN channel model: $x[k] = h_A(s_A[k] + hs_B[k]) + w$. Let's set $h_A = 1$. The useful signal is $u[k] = s_A[k] + hs_B[k]$. Utilizing the linear modulation with Nyquist pulses we can decode symbol-wise. We will use a maximum likelihood estimator. Demodulator metric is a likelihood function $\Lambda(\mathcal{C}(b_A, b_B)) = p(x|\mathcal{C}(b_A, b_B))$, x – received signal, $\mathcal{C}(b_A, b_B)$ – hierarchical data symbols in constellation space. It can be shown that for high SNR can be the metric approximated by *euclidean (constellation space) distance*. The metric would be the same for equally distributed data, but our data symbols are distributed in this way: $p(-2) = 0.25$, $p(0) = 0.5$, $p(2) = 0.25$ due to the 2 BPSK pairs ($[-1, 1]$, $[1, -1]$) corresponding to hierarchical point 0. Nevertheless it is not optimal we will use the euclidean distance demodulator metric because of simpler implementation.

1.7 Two Way Relay Channel with QPSK

Also the HDF relay mode is assumed. Let's number constellation points as: $q_k = \exp(j\pi k/4)$, $k \in [0 : 3]$. The task is to find a hierarchical function χ resolving clashes for the relative attenuation $h = 1$. First we will plot the constellation points attributable to useful signal (we use $[b_A, b_B]$ labelling):

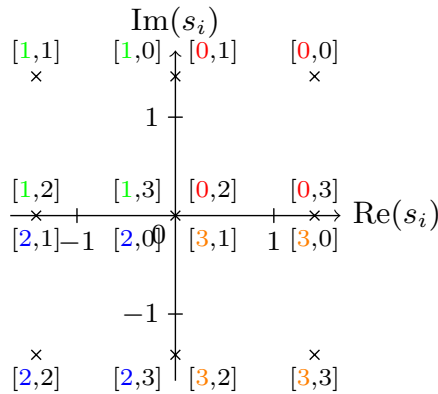


Figure 1.6. Hierarchical const. points for QPSK, $h = 1$.

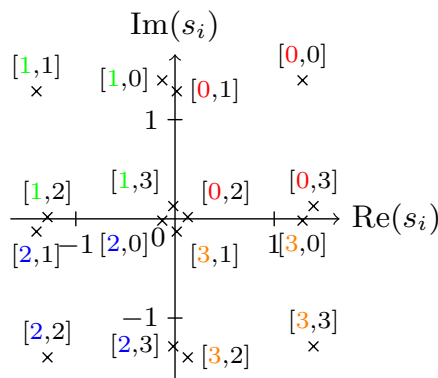


Figure 1.7. Hierarchical const. points for QPSK, $h = 0.9e^{j5\pi/180}$.

We have to find χ that resolves all clashes, i.e. all points in a clash have to be mapped to a one network coded symbol (ordinary QPSK point). The clash is e. g. a group of 4 points around (0,0), a group of 2 points above (0,1) ($[1,0]$, $[0,1]$). These are some examples of the mapping:

$s_A \setminus s_B$	0/00	1/01	2/10	3/11
0/00	0/00	1/01	2/10	3/11
1/01	1/01	3/11	0/00	2/10
2/10	2/10	0/00	3/11	1/01
3/11	3/11	2/10	1/01	0/00

$s_A \setminus s_B$	0/00	1/01	2/10	3/11
0/00	0/00	1/01	2/10	3/11
1/01	1/01	0/00	3/11	2/10
2/10	2/10	3/11	1/01	0/00
3/11	3/11	2/10	0/00	1/01

$s_A \setminus s_B$	0/00	1/01	2/10	3/11
0/00	0/00	1/01	2/10	3/11
1/01	1/01	0/00	3/11	2/10
2/10	2/10	3/11	0/00	1/01
3/11	3/11	2/10	1/01	0/00

Table 1.3. Some possible mappings χ (resolving clashes) for QPSK in $[0:3]$ (\mathbb{N} from 0 up to 3) and binary form.

Using Karnaugh map method, the last mapping can be recognized as bit-wise XOR function – linear function on a \mathbb{Z}_2 field. The others are non-linear mappings. We will mark the 9 constellation symbols by numbers $[1:9]$ (for $h = 1$) with the help of a table depicting constellation space points:

7	4	1
8	5	2
9	6	3

Table 1.4. Hierarchical symbols labelling for QPSK (for $h = 1$).

Demodulator metric in the 2WRC with QPSK is (as for BPSK) $\Lambda(\mathcal{C}(b_A, b_B)) = p(x|\mathcal{C}(b_A, b_B))$. Also for QPSK modulation we will use an ordinary euclidean distance metric even though it is not optimal.

Chapter 2

Error performance in 2WRC and Matlab implementation

We will deal with uncoded modulation ($q_n(d_n, \sigma_n) = d_n$) in discrete time in complex envelopes. The system model is the following: 1) data flow in MAC (Multiple Access Channel), 2) data flow from relay to destination B. We will examine the over all error rate of the data flow. The symbol-wise approach will be used - the data will not be grouped to packets.

2.1 Discrete data modulation

First, the data are created and stored in .csv files as long ($N = 10^5$) and short ($N = 10$) version. The data symbols are stored as $\{0, 1\}$ for BPSK, $\{0, \dots, 3\}$ for QPSK (symbol index). The creation and saving is done in the script `generate_data.m`. The data stored in the .csv files will be used to examine error rate.

Convolution implementation. To convolve 2 vectors (linear convolution - LTI filtering) we pad the vectors by zeros so to have length equal to sum of vectors lengths - 1 (the length of the linear convolution of the original vectors). It can be shown, that the circular convolution of so padded vectors is equal to a linear convolution of original vectors. The circular convolution of two vectors is done using Discrete Fourier Transform (DFT) by inverting element-wise product of their spectra (DFT of vectors). The implementation is done in the function `convol.m`.

Data modulation. To modulate data the RRC or REC modulation pulse will be used, they are sampled from the continuous time and scaled so to have unit energy (in discrete time) defined as: $E_g = \frac{1}{2} \sum_{i=0}^{N_{\text{samp}}-1} |g[i]|^2$. N_{samp} is number of the pulse samples. The vectors of samples of the pulses are created by functions `REC_pulse.m`, resp. `RRC_pulse.m`. The modulated signal samples are of the form:

$$s[k] = \sum_{n=0}^{N-1} d[n]g[k - nN_s]. \quad (1)$$

(the data length is assumed to be N) N_s is symbol period – number of samples associated with one symbol. To get such a signal $s[k]$, the following procedure is done:

- The data are zero-padded by $N_s - 1$ zeros. The zero-padded data sequence can be expressed as a weighted sum of (discrete) Dirac pulses: $(d_k)_{k=0}^{N-1} = d[k] = \sum_{n=0}^{N-1} d_n \delta[k - nN_s]$.
- Then the data are filtered by a modulation pulse: $\left(\sum_{n=0}^{N-1} d_n \delta[k - nN_s] \right) * g[k] = \sum_{n=0}^{N-1} d_n g[k - nN_s]$ (filter response to a shifted Dirac pulse is the same shifted impulse response).

The procedure is done by function `modulate_data.m`. It takes the data in constellation space, modulation pulse samples and symbol period N_s .

The theory assume pulses to be non-causal, starting from (assume pulse length N_p odd) $(N_p - 1)/2$, ending in $(N_p - 1)/2$. We will use causal pulse. Let's label the shifted pulse samples as \bar{g} pulse samples (see Fig. 2.1).

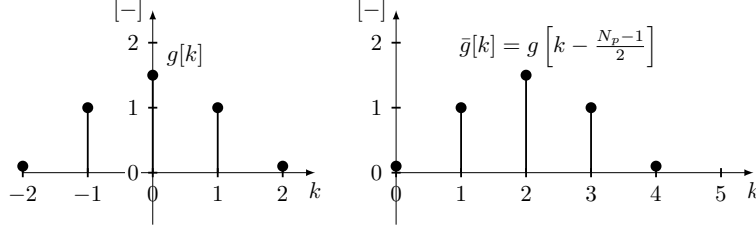


Figure 2.1. Schematically depicted a theoretical pulse samples $g[k]$ and the causal pulse samples: $\bar{g}[k]$. In the picture $N_p = 5$.

Let's mark the zero padded data (used by modulation procedure (1)) by d_δ (data impulse train). When calculating (linear) convolution of the pulse and d_δ sequence (numbered from 0 to $(N - 1)N_s$), the output is:

$$\begin{aligned}
 (d_\delta * g)[k] &= \sum_{l=-(N_p-1)/2}^{(N_p-1)/2} d_\delta[k-l]g[l] = \sum_{l=-(N_p-1)/2}^{(N_p-1)/2} d_\delta[k-l]\bar{g}\left[l + \frac{N_p-1}{2}\right] \\
 &= \left| p := l + \frac{N_p-1}{2} \right| = \sum_{p=0}^{N_p-1} d_\delta\left[k - \left(p - \frac{N_p-1}{2}\right)\right] \bar{g}[p] \\
 &= (d_\delta * \bar{g})\left[k + \frac{N_p-1}{2}\right].
 \end{aligned} \tag{2}$$

The convolution using original pulse samples is equal to the convolution using shifted pulse samples in the opposite shifted output point.

2.2 Data flows in 2WRC and error rate

We examine the data flow in the functions `twrc_bpsk.m` and `twrc_qpsk.m`. The functions take data vectors (from binary, resp. $\{0, \dots, 3\}$ set), pulse samples, symbol period N_s , relative channel attenuation h , and SNR (signal to noise ration, in [dB]). The procedure is following:

1. First phase - MAC. From the integer data the constellation points are computed. The constellation data vectors are marked with suffix `const`.
2. Then the data are modulated using pulse samples (`pulse`). They are marked as `sA`, resp. `sB`.
3. The $\mathbf{x} = \mathbf{sA} + \mathbf{h} * \mathbf{sB}$ (useful) sample-wise signal is created.
4. A complex noise samples \mathbf{w}_- are created using standard function `randn`. The function returns real random numbers of normal distribution with variance $\sigma = 1$. \mathbf{w}_- is composed of the normally distributed real and imaginary part. It is normed by $1/\sqrt{2}$ factor so to have unit complex variance.
5. The \mathbf{w} noise is created so to respect given SNR (often marked as γ) w.r.t. source A - s_A samples. We consider γ to be defined as: $\gamma = E_b/N_0$, ($N_0 = E\gamma^{-1}$), where E_b is an energy per 1 bit of information, N_0 is a power spectral density of AWGN. The noise variance we will assume to be $\sigma_w = (2N_0)^{1/2}$. For BPSK and QPSK the symbol energy is $E_s = \frac{1}{2}$. Then a bit energy of BPSK symbols is $E_b = E_s/\log_2(2) = 1/2$, bit energy of QPSK symbols is $E_b = E_s/\log_2(4) = 1/4$. Using $\gamma_{dB} = 10 \log_{10}(\gamma)$, ($\gamma = 10^{\gamma_{dB}/10}$) we can express the noise variance as: $\sigma = 10^{-\gamma_{dB}/20}$ for BPSK, $\sigma =$

$\frac{1}{\sqrt{2}}10^{-\gamma_{dB}/20}$. The w noise is then $w = 2 \cdot \text{sigma} \cdot w$. The multiplying by 2 is there because of working in complex envelopes.

6. The noise is additively added to useful signal x : $y = x + w$.
7. The received data are demodulated from y samples by convolution with flipped pulse samples (as in the Fig.1.1) and should be sampled, but need to be slightly arranged because of pulse causality. The REC and RRC pulses are symmetric, so $g[k] = g[-k]$ (for theoretical pulse) and real: $g[k] = g^*[k]$. So the matched filtering is equivalent to filtering using $g[k]$ samples. Mark $d_{A\delta}$ – impulse train of data from A and $d_{B\delta}$ – impulse train of data from B . We can notice, that useful signal: $u[k] = s_A[k] + h s_B[k] = (d_{A\delta}[k] + h d_{B\delta}[k]) * g[k]$ can be formed as a convolution of $d_{AB\delta} := d_{A\delta} + h d_{B\delta}$ with pulse samples g , i.e. a modulation of the hypothetical constellation symbols $\mathcal{C}_{AB}[k] := \mathcal{C}(b_A, b_B)[k] = \mathcal{C}(b_A[k], b_B[k]) = \mathcal{C}(b_A[k]) + h \mathcal{C}(b_B[k])$. The following holds:

$$y[k] = x[k] + w[k] = d_{AB\delta}[k] * g[k] + w[k] \stackrel{(2)}{=} (d_{AB\delta} * \bar{g}) \left[k + \frac{N_p - 1}{2} \right] + w[k] \quad (3).$$

Now define y_1 vector as:

$$y_1[k] = (d_{AB\delta} * \bar{g})[k] + w \left[k - \frac{N_p - 1}{2} \right] = y \left[k - \frac{N_p - 1}{2} \right]. \quad (4)$$

The y can be expressed as:

$$y[k] = y_1 \left[k + \frac{N_p - 1}{2} \right].$$

Mark the yy vector as the demodulator filter output:

$$\begin{aligned} yy[k] &= (y * g)[k] = (y * \bar{g}) [k + (N_p - 1)/2] \\ &= (\bar{g} * y) [k + (N_p - 1)/2] = (\bar{g} * y_1) [k + 2(N_p - 1)/2] \\ &= (\bar{g} * d_{AB\delta} * \bar{g})[k + (N_p - 1)] + \bar{g}[k] * w[k - (N_p - 1)/2] \\ &= d_{AB\delta}[k + (N_p - 1)] + (g * w)[k]. \end{aligned} \quad (5)$$

We have used the Nyquist property of a real symmetric pulse:

$$\begin{aligned} \langle \bar{g}[k - mN_s], \bar{g} \rangle &= \sum_k \bar{g}[k - mN_s] \bar{g}^*[k] = \sum_k \bar{g}[mN_s - k] \bar{g}[k] \\ &= \bar{g}[k - mN_s] * \bar{g}[k] = \delta_{0m}. \end{aligned} \quad (6)$$

The $(g * w)$ part is a noise projection to the g pulse. Mark that as \bar{w} .

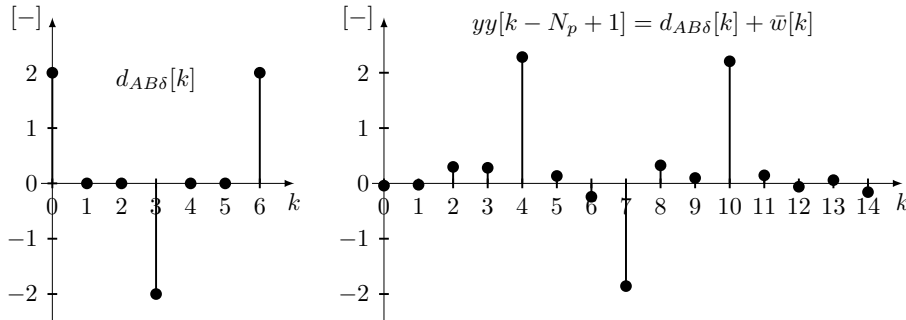


Figure 2.2. Schematically depicted an example of $d_{AB\delta}$ impulse train and the matched filter output yy for $N_p = 5$, $N_s = 3$, $h = 1$.

The output: yy is periodically sampled with period N_s . It can be seen, that we need to start sampling yy vector at the N_p -th sample. Therefore in the Matlab implementation the yy is cropped from the beginning by $N_p - 1$ initial symbols.

8. Next, the detection is explained. The detection is divided to the two approaches: 1) with unknown channel attenuation, 2) with known channel attenuation. The switching between the methods is done by commenting out a not required part and letting uncommented the desired part (in the functions `twrc_bpsk.m`, and `twrc_qpsk.m`).

- Detection with *unknown* channel attenuation h . The samples of demodulation filter output z are provided to likelihood function to test for every H-symbol (Hierarchical symbol) where expected singular fading $h = 1$. So, for BPSK there are 3 H-symbols and for QPSK there are 9 H-symbols. We save the symbol (by its index) for that the likelihood function is maximized in vector `outputR`. Then the χ mapping is done from detected H-symbols to the ordinary BPSK, resp. QPSK. χ is chosen to be bitwise XOR function.
- Detection with *known* channel attenuation h . All possible H-symbols are created respecting discrete data order. They are marked as `H_BPSK_symb`, resp. `H_QPSK_symb`. The general const. point is found so to maximize channel likelihood function. The maximization is done for every filter output sample z ($\in \mathbb{C}$). The extreme is found using generalized (Wirtinger) derivative (because of the Λ function is not satisfying Cauchy-Riemann conditions) The z is a sample of the filter output, from that we try to detect broadcast H-symbol. Mark that H-symbol by c . The sample z is constant when finding c .

$$\begin{aligned}\Lambda(c) &= \text{Re}\{c^*z\} - |c|^2E = \frac{c^*z + cz^*}{2} - cc^*E, \\ \frac{\tilde{\partial}\Lambda}{\partial c} &= \frac{z^*}{2} - c^*E \stackrel{!}{=} 0, \\ c^* &= \frac{z^*}{2E}, \\ c &= \frac{z}{2E}.\end{aligned}\tag{7}$$

The Wirtinger derivative treats c^* as a constant (when defferentiating w.r.t. c) and gives the same extrema points as using transformation to Cartesian coordinates x , y and finding point for that the gradient is equal zero. So the extreme of likelihood function is reached when H-symbol $c = z/(2E)$. Then the closest to the c regular (without any noise) H-symbol is found (from the `H_BPSK_symb`, resp `H_QPSK_symb` points). The H-symbol is remembered (in vector `outputR`). The indexes of H-points are used to find out which were the original b_A and b_B symbols. And from the b_A , b_B symbols is computed $\chi = \text{bit-wise XOR}$ (hierarchical) function. The function is correct also for singular fading $h = 1$, because we have mapped all symbols from a clash to one BPSK/QPSK symbol.

9. Second phase – BC: R broadcasts data to D_B and D_A . We are interested in the D_B . The ordinary modulation is used (BPSK, resp. QPSK). Data are converted to constellation symbols and modulated. `sR` contains samples of the modulated broadcast signal. There is only the common channel attenuation, that we have set to 1 in MAC phase. So channel attenuation is assumed 1.
10. The complex noise is created and added to useful signal. $y2 = sR + w$
11. The demodulator filtering and cropping output is done to prepare the filter output vector for comfortable sampling.

12. Then a classical detection is done. We use values of likelihood function in the (non-scaled) ordinary BPSK, resp. QPSK points. We are finding the point for which the Λ is maximized. The point is stored in vector `outputA` (using the point index).
13. Then, in discrete domain of numbers from sets $\{0, 1\}$, resp. $\{0, \dots, 3\}$, the original data (`dataA`, `dataB`) are used. We form the χ function (bit-wise XOR) of them and compare with the received data at destination B.
14. Finally, the error rate is computed. R_{err} – number of wrong transmitted symbols divided by number of all symbols.

2.3 A note on a detector

We may notice, that detection processing (for known h) – finding the closest H-point to the c point, that maximizes Λ function, divides the constellation space to cells. Every cell contains a H-point in the middle and points that are nearer to the cell central H-point than to the all other H-points. Such a plain division to cells is called *decoder decision regions*. The division can be obtained as a Voronoi diagram. This is a result of using a detector metric based on Euclidean distance.

Now take a look to 2WRC with BPSK modulation. We will examine a right detector metric. Here we will assume the Λ to be function of an ordinary BPSK points that are created at relay by mapping the H-points to the ordinary constellation points. Let's mark a BPSK symbol by α . The right detector metric is:

$$\Lambda(\alpha) = p(z|\alpha). \quad (8)$$

The z is the H-point (constellation point $\in \mathbb{C}$) minimizing demodulator metric. The value of Λ is for the same $\mathcal{C}(b_A)$, $\mathcal{C}(b_B)$ (resp. b_A , b_B) symbols:

$$\begin{aligned} \Lambda(-1) &= p(z|-1) = p(z|\mathcal{C}(b_A) = -1, \mathcal{C}(b_B) = -1) \frac{p(\mathcal{C}(b_A) = -1)p(\mathcal{C}(b_B) = -1)}{p(\mathcal{C}_{AB} = -1)} \\ &\quad + p(z|\mathcal{C}(b_A) = 1, \mathcal{C}(b_B) = 1) \frac{p(\mathcal{C}(b_A) = 1)p(\mathcal{C}(b_B) = 1)}{p(\mathcal{C}_{AB} = -1)} \\ &= p(z|\mathcal{C}_{AB} = -1 - h)/2 + p(z|\mathcal{C}_{AB} = 1 + h)/2 \\ &= p_w(z - (-1 - h))/2 + p_w(z - (1 + h))/2 \\ &= \frac{1}{2\sqrt{2\pi\sigma_w^2}} \left[e^{-|z - (-1 - h)|^2 / (2\sigma_w^2)} + e^{-|z - (1 + h)|^2 / (2\sigma_w^2)} \right]. \end{aligned} \quad (9)$$

Similarly for the different b_A , b_B symbols:

$$\Lambda(1) = \frac{1}{2\sqrt{2\pi\sigma_w^2}} \left[e^{-|z - (1 - h)|^2 / (2\sigma_w^2)} + e^{-|z - (-1 + h)|^2 / (2\sigma_w^2)} \right]. \quad (10)$$

We would like to find the border between the detector regions for ± 1 . The z points forming border is characterized by a condition: $\Lambda(-1) = \Lambda(1)$. By an adjusting the relation with exponentials we get the reformed relation:

$$\tanh\left(\frac{\text{Re}\{z\}}{\sigma_w^2}\right) \tanh\left(\frac{\text{Re}\{zh^*\}}{\sigma_w^2}\right) = \tanh\left(\frac{\text{Re}\{h\}}{\sigma_w^2}\right). \quad (11)$$

If we choose $z = x + jy$ and $h = h_x + jh_y$, the relation is:

$$\tanh\left(\frac{x}{\sigma_w^2}\right) \tanh\left(\frac{xh_x + yh_y}{\sigma_w^2}\right) = \tanh\left(\frac{h_x}{\sigma_w^2}\right). \quad (12)$$

Now, some noise variances σ_w were chosen to depict a difference between decision regions acquired by the euclidean decoder metric and the right decoder metric. The σ_w is assumed for continuous time signals. The coloured lines show the right decision border for the attributable noise variance.

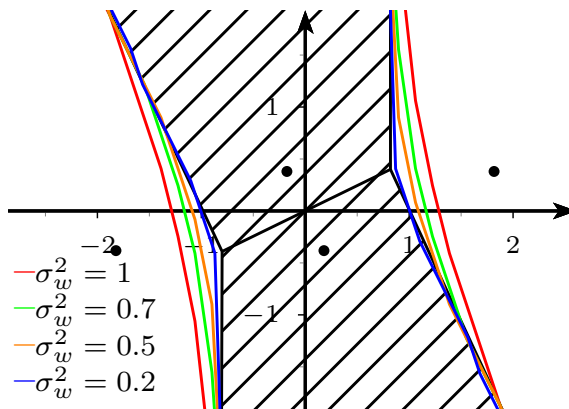


Figure 2.3. A decoder decision region for BPSK, $h = 0.9 \exp(j25\pi/180)$ based on the euclidean distance – hatched, and the correct decoder metric in dependence on σ_w^2 values.

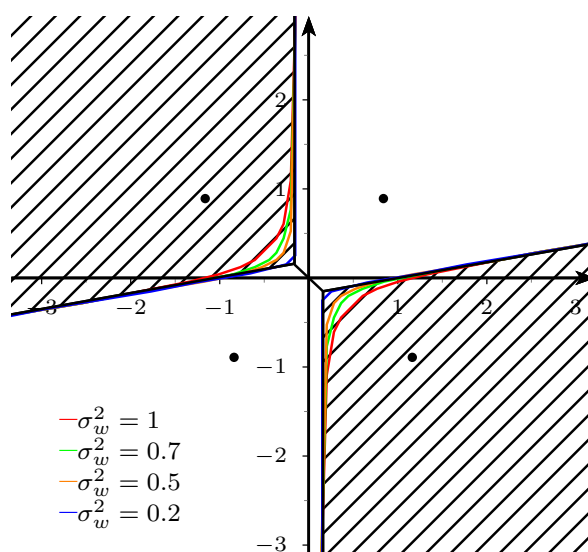


Figure 2.4. A decoder decision region for BPSK, $h = 0.9 \exp(j100\pi/180)$ based on the euclidean distance – hatched, and the correct decoder metric in dependence on σ_w^2 values.

The computation of the borders is done in the Matlab script `real_decoder_metric.m`. First relative channel attenuation \mathbf{h} is chosen. The values of σ_w^2 are selected. x, y grid is created. All possible H-points are generated. A Voronoi diagram is drawn. For all sigma values a noise probability density function is defined. A logical matrix variable `ind` indicates the points in x, y grid, that belong to the region for the relay BPSK symbol 1. Then the grid is plotted.

For the 2WRC with QPSK only the detection regions based on the euclidean distance are stated. The same hatched regions belong to a one relay mapped QPSK symbol not only in one picture.

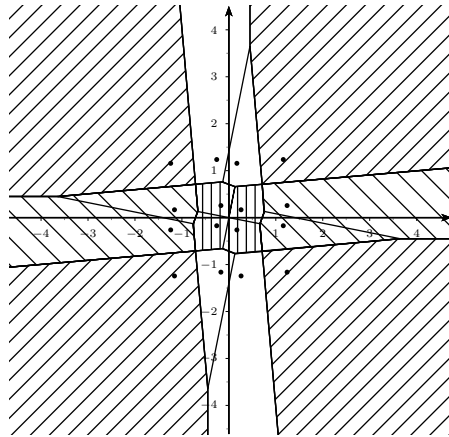


Figure 2.5. A decoder decision region for QPSK, $h = 0.7 \exp(j5\pi/180)$ based on the euclidean distance.

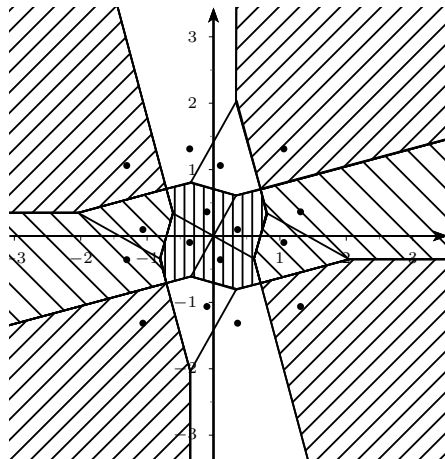


Figure 2.6. A decoder decision region for QPSK, $h = 0.7 \exp(j15\pi/180)$ based on the euclidean distance.

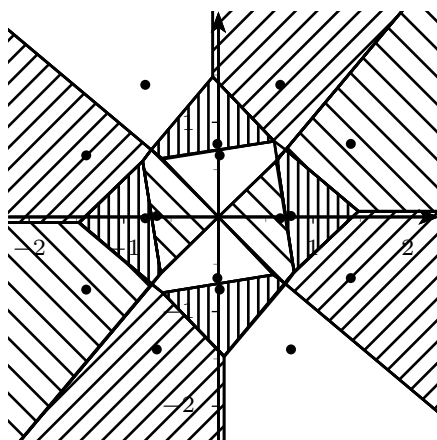


Figure 2.7. A decoder decision region for QPSK, $h = 0.7 \exp(j50\pi/180)$ based on the euclidean distance.

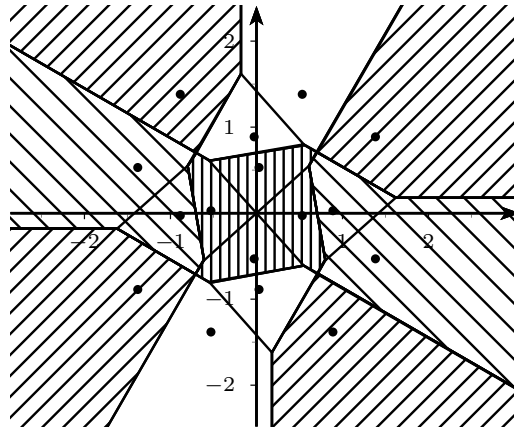


Figure 2.8. A decoder decision region for QPSK, $h = 0.7 \exp(j150\pi/180)$ based on the euclidean distance.

2.4 Conclusion and results

The functions `twrc_bpsk.m` and `twrc_qpsk.m` were called in script `using_func.m`. The data generated by script `generate_data.m` were used as a demonstration data (the long version with 10^5 symbols). A processing in the script `using_func.m` consist of:

- First there is chosen a pulse sampling period $T_{\text{samp}} = 0.02$. The symbol period were chosen $N_s = 5$. The symbol period is parallelly expressed in continuous time as $T_s = N_s * T_{\text{symb}}$. The L is the number of interfering symbols. It should be one for REC pulse and ≥ 1 for RRC pulse, due to the REC pulse is Nyquist only for the $L = 1$ in contrast with RRC pulse, that is Nyquist for all $L \geq 1$. According to L we choose pulse, that is assigned to vector \mathbf{g} as the pulse samples.
- Then a relevant data are read. The attenuation is chosen by it's amplitude a and phase in degrees ϕ .
- Then for each combination of channel damping and SNR value in [dB] is computed an error rate R_{err} . The same procedure applies to QPSK data.

For BPSK data the following results were reached:

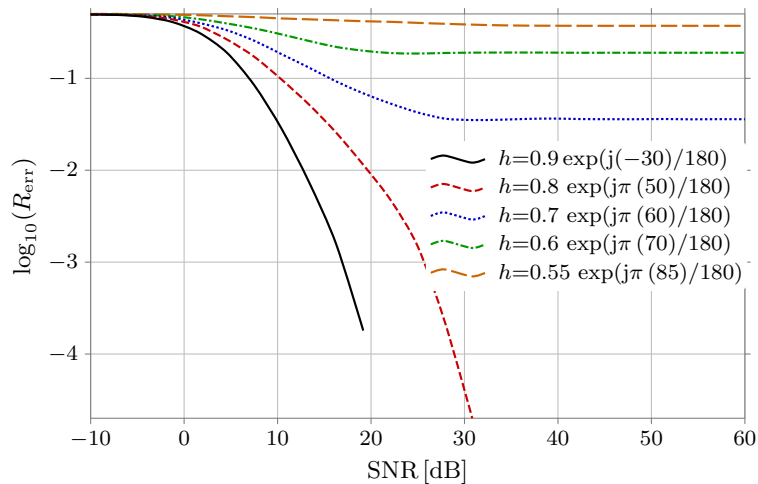


Figure 2.9. The error rate in 2WRC channel using BPSK depending on SNR with h unknown .

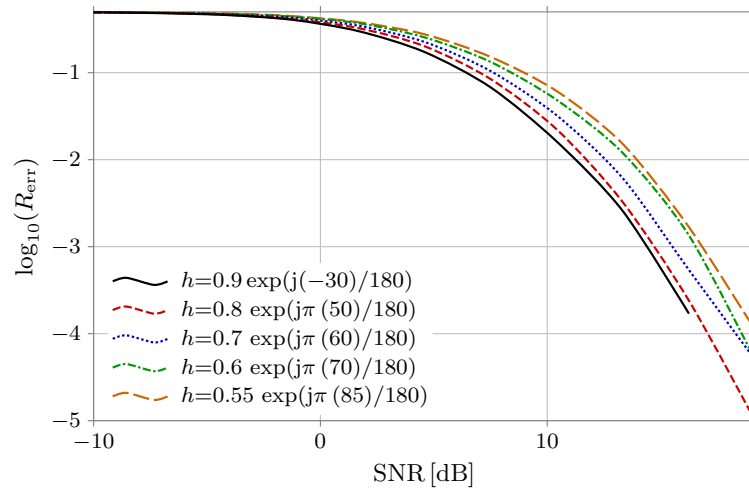


Figure 2.10. The error rate in 2WRC channel using BPSK depending on SNR with \mathbf{h} known.

For the QPSK data the following dependencies were identified:

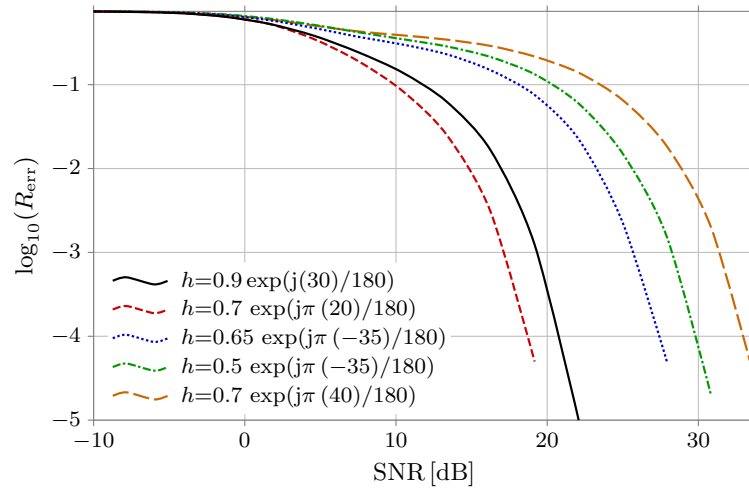


Figure 2.11. The error rate in 2WRC channel using QPSK depending on SNR with \mathbf{h} unknown.

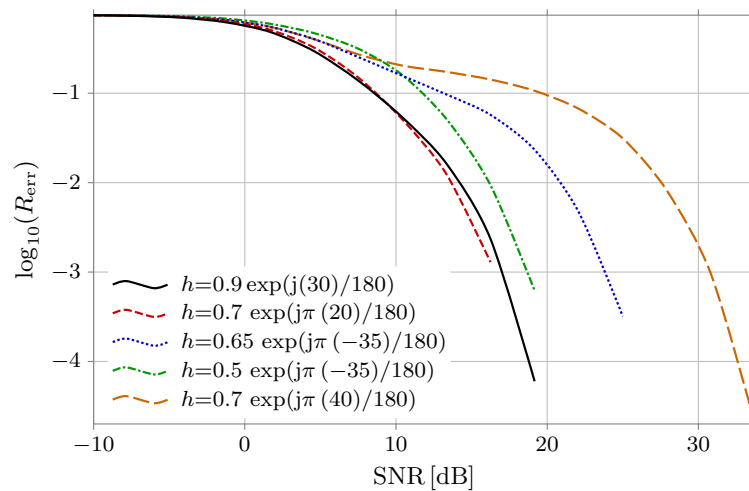


Figure 2.12. The error rate in 2WRC channel using QPSK depending on SNR with \mathbf{h} known.

The number of symbols used for the figures was 10^5 . When processing more times we can observe slightly different plots. It is observable for the $\log_{10}(R_{\text{err}})$ less than cca -4 . I.e. there was less than $10^{(-4)} \cdot 10^5 = 10$ error symbols. The low number is the reason. From the plots it is apparent, that we can't ignore knowledge of relative channel attenuation h .



References

1. Sýkora Jan, Burr Alister. *Wireless Physical Layer Network Coding.*, 2016.
2. Weisstein, Eric W. *Vector Space.*, <http://mathworld.wolfram.com/VectorSpace.html>, 2017
3. Sýkora Jan. *Digital Communications – lecture slides.* 2016.

Appendix A

Attached files

```
data
|-----data_BPSK.csv
|-----data_BPSK_long.csv
|-----data_QPSK.csv
|-----data_QPSK_long.csv

matlab_files
|-----convol.m
|-----generate_data.m
|-----modulate_data.m
|-----real_decoder_metric.m
|-----REC_pulse.m
|-----RRC_pulse.m
|-----twrc_bpsk.m
|-----twrc_qpsk.m
|-----using_func.m
```