TECHNISCHE UNIVERSITÄT ILMENAU

Fakultät für Informatik und Automatisierung

Robust Distributed Communication Techniques

Masterarbeit

zur Erlangung des akademischen Grades Masteringenieur

vorgelegt von:	Soheyl Gherekhloo
geboren:	09. September.1985 in Mashhad, Iran
Studiengang:	Elektrotechnik und Informationstechnik
Studienrichtung:	Informationstechnik und Kommunikationstechnik
Anfertigung im Fachgebiet:	Nachrichtentechnik Institut für Informationstechnik Fakultät für Elektrotechnik und Informationstechnik
verantwortlicher Professor:	Prof. Dr.-Ing. Martin Haardt
Erster wissenschaftlicher Betreuer:	M.Sc Bilal Zafar
Zweiter wissenschaftlicher Betreuer:	Dipl.-Ing. Florian Römer
Datum:	3. August 2016

TECHNISCHE UNIVERSITÄT ILMENAU

Fakultät für Elektrotechnik und Informationstechnik

Masterarbeit

Tensor based system modelling for GFDM

vorgelegt von:	Bulat Valeev
geboren am:	13. 06. 1991 in Kazan
Studiengang:	Elektrotechnik und Informationstechnik
Studienrichtung:	Communication and signal processing
Anfertigung im Fachgebiet:	Communication and signal processing
	Fakultät für Elektrotechnik und Informationstechnik
Verantwortlicher Professor:	Univ.-Prof. Dr.-ing.Martin Haardt
Wissenschaftlicher Betreuer:	M. Sc. Kristina Naskovska
Datum:	15 Aug 2016

Acknolegment

Firstly, I would like to express my sincere gratitude to the GRIAT organization personal from the Kazan. There are a lot of them, thanks for giving me chance to start this thesis Secondly, I am very grateful to Kristina Naskovska for help and supervision. Without her advice, most of this work would not have been possible Also I am very grateful to the "RIIT" department in the University where I studied the bachelor degree. They supported a lot throughout my master research. Finally I would like to thank my family for constantly supporting me in work during the last half of year in Kazan. It is the great success for me to achieve this point in the life

Abstract

The radio resources become more and more expensive nowadays. The requirements for the transmission systems become more and more demanding. The necessary transmission rate also become more and more high. This demand increase complexity for a designed systems.

Generalized frequency division modulation is the new approach to increase spectral efficiency in comparison to the modern OFDM scheme. GFDM uses time mixing for the symbols in one transmission block to increase density of the transmitted data. The new approach is flexible and can adjust a frequency overlap with one variable. The frequency overlap slightly increase the SER of the system, but decrease out-of-band radiation. The system also can adjust the number of turned on subcarriers in case if frequency sharing used. The system can wok in the same frequency bands as ther legal users and doesn't interfere with them. The transmission system become complex in the symbol estimation, but introduce the self-interference between symbols.

GFDM modulation can be explained with the PARATUCK2 tensor model to present modulation scheme in more efficient and natural way. This model explain the transmitted data as the slice-wise product between three different matrices and two tensors. This notation allow to introduce time and frequency domain separately for the symbols in one transmission block. The PARATUCK2 model can be exploited to find more efficient ways to estimate unknown symbols or update channel variables

Contents

1 Introduction

Today radio resources become expensive due to the huge number of used systems. Smart ways to radio resource allocation is getting more and more important in wireless communication. The spectrum sharing ways should be done very carefully, to hold interference for legal systems. Additional approaches to increase frequency efficiency raise the complexity of the wireless systems [Cox08]. The fourth generation was the OFDM system. The OFDM system doesn't provide the necessary inter-system interference level for the nearby frequency bands [MMI14][Skl01]. In addition, the OFDM system based on the orthogonality between different sub-carriers. OFDM use additional time domain filter to archive orthogonality between sub-carriers. In the next generation proposed more efficient frequency resources usage. A new technique is called generalized frequency division modulation. It based on the non-orthogonal sub-carrier location [MMI14] [MMF14]. The transmitter put the sub-carrier frequencies with overlap. The system provides such behaviour with root-raised cosine filters instead of the *sinc* filters. Selected approach allow decreasing out-of-band radiation, but introducing self-interference in the system. The self-interference decrease performance in the system. New filter design add flexibility to the system because has the adjusting coefficient. The system adjust self-interference to find the trade-off between performance and frequency efficiency. Another requirement is spectrum-sensing approach for the transmitter and receiver side. For that case receiver should estimate both the transmitted symbols and the used sub-carriers. This task has uncertainty in case if all symbols is unknown. To estimate both of the variables the receiver should know additional information. The receiver may know the first column of the transmitted symbol matrix. All explained approaches make receiver and transmitter more complex. The receiver in the new system must solve much more complicated task to provide the same BER. The algorithms to symbol estimation also become more complicated. There is a way to simplify the definition of the modulation matrix. We can use the tensor algebra and explain the transmission with a more natural and easy way.

1.1 Tensor based processing

A tensor is a multidimensional array (1.1)[KB09]. In other words, the Nth-order tensor is a matrix which has elements over N different dimensions. The first and second order tensors correspond to the arrays and matrices from linear algebra[BSG97]. The tensors, which has number of dimension higher or equal to three are called higher order tensors.

$$\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \cdots I_N} \tag{1.1}$$

The order of the tensor is the number of dimensions which tensor has. In this thesis scalars are denoted by the lowercase letters e.g. a. The arrays are denoted by boldface lowercase letters e.g. **a**. The matrices are written as the boldface capital letters e.g. **A**. Higher order tensors are denoted by capital calligraphic letters e.g. $\mathcal{A}$. Tensors explain data with high number of variables in a more natural way. Tensor based explanation of the data allow to consider all dependencies between dimensions of the data. The N-th order unfolding of the tensor is the stacked version of the tensor with respect to the N-th dimension (1.2)(1.3)(1.4)(1.5)[LMV00]. In the N-th order unfolding the elements of the each columns is fixed for the all dimensions except N. The N-th order unfolding has the number of row equal to the N dimension size, the number of the column equal to the product between other dimensions. The order of the columns are defines as the notation of the unfolding. In this paper is considered MATLAB ordering[AB00]. The MATLAB notation change dimensions in the increasing order.

$$\mathcal{X}_{:,:,1} = \begin{bmatrix} 1 & 3 \\ 0 & 1 \end{bmatrix} \ \mathcal{X}_{:,:,2} = \begin{bmatrix} 2 & -3 \\ -1 & 2 \end{bmatrix} \tag{1.2}$$

$$\mathcal{X}_{[1]} = \begin{bmatrix} 1 & 3 & 2 & -3 \\ 0 & 1 & -1 & 2 \end{bmatrix} \tag{1.3}$$

$$\mathcal{X}_{[2]} = \begin{bmatrix} 1 & 0 & 2 & -1 \\ 3 & 1 & -3 & 2 \end{bmatrix} \tag{1.4}$$

$$\mathcal{X}_{[3]} = \begin{bmatrix} 1 & 0 & 3 & 1 \\ 2 & -1 & -3 & 2 \end{bmatrix} \tag{1.5}$$

The N-th order tensor is rank-one if it can be presented as the outer products of N vectors. The Hitchcock proposes the tensor decomposition in the 1927. He expressed the higher order tensor as the sum of the rank-one tensors (1.6)(1.7).

$$\mathcal{X} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \tag{1.6}$$

$$\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times I_3} \mathbf{a}_r \in \mathbb{C}^{I_1 \times 1} \mathbf{b}_r \in \mathbb{C}^{I_2 \times 1} \mathbf{c}_r \in \mathbb{C}^{I_3 \times 1} \tag{1.7}$$

The tensor can be decomposed as the number of rank-one tensors. Decomposition called "CP" or CANDECOMP/PARAFAC decomposition[KB09]. The "CP" decomposition factorizes the tensor into a sum of rank-one tensors components. Sum of rank-one tensor can be presented in another form (1.8) . The arrays corresponding to the same dimension can be stacked in the one matrix for each dimension respectively. This notation is called the "factor matrices". The factor matrices present the tensor unfolding in simplified form (1.11)(1.12)(1.13).

$$\mathcal{X} = \mathcal{I} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} \tag{1.8}$$

$$\mathcal{X} \times_n \mathbf{A} = \mathbf{A} \mathcal{X}_{[n]} \tag{1.9}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 \cdots \mathbf{a}_R \end{bmatrix} \tag{1.10}$$

$$\mathcal{X}_{[1]} = \mathbf{A}(\mathbf{C} \diamond \mathbf{B})^T \tag{1.11}$$

$$\mathcal{X}_{[2]} = \mathbf{B}(\mathbf{C} \diamond \mathbf{A})^T \tag{1.12}$$

$$\mathcal{X}_{[3]} = \mathbf{C}(\mathbf{A} \diamond \mathbf{B})^T \tag{1.13}$$

There are many decomposition algorithms. One of the most popular is the Alternating least squares algorithm. The algorithm is explained as follows:

- Set **C** and **B** as random valued matrices
- Solve equation for the first unfolding with respect to the **A** with given **C** and **B**
- Solve equation for the first unfolding with respect to the **B** with given **C** and **A**
- Solve equation for the first unfolding with respect to the **C** with given **A** and **B**
- Check the tolerance of residual, if residual higher than tolerance repeat from step 2

1.1.1 PARATUCK2 model

The PARATUCK2 model is the combination between the PARAFAC [BSG97] and TUCKER2 [KB09] tensor models. The PARATUCK2 [KB09] combines decomposition between five different matrices into one tensor. The PARATUCK2 model in general is explained as the three-dimensional tensor and is defined in scalar form as given in (1.14) [AFX13]

$$x_{i_1,i_2,t} = \sum_{f=1}^{F} \sum_{t_s=1}^{T_s} a_{i_1,f} c^{[a]}_{t,f} s_{f,t_s} c^{[b]}_{f,t_s} b_{i_2,t_s} \tag{1.14}$$

$$\mathcal{X} \in \mathbb{C}^{I_1 \times, I_2 \times T} \mathbf{A} \in \mathbb{C}^{I_1 \times F} \mathbf{B} \in \mathbb{C}^{I_2 \times T_s}$$

$$\mathbf{C}^{[a]} \in \mathbb{C}^{T \times F} \mathbf{C}^{[b]} \in \mathbb{C}^{T \times T_s}$$

where $x_{i_1,i_2,t}$ is (i_1, i_2, t) -th entry of the resulting tensor $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times T}$.The **S** matrix is the core matrix of the PARATUCK2 model. The matrices **A** and **B** show connection between the core matrix $\mathbf{S}$ to the first and the second dimensions. The $\mathbf{C}^{[\mathbf{a}]}$ and $\mathbf{C}^{[\mathbf{b}]}$ are the scaling matrices defining the $\mathbf{A}$ and $\mathbf{B}$ relations to the core matrix $\mathbf{S}$. Every element of the core matrix links the $\mathbf{A}$ and $\mathbf{B}$ matrices as the scaling coefficient[BSG97].

The PARATUCK2 model possible to write in the two forms: slice-by-slice multiplication form and vectorized form. In the slice-by-slice product form the equation for the tensor is written for each slice separately (1.15). Explained form allow writing the model as matrix multiplication with change only in two matrices for different slices. The $\mathbf{C}^{[\mathbf{a}]}$ and $\mathbf{C}^{[\mathbf{b}]}$ is written as the tensor with elements on the diagonal in the first and second dimension. The linear combination between tensors become the scaling coefficients for columns in the each i_3 point. In that case the PARATUCK2 model will be written as the slice-wise multiplication (1.15) over the matrices $\mathbf{A}$ and $\mathbf{B}$ from the corresponding side with diagonalized tensors $\mathbf{C}^{[\mathbf{a}]}$ and $\mathbf{C}^{[\mathbf{b}]}$ for the corresponding slice[BSG97].

$$\mathbf{X}_{:,:,i} = \mathbf{A} \cdot diag(\mathbf{C}^{[\mathbf{a}]}{}_{:,i}) \cdot \mathbf{S} \cdot diag(\mathbf{C}^{[\mathbf{b}]}{}_{:,i}) \cdot \mathbf{B} \tag{1.15}$$

Vectorized form of tensor $\mathcal{X}$ is another form to write the PARATUCK2 model(1.16) . We can write three equations with respect to the vectorized versions of the $\mathbf{A}$ (1.17),$\mathbf{S}$(1.16),$\mathbf{B}$(1.18)[AFX13].Structure of the tensor in the defined equation is lost, because we have only vectorized version of the tensor. The slice-wise model is replaced via matrix operations and allow operating with matrices instead of the tensors of different matrix operation for different slices[AFX13].

$$vec(\mathcal{X}) = \begin{bmatrix} (\mathbf{C}^{[\mathbf{a}]\mathbf{T}} \diamond \mathbf{C}^{[\mathbf{b}]\mathbf{T}})^T \diamond (vec(\mathbf{A}_{1,:}) \otimes \mathbf{B}) \\ (\mathbf{C}^{[\mathbf{a}]\mathbf{T}} \diamond \mathbf{C}^{[\mathbf{b}]\mathbf{T}})^T \diamond (vec(\mathbf{A}_{2,:}) \otimes \mathbf{B}) \\ \vdots \\ (\mathbf{C}^{[\mathbf{a}]\mathbf{T}} \diamond \mathbf{C}^{[\mathbf{b}]\mathbf{T}})^T \diamond (vec(\mathbf{A}_{I_1,:}) \otimes \mathbf{B}) \end{bmatrix} vec(\mathbf{S}) \tag{1.16}$$

$$vec(\mathcal{X}) = (\mathbf{I} \otimes (((\mathbf{B} \diamond \mathbf{C}^{[b]}) \cdot \mathbf{S}^T) \odot \mathbf{C}^{[a]}))vec(\mathbf{A}) \tag{1.17}$$

$$vec(\mathcal{X}) = (\mathbf{I} \otimes (((\mathbf{A} \diamond \mathbf{C}^{[a]}) \cdot \mathbf{S}) \odot \mathbf{C}^{[b]}))vec(\mathbf{B}) \tag{1.18}$$

The PARATUCK2 model in vectorized form possible to simplify with the Hadamard and the Khatri-Rao product notation in case if **A** and **B** have only one dimension (1.2). The definition is provided via vectorization over the resulting tensor.The PARATUCK2 model become array in the third dimension. The PARATUCK2 with one dimensioned **a** and **b** model is explained in the matrix notation with two different equation (1.19)[AFX13]. The difference between equations is order of the multiplication between matrices. The Khatri-Rao and the Hadamard product is sensible to the multiplication order and change of the multiplication order, change overall model construction. A transpose of the **X** also change the matrix multiplication order.

$$vec(\mathcal{X})^T = \mathbf{a} \cdot (\mathbf{C}^{[a]T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \tag{1.19}$$

$$\mathbf{C}^{[a]} \in \mathbb{C}^{T \times F} \mathbf{C}^{[b]} \in \mathbb{C}^{T \times T_s} \mathbf{S} \in \mathbb{C}^{F \times T_s}$$

The explained PARATUCK2 model will be used in the further work to define the GFDM system and allow to calculate solutions in the more natural form.

1.2 Generalized frequency division multiplexing systems

The generalized frequency division modulation (GFDM) is the new approach to increase the spectral efficiency of the data transmission systems[FKB09] [FMD16]. The GFDM is extended version of the orthogonal frequency division modulation system with the additional overlapping in the frequency domain. The main principle of the GFDM system is transmission of the symbols during one transmission block on the number of sub-carriers and in the number of the time slots. Each symbol corresponding to the certain time slot is modulated with time domain filter. Time domain filter provides certain envelope in the time domain. The GFDM system use special form of the time domain filters which has the overlap with the symbols on the nearest sub-carriers. This approach provide more dense symbol arrangement in the frequency domain[DMLF12].

1.2.1 Time filter construction principles

In the GFDM system are used two main filters: raised cosine and root-raised cosine filter instead of the *sinc* pulse-shaping filter in the OFDM system. Both of the filters have the variable coefficient α which allow to adjust the overlapping in the frequency domain from 0 to 1 where when $\alpha = 0$ the overlap is minimal and when $\alpha = 1$ the overlap is maximal[AMDG13][MMM+15]. That variable increase significantly inter-symbol interference and inter-channel interference and allow pitting symbols more dense than in the OFDM system. The additional price for efficient frequency usage is receiver complexity. The receiver must separate all time symbols inside the one transmission block, even if the symbols is overlapped.

The raised cosine filter envelope is provided in the (1.20) below from the [Gar05] paper. In the (1.20) T is the symbol duration in the time samples, and α is $roll-off$ factor, which is described above[MMF14]. The envelope of the raised cosine filter with different $roll-off$ factors is presented in the 1.1. The following frequency envelope is presented in the 1.2.

$$h(t) \quad = \quad \begin{cases} \frac{\pi sinc(\frac{1}{2\alpha})}{4} & t = \pm\frac{T}{2\alpha} \\ sinc(1/T)\frac{cos(\pi\alpha t/T)}{1-(2\alpha t/T)^2} & otherwise \end{cases} \tag{1.20}$$

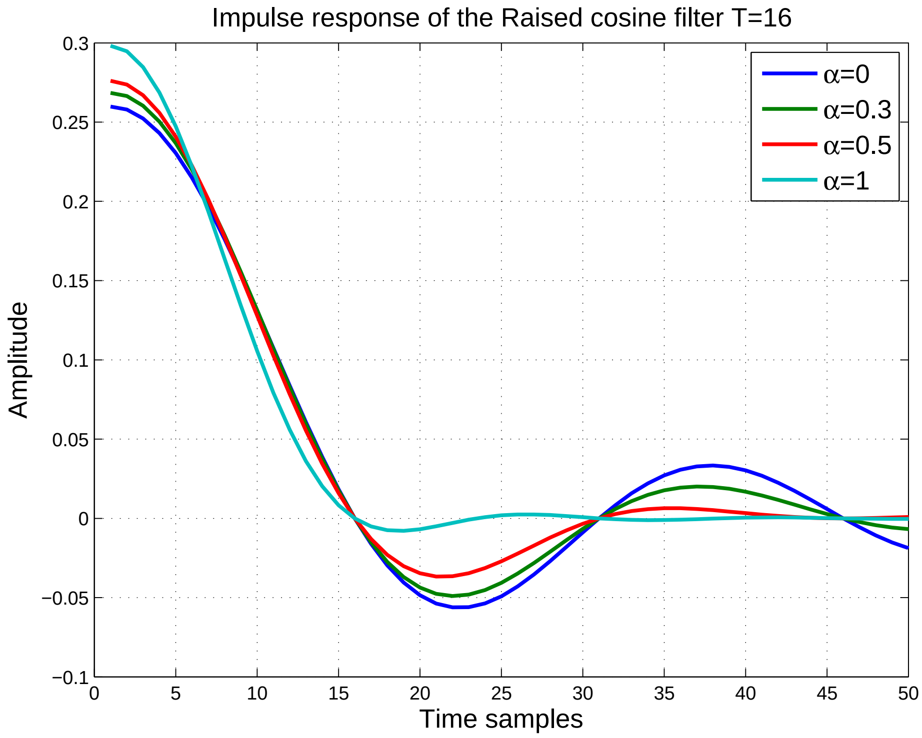

Figure 1.1: *Impulse response of the raised cosine filter with different roll-off factor* α

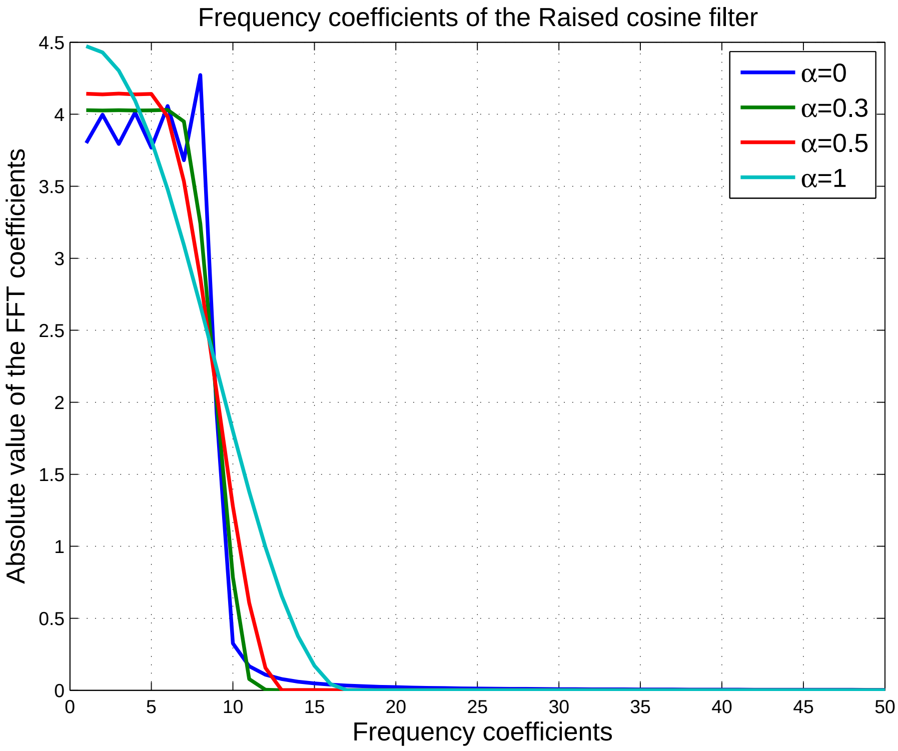

Figure 1.2: *Frequency envelope of the raised cosine filter with different roll-off factor* α

The root-raised cosine filter equation is provided in the (1.21) and given in the [MMI14] paper. In the (1.21) T is the symbol duration in the time samples, and α is $roll - off$ factor, which was described above.The envelope of the root-raised cosine filter with different $roll - off$ factors is presented in the 1.3. The following frequency envelope is presented in the 1.4.

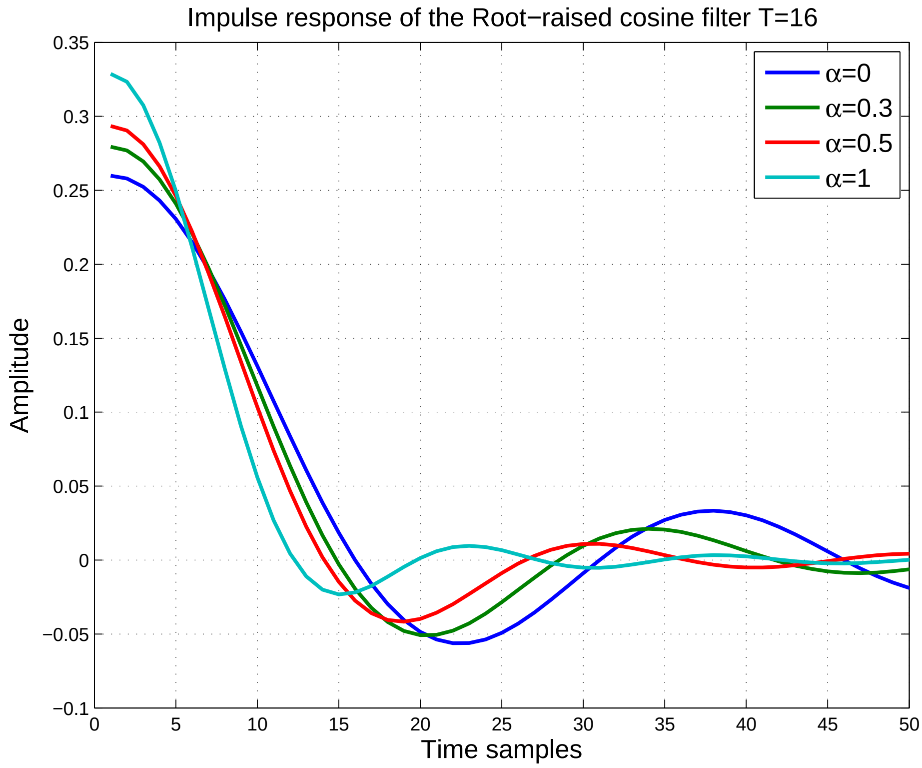

Figure 1.3: *Impulse response of the root-raised cosine filter with different roll-off factor* α

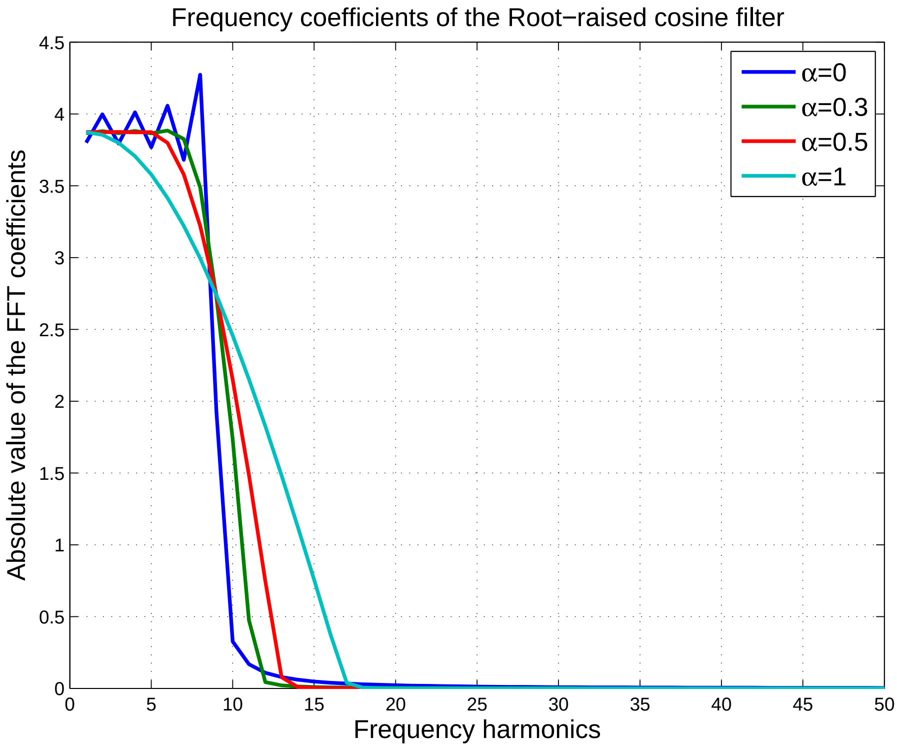

Figure 1.4: *Frequency envelope of the root-raised cosine filter with different roll-off factor α*

$$h(t) = \begin{cases} \frac{1-\alpha+4\alpha/\pi}{\sqrt{T}} & if \quad t=0 \\ \frac{\alpha}{\sqrt{2T}}[(1+\frac{2}{\pi})sin(\frac{\pi}{4\alpha}) + (1-\frac{2}{\pi})cos(\frac{\pi}{4\alpha})] & if \quad t=\pm T/4\alpha \\ \frac{1}{\frac{t\pi}{\sqrt{T}}(1-\frac{4*\alpha t}{T})^2}(sin(\frac{\pi t(1-\alpha)}{T}) + \frac{4\alpha t}{T}cos(\frac{\pi t(1+\alpha)}{T})) & otherwise \end{cases} \tag{1.21}$$

The comparison between the *sinc*, raised cosine and root-raised cosine filter is shown in the fig. 1.5. As we can see the overlap in the frequency domain spread energy from the symbol over the higher frequency range than for the *sinc* filter 1.6.

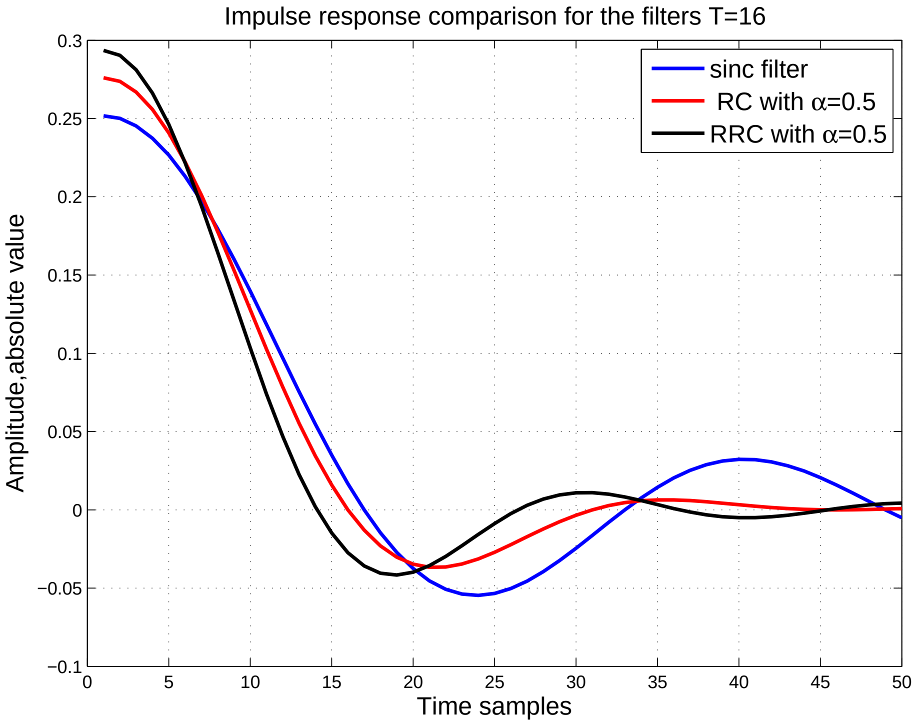

Figure 1.5: *Comparison between raised cosine root-raised cosine and sinc filter in the time domain*

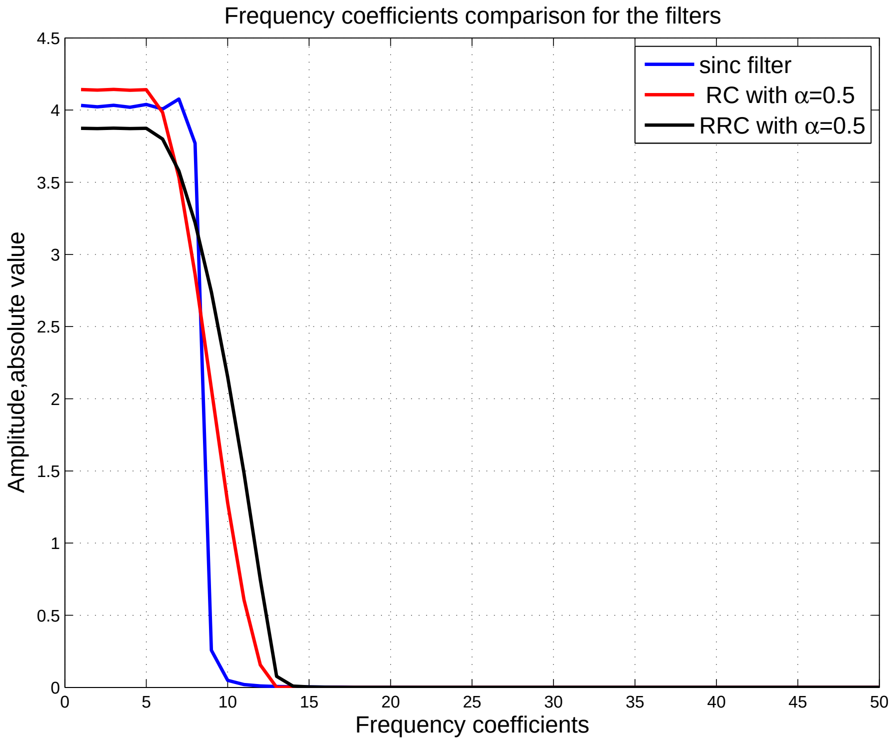

Figure 1.6: *Comparison between raised cosine root-raised cosine and sinc filter in the frequency domain*

2 Generalized frequency division multiplexing single input single output

2.1 System model

In this subsection will be explained the system model for the GFDM . We consider the system where transmitted signal is sent through the channel. The channel in physical meaning is the number of multipath propagation components from transmitter to the receiver or the convolution of the transmitted signal with some channel profile. We explain the received signal as matrix product between convolution matrix and the transmitted data array (2.1), where $\mathbf{y}$ is received array, $\mathbf{x}$ is transmitted array and $\mathbf{H}$ is channel convolution matrix. The different channel models are considered in the thesis(2.2)(2.4)[MMM+15].

$$\mathbf{y} = \mathbf{H}\mathbf{x} \tag{2.1}$$

$$\mathbf{y}, \mathbf{x} \in \mathbb{C}^{T \times 1} \mathbf{H} \in \mathbb{C}^{T \times T}$$

$$\mathbf{x} = \mathcal{X}_{[3]} = vec(\mathcal{X}) = \mathbf{\Omega}_1 vec(\mathbf{S}) \tag{2.2}$$

$$\mathbf{\Omega}_1 \in \mathbb{C}^{T \times T_s \cdot F}$$

$$\mathbf{\Omega}_1 = (\mathbf{C}^{[b]T} \diamond \mathbf{C}^{[a]T})^T \diamond (\mathbf{b}^{\mathbf{T}} \otimes \mathbf{a}) \tag{2.3}$$

$$\mathcal{X}_{:,:,i} = \mathbf{a} \cdot diag(\mathbf{C}^{[a]}_{:,i}) \cdot \mathbf{S} \cdot diag(\mathbf{C}^{[b]}_{:,i}) \cdot \mathbf{b} \tag{2.4}$$

The transmitted data is the third order PARATUCK2 tensor in the vectorized or unfolding form. The symbol matrix $\mathbf{S}$ introduced as the core matrix in the PARATUCK2 model[KB09]. The symbols are written in the core matrix without any correction. Each column in the $\mathbf{C}^{[\mathbf{a}]}$ matrix corresponds to the certain sub-carrier frequency for the stream modulation(2.5)(2.6).The $\mathbf{C}^{[\mathbf{a}]}$ matrix is constructed as the sub-carrier frequency time samples in the each column, where column i will correspond to the i sub-carrier in the GFDM system. The overall equation to the $\mathbf{C}^{[\mathbf{a}]}$ construction is explained in the (2.1).

$$c^{[a]}_{t,f} = e^{-j2\pi\cdot(t\cdot f)/T_s} \tag{2.5}$$

$$\mathbf{C}^{[a]} = \begin{bmatrix} e^{-j2\pi\cdot 0} & e^{-j2\pi\cdot 0} & \dots & e^{-j2\pi\cdot 0} \\ e^{-j2\pi\cdot 0} & e^{-j2\pi\cdot 1/T_s} & \dots & e^{-j2\pi\cdot F/T_s} \\ e^{-j2\pi\cdot 0} & e^{-j2\pi\cdot 2/T_s} & \dots & e^{-j2\pi\cdot 2F/T_s} \\ & & \vdots & \\ e^{-j2\pi\cdot 0} & e^{-j2\pi\cdot T/T_s} & \dots & e^{-j2\pi\cdot T\cdot F/T_s} \end{bmatrix} \tag{2.6}$$

Each column in the $\mathbf{C}^{[\mathbf{b}]}$ define spread of the certain symbol over the all transmission block (2.7)[BSG97]. Elements in the columns are the time filter functions for the symbols. Each column is the shifted version of the other. The shift between near columns is equal to the T/T_s (2.8). The time filters functions are the root-raised cosine filter with adjustable α coefficient.

$$u(t) = \begin{cases} \frac{1-\alpha+4\alpha/\pi}{\sqrt{T}} & if \quad t=0 \\ \frac{\alpha}{\sqrt{2T}}[(1+\frac{2}{\pi})sin(\frac{\pi}{4\alpha})+(1-\frac{2}{\pi})cos(\frac{\pi}{4\alpha})] & if \quad t=\pm T/4\alpha \\ \frac{1}{\frac{t\pi}{\sqrt{T}}(1-\frac{4*\alpha t}{T})^2}(sin(\frac{\pi t(1-\alpha)}{T})+\frac{4\alpha t}{T}cos(\frac{\pi t(1+\alpha)}{T})) & otherwise \end{cases} \tag{2.7}$$

$$\mathbf{C}^{[b]} = \begin{bmatrix} \mathbf{u}_0 & \mathbf{u}_1 & \cdots & \mathbf{u}_{T_s} \end{bmatrix} \tag{2.8}$$

The **B** matrix become **b** array and in the physical meaning corresponds to the time variation inside the one transmission block for different symbols or as selection coefficients for the different time slots[MMI14]. We assume that system is linear time invariant and therefore **b** become array with all-ones structure (2.9).

$$\mathbf{b} = \mathbf{1}_{T_s \times 1} \tag{2.9}$$

$$\mathbf{b} \in \mathbb{R}^{T_s \times 1} \tag{2.10}$$

The **A** matrix become **a** array and in the physical meaning corresponds to the transmission coefficient for each sub-carrier frequency stream or selection array for sub-carriers inside of the one transmission block (2.11). We will consider the system where the **a** will have two typical structures. First is the array with all-ones in case if there is no overlapping in the frequency domain systems and known to the receiver. In the second case the values in the **a** is unknown for the receiver and have random variables from the range $[0, 1]$

$$\mathbf{a} = \mathbf{1}_{1 \times F} \tag{2.11}$$

$$\mathbf{a} \in \mathbb{C}^{1 \times F} \tag{2.12}$$

The transmitted data is defined in equation (2.13) where each matrix is explained above. The PARATUCK2 model can be defined in the different equations (2.14) (2.15) with vectorization operation for the resulting tensor. The vectorized model is sensible to transpose of the resulting matrix and to the order change.

$$\mathbf{x}^T = \mathbf{a} \cdot (\mathbf{C}^{[a]T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \tag{2.13}$$

$$\mathbf{x} = ((\mathbf{a} \diamond \mathbf{C}^{[a]})^T \cdot \mathbf{S}) \odot \mathbf{C}^{[b]}\mathbf{b}) \tag{2.14}$$

$$\mathbf{x} = ((\mathbf{b}^T \diamond \mathbf{C}^{[b]})^T \cdot \mathbf{S}^T) \odot \mathbf{C}^{[a]T}\mathbf{a}^T) \tag{2.15}$$

$$\mathbf{x} \in \mathbb{C}^{T\times 1}\mathbf{a} \in \mathbb{C}^{1\times F}\mathbf{C}^{[a]} \in \mathbb{C}^{T\times F}$$

$$\mathbf{C}^{[b]} \in \mathbb{C}^{T\times T_s}\mathbf{b} \in \mathbb{C}^{T_s\times 1}\mathcal{S} \in \mathbb{C}^{F\times T_s}$$

2.2 Additive white Gaussian noise channel

In this section we assume the GFDM transmitter-receiver pair in the SISO case with the additive white Gaussian noise at the receiver and memoryless flat-frequency channel[AMDG13]. Therefore system model at the receiver become equation (2.16).

$$\mathbf{y} = \mathbf{\Omega}_1 vec(\mathbf{S}) + \mathbf{n} \tag{2.16}$$

$$\mathbf{\Omega}_1 \in \mathbb{C}^{T\times T_s\cdot F}$$

$$\mathbf{\Omega}_1 = (\mathbf{C}^{[b]T} \diamond \mathbf{C}^{[a]T})^T \diamond (\mathbf{b} \otimes \mathbf{a}) \tag{2.17}$$

$$\mathbf{y}, \mathbf{n} \in \mathbb{C}^{1\times T}\mathbf{S} \in \mathbb{C}^{F\times T_s}$$

$$\mathbf{a} \in \mathbb{C}^{1\times F}\mathbf{C}^{[a]} \in \mathbb{C}^{T\times F}\mathbf{C}^{[b]} \in \mathbb{C}^{T\times T_s}\mathbf{b} \in \mathbb{C}^{T_s\times 1}$$

2.2.1 Zero forcing solution

The receiver can find the optimal transmitted values via optimization task solution[BSG99]. The demodulation problem can be rewritten in the residual form. We can rewrite $\mathbf{y}$ as the product of the known modulation matrix with the vectorized symbols matrix (2.18). We introduce the matrix $\mathbf{\Omega}_1$ in (2.19) and define $vec(\mathbf{y})$ as the matrix product between the $\mathbf{\Omega}_1$ and estimated $vec(\hat{\mathbf{S}})$ (2.20). We define residual equation which is difference between received data and the estimated at the receiver symbols (2.21). The $\boldsymbol{\Omega}_1$ is given from the paper [AFX13]. If the $\mathbf{a}$ and $\mathbf{b}$ have all-ones structure we can neglect them and take as the Khatri-Rao product between $\mathbf{C}^{[\mathbf{a}]}$ and $\mathbf{C}^{[\mathbf{b}]}$ (2.21)

$$vec(\mathbf{x}) = \boldsymbol{\Omega}_1 \cdot vec(\hat{\mathbf{S}}) \tag{2.18}$$

$$\boldsymbol{\Omega}_1 \in \mathbb{C}^{T \times T_s \cdot F}$$

$$vec(\hat{\mathbf{S}}) \in \mathbb{C}^{F \cdot T_s \times 1}$$

$$\boldsymbol{\Omega}_1 = (\mathbf{C}^{[b]T} \diamond \mathbf{C}^{[a]T})^T \diamond (\mathbf{b} \otimes \mathbf{a}) \tag{2.19}$$

$$\boldsymbol{\Omega}_1 = (\mathbf{C}^{[b]T} \diamond \mathbf{C}^{[a]T})^T \tag{2.20}$$

$$r_1 = \mathbf{y} - \boldsymbol{\Omega}_1 \cdot vec(\hat{\mathbf{S}}) \tag{2.21}$$

The receiver must solve the system of linear equations. The problem is solved in the second norm sense (2.22) and also known as the Zero-Forcing receiver [DMLF12].The second norm function is convex, with one global minimum point. To find minimum point the receiver find partial derivative of the objective function (2.23), equate it to the zero and solve the system of linear equations. The estimated $\mathbf{S}$ matrix is complex valued, which makes objective function non-holomorphic. To solve optimization problem we have used the Wirtinger calculus and find the partial derivative with respect to the $vec(\hat{\mathbf{S}}^*)$ and equate it to the zero. Next calculations step come to the classical least squares solution of the objective function (2.27)[DMLF12].

$$\min_{vec(\hat{\mathbf{S}})} r_1^H \cdot r_1 = \min_{vec(\hat{\mathbf{S}})} || \, r_1 \, ||^2 \tag{2.22}$$

$$r_1^H \cdot r_1 = (vec(\mathbf{y}) - \mathbf{\Omega}_1 \cdot vec(\hat{\mathbf{s}}))^H \cdot (vec(\mathbf{y}) - \mathbf{\Omega}_1 \cdot vec(\hat{\mathbf{s}})) \tag{2.23}$$

$$r_1^H \cdot r_1 = || \; vec(\mathbf{y}) \; ||^2 + vec(\hat{\mathbf{s}})^H \mathbf{\Omega}_1^H \mathbf{\Omega}_1 vec(\hat{\mathbf{s}})$$

$$-vec(\mathbf{y})^H \mathbf{\Omega}_1 vec(\hat{\mathbf{s}}) - vec(\hat{\mathbf{s}})^H \mathbf{\Omega}_1^H vec(\mathbf{y}) \tag{2.24}$$

$$\frac{\delta r_1^H \cdot r_1}{\delta vec(\hat{\mathbf{s}}^*)} = \mathbf{\Omega}_1^H \mathbf{\Omega}_1 vec(\hat{\mathbf{s}}) - \mathbf{\Omega}_1^H vec(\mathbf{y}) = 0 \tag{2.25}$$

$$\mathbf{\Omega}_1^H \mathbf{\Omega}_1 vec(\hat{\mathbf{s}}) = \mathbf{\Omega}_1^H vec(\mathbf{y}) \tag{2.26}$$

$$vec(\hat{\mathbf{s}})_{opt} = (\mathbf{\Omega}_1^H \mathbf{\Omega}_1)^{-1} \mathbf{\Omega}_1^H vec(\mathbf{y}) \tag{2.27}$$

The main equation to symbol estimation is the (2.27) which we use in the receiver. We simply multiply received vector with the predefined at the receiver matrix.

2.2.2 Matched filter solution

Another solution for the problem is so called Matched filter receiver. The receiver multiply received data with the Hermitian transpose of the modulation matrix[FKB09]. We can look at the singular value decomposition of the $\mathbf{\Omega}_1$ (2.28) and write in the same form the inverse of the matrix due to the unitary structure of the matrix $\mathbf{U}$ (2.29). The inverse of the $\mathbf{\Sigma}$ is inverse of the each element of the matrix due to the diagonal structure[FKB09]. With increasing of the α coefficient in the time filter the condition number of the $\mathbf{\Omega}_1$(2.31) decrease and inverse of the matrix will increase noise components with decreasing minimal singular value of the $\mathbf{\Omega}_1$(2.31). We can multiply the received data with Hermitian of the $\mathbf{\Omega}_1$(2.32)[DMLF12]. In that case the structure of the singular components inside of the calculated matrix become the same and noise components will have the same power(2.33).

$$\mathbf{\Omega}_1 = \mathbf{U\Sigma V^H} \tag{2.28}$$

$$\mathbf{\Omega}_1^{-1} = \mathbf{V\Sigma^{-1}U^H} \tag{2.29}$$

$$\mathbf{n} = \mathbf{UI}\delta^{\mathbf{2}}\mathbf{V^H} \tag{2.30}$$

$$E(\mathbf{\Omega}_1^{-1}\mathbf{n}) = \mathbf{V}(\mathbf{\Sigma^{-1}} \cdot \mathbf{I}\delta^{\mathbf{2}})\mathbf{V^H} \tag{2.31}$$

$$\mathbf{\Omega}_1^H = \mathbf{V\Sigma U^H} \tag{2.32}$$

$$E(\mathbf{\Omega}_1^H\mathbf{n}) = \mathbf{V}(\mathbf{\Sigma} \cdot \mathbf{I}\delta^{\mathbf{2}})\mathbf{V^H} \tag{2.33}$$

$$vec(\mathbf{S})_{opt} = \mathbf{\Omega}_1^H(\mathbf{y}) \tag{2.34}$$

The main equation to symbol estimation is the (2.34) which we use in the receiver. We simply multiply received vector with the predefined at the receiver matrix.

2.3 Selection coefficients search

Explained above the coefficients from the **a** are used as the selection coefficients for the sub-carriers. In the GFDM system is allowed to disable some sub-carriers to decrease interference with other systems which work in the same frequency range[FKB09].The transmitted may use the spectrum sensing approach and doesn't use the number of subcarriers inside of the transmission block. Receiver will not know which sub-carriers are used and must estimate information from the received data(2.35). The information about used sub-carriers may transmitted as additional to find coefficients at the receiver side. We can use the PARATUCK2 model to estimate information about used subcarriers at the receiver. The received signal is defined as (2.35), where the selection coefficients become in the $\mathbf{a}$

$$\mathbf{y} = \mathbf{\Omega}_1 vec(\mathbf{S}) + \mathbf{n} \tag{2.35}$$

We formulate equation to vectorize the $\mathbf{a}$ from the right hand side as (2.36) and write the problem statement to find the coefficients as the residual between received signal and estimated channel (2.37).

$$\mathbf{\Omega}_2 = ((\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T \cdot S^T) \odot \mathbf{C}^{[a]} \tag{2.36}$$

$$r_2 = \mathbf{y} - \mathbf{\Omega}_2 \cdot vec(\mathbf{a}) \tag{2.37}$$

2.3.1 Semi-blind receiver

The receiver estimates the transmission coefficients and the symbols in case receiver will know at least one symbol at each sub-carrier frequency. The receiver will estimate in the certain time slot the transmission coefficient for the overall block of the data and will find all other symbols according to the LS solution or optimization algorithm. We should write the objective function and minimize it to implement the semi-blind receiver. The unknown variables are $\mathbf{S}$ and the $\mathbf{a}$. The known symbols is added in the objective function as constraint or via decreasing number of the variables. Second way decreases scale of the objective function, but problem formulation become non-trivial. We add known symbol information as the additional constraint in the objective function (2.38). There are two ways to solve the task, joint and approximated solutions. The joint solution consider all dependences between r_1, r_2, r_3. The second approach calculate solution for three residual function separately and use non-linear optimization process to stack three equations in one. We implemented both of them.

2.3.2 Approximated semi-blind receiver

Problem statement is defined in (2.39). We minimize objective function with assumption: $\mathbf{r}_1$ and $\mathbf{r}_2$ does not depend from each other. We don't used the Lagrangian multipliers to analytical optimization and find minimum point in the optimization process. The residual function r_1 corresponds to the objective function with vectorized symbol matrix $\hat{\mathbf{S}}$. The $\mathbf{\Omega}_1$ is defined previously and corresponds to the modulation matrix.

$$\mathbf{r}_1 = vec(\mathbf{y}) - \mathbf{\Omega}_1 \cdot vec(\hat{\mathbf{S}})$$

The residual function r_2 corresponds to the objective function with vectorized array $\mathbf{a}$. The $\mathbf{\Omega}_2$ is defined previously and corresponds to the transmitted data with separated sub-carrier coefficients.

$$\mathbf{r}_2 = vec(\mathbf{y}) - \mathbf{\Omega}_2 \cdot vec(\hat{\mathbf{a}})$$

The residual function r_3 corresponds to the constraint for the known symbols, where $\mathbf{S}_{sel}$ become the selection matrix, where on the main diagonal is ones in points where $vec(\hat{\mathbf{S}})$ is known. The $\mathbf{q}$ array is the array with vectorized known symbols in the $vec(\hat{\mathbf{S}})$ and zeros in the unknown points.

$$\mathbf{r}_3 = \mathbf{q} - \mathbf{S}_{sel} vec(\hat{\mathbf{S}}); \tag{2.38}$$

$$\mathbf{S}_{sel} = diag(\mathbf{q})^{-1} diag(\mathbf{q})$$

$$\mathbf{q} \in \mathbb{C}^{T_s \cdot F \times 1} \mathbf{S}_{sel} \in \mathbb{C}^{T_s \cdot F \times T_s \cdot F}$$

$$\mathbf{r}_1 \in \mathbb{C}^{T_s \cdot F \times 1} \mathbf{r}_2 \in \mathbb{C}^{F \times 1} \mathbf{r}_3 \in \mathbb{C}^{T_s \cdot F \times 1}$$

$$\min_{vec(\hat{\mathbf{S}})} r_1^H r_1 \tag{2.39}$$

$$\min_{vec(\mathbf{a})} r_2^H r_2 \tag{2.40}$$

$$\min_{vec(\rlap{\mathbf{b}}\mathbf{S})} r_3^H r_3 \tag{2.41}$$

$$\begin{bmatrix} \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\rlap{\mathbf{b}}\mathbf{S})^*} \\ \frac{\delta \mathbf{r}_2^H \mathbf{r}_2}{\delta vec(\rlap{\mathbf{b}}\mathbf{a})^*} \\ \frac{\delta \mathbf{r}_3^H \mathbf{r}_3}{\delta vec(\rlap{\mathbf{b}}\mathbf{S})^*} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{2.42}$$

$$\begin{bmatrix} \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\rlap{\mathbf{b}}\mathbf{S})^*} \\ \frac{\delta \mathbf{r}_2^H \mathbf{r}_2}{\delta vec(\rlap{\mathbf{b}}\mathbf{S})^*} \\ \frac{\delta \mathbf{r}_3^H \mathbf{r}_3}{\delta vec(\rlap{\mathbf{b}}\mathbf{S})^*} \end{bmatrix} = \begin{bmatrix} -\mathbf{\Omega}_1^H \mathbf{r}_1 \\ -\mathbf{\Omega}_2^H \mathbf{r}_2 \\ -\mathbf{S}_{sel}^H \mathbf{r}_3 \end{bmatrix} \tag{2.43}$$

$$\frac{\delta r_1^H r_1}{\delta vec(\rlap{\mathbf{b}}\mathbf{S}^*)} = \mathbf{\Omega}_1^H \mathbf{\Omega}_1 vec(\rlap{\mathbf{b}}\mathbf{S}) - \mathbf{\Omega}_1^H vec(\mathbf{y}) = 0 \tag{2.44}$$

$$\frac{\delta r_2^H r_2}{\delta vec(\rlap{\mathbf{b}}\mathbf{a}^*)} = \mathbf{\Omega}_2^H \mathbf{\Omega}_2 vec(\rlap{\mathbf{b}}\mathbf{a}) - \mathbf{\Omega}_2^H vec(\mathbf{y}) = 0 \tag{2.45}$$

$$\frac{\delta r_3^H r_3}{\delta vec(\rlap{\mathbf{b}}\mathbf{S}^*)} = \mathbf{S}_{sel}^H \mathbf{S}_{sel} vec(\rlap{\mathbf{b}}\mathbf{S}) - \mathbf{S}_{sel}^H vec(\mathbf{q}) = 0 \tag{2.46}$$

$$\mathbf{S}_{sel}^H \mathbf{S}_{sel} = \mathbf{S}_{sel} \tag{2.47}$$

We can stack (2.47)(2.46)(2.45) in one system of non-linear equations.There are many ways to sole the system of non-linear equations. We considered two ways to solve the presented system of non-linear equations:

- ALS algorithm[Fle00]
- Newton method[Kel03]

The Newton method solution for system of non-linear equation includes two steps[Deu04]:

- Solve the system of the linear equations in the given point $[vec(\mathbf{S})^n, vec(\mathbf{a})^n]$ (2.48).
- Renew the given point $[vec(\mathbf{S})^n, vec(\mathbf{a})^n]$ with solution in the step 1 (2.50).
- Check is the objective function value less than tolerance. If objective function value higher than tolerance repeat process again.

There are two additional approaches, which could be implemented in the Newton algorithm. One is the Powell-Wolf rule[DES82][DS83] for the step-size variable adjust. The second is the Levenberg-Marquardt regularization[Fle00][TA79] algorithm. The main equation of the Newton algorithm for the non-linear equations is presented in the (2.48), where $\mathbf{d}$ is the derivative array which is minimized to the zero and J is the Jacobian matrix of the $\mathbf{d}$ array. The definition of the partial derivative array $\mathbf{d}$ array is explained in the equation(2.51).

$$\mathbf{J}\delta\theta = -\mathbf{d} \tag{2.48}$$

$$\delta\theta = -\mathbf{J}^{+}\mathbf{d} \tag{2.49}$$

$$\theta^{k+1} = \theta^{k} + \alpha\delta\theta \tag{2.50}$$

$$\alpha = (0; 1]$$

$$\mathbf{d} = \begin{bmatrix} \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\mathbf{S})^*} \\ \frac{\delta \mathbf{r}_2^H \mathbf{r}_2}{\delta vec(\mathbf{a})^*} \\ \frac{\delta \mathbf{r}_3^H \mathbf{r}_3}{\delta vec(\mathbf{S})^*} \end{bmatrix} = 0 \tag{2.51}$$

We must calculate partial derivative from the $\mathbf{d}$ to find the Jacobian matrix (2.52).Because the function(2.51) is holomorphic, we don't use the Wirtinger calculus in Jacobian calculation. We calculate partial derivative with respect to the $\delta\theta$ (2.53).

$$\mathbf{J} = \frac{\delta \mathbf{d}}{\delta \theta} \tag{2.52}$$

$$\mathbf{J} = -\begin{bmatrix} \mathbf{\Omega}_1^H \mathbf{\Omega}_1 + \mathbf{S}_{sel} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Omega}_2^H \mathbf{\Omega}_2 \end{bmatrix} \tag{2.53}$$

The receiver solves the system of the linear equations at the each iteration and update the search vector for the next step(2.50). The optimization algorithm decrease the derivative of residual to the zero, and converge due to the convexity of the second norm[Ste14]. The following algorithm is defined in the next steps:

- Set the initial point θ randomly.
- Calculate the partial derivative $\mathbf{d}$(2.51) and Jacobian $\mathbf{J}$(2.53) in the given point θ
- Solve the system of linear equations (2.49)
- Renew the variable point with (2.50)
- Check the residual. If residual reduction is higher than limit, repeat process from step 2.

2.3.3 Joint semi-blind receiver

We can minimize the objective function (2.54) without assumptions and approximations. General form makes implementation more complex. It should be noticed that $\mathbf{r}_1$ and the $\mathbf{r}_2$ is the same equations but explained from the different point of view: from the unknown symbols or unknown sub-carrier coefficients.

$$\mathbf{r}_1 = vec(\mathbf{y}) - \mathbf{\Omega}_1 \cdot vec(\mathbf{S}) = \mathbf{r}_2$$

$$\mathbf{r}_2 = vec(\mathbf{y}) - \mathbf{\Omega}_2 \cdot vec(\mathbf{h})$$

$$\mathbf{r}_3 = \mathbf{q} - \mathbf{S}_{sel} vec(\mathbf{S});$$

$$vec(\mathbf{y}) - \mathbf{\Omega}_1 \cdot vec(\mathbf{S}_1) = vec(\mathbf{y}) - \mathbf{\Omega}_2 \cdot vec(\mathbf{a}_1) \tag{2.54}$$

We write the objective function. The function in non-holomorphic. We find the partial derivatives with Wirtinger calculus. We can use the (2.56) property of the derivatives to find the derivatives separately for each function. To calculate the cross derivatives for the $\mathbf{r}_1^H \mathbf{r}_1$ with respect to the $\mathbf{a}^*$ we use the (2.59) which allow to change the left hand part of equation, where $\mathbf{a}^*$ is clearly defined[AFX13].

$$\min_{\begin{bmatrix} vec(\mathbf{S}) \\ vec(\mathbf{a}) \end{bmatrix}} \mathbf{r}_1^H \mathbf{r}_1 + \mathbf{r}_3^H \mathbf{r}_3 \tag{2.55}$$

$$\mathbf{G} = \mathbf{r}_1^H \mathbf{r}_1 + \mathbf{r}_3^H \mathbf{r}_3$$

$$\frac{\delta \mathbf{G}}{\delta \begin{bmatrix} vec(\mathbf{S}^*) \\ vec(\mathbf{a}^*) \end{bmatrix}} = \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta \begin{bmatrix} vec(\mathbf{S}^*) \\ vec(\mathbf{a}^*) \end{bmatrix}} + \frac{\delta \mathbf{r}_3^H \mathbf{r}_3}{\delta \begin{bmatrix} vec(\mathbf{S}^*) \\ vec(\mathbf{a}^*) \end{bmatrix}} \tag{2.56}$$

$$\frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta \begin{bmatrix} vec(\mathbf{S}^*) \\ vec(\mathbf{a}^*) \end{bmatrix}} = \begin{bmatrix} \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\mathbf{S}^*)} \\ \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\mathbf{a}^*)} \end{bmatrix} \tag{2.57}$$

$$\frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\mathbf{S}^*)} = -\mathbf{\Omega}_1^H (vec(\mathbf{y}) - \mathbf{\Omega}_1 vec(\mathbf{S})) = -\mathbf{\Omega}_1^H \mathbf{r}_1 \tag{2.58}$$

$$\frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\mathbf{a}^*)} = -\mathbf{\Omega}_2^H (vec(\mathbf{y}) - \mathbf{\Omega}_2 vec(\mathbf{a})) = -\mathbf{\Omega}_2^H \mathbf{r}_2 \tag{2.59}$$

$$\frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\rlap{\mathbf{b}}\mathbf{a}^*)} = \mathbf{\Omega}_2^H \mathbf{\Omega}_1 vec(\rlap{\mathbf{b}}\mathbf{S}) - \mathbf{\Omega}_2^H vec(\mathbf{y}) \tag{2.60}$$

$$\frac{\delta \mathbf{r}_3^H \mathbf{r}_3}{\delta vec(\rlap{\mathbf{b}}\mathbf{S}^*)} = \mathbf{S}_{sel}^H \mathbf{S}_{sel} vec(\rlap{\mathbf{b}}\mathbf{S}) - \mathbf{S}_{sel}^H vec(\mathbf{y}) \tag{2.61}$$

$$\frac{\delta \mathbf{r}_3^H \mathbf{r}_3}{\delta vec(\rlap{\mathbf{b}}\mathbf{a}^*)} = \mathbf{0} \tag{2.62}$$

The optimization function become the system of non-linear equations. As explained previously, to solve the system of the non-linear equations we use the Newton method.The $\mathbf{d}$ is presented in the (2.67). The function is holomorphic[HJ07] and we can calculate Jacobian matrix $\mathbf{J}$ without Wirtinger calculus (2.66). The r_1 and the r_2 is equal function in the different notation. We use this property when we calculate cross variable derivatives. The resulting Jacobian matrix (2.68) is different from the Jacobian matrix presented in the approximated solution (2.68).

$$\mathbf{J}\delta\theta = -\mathbf{d} \tag{2.63}$$

$$\delta\theta = -\mathbf{J}^{+}\mathbf{d} \tag{2.64}$$

$$\theta^{k+1} = \theta^k + \alpha\delta\theta \tag{2.65}$$

$$\alpha = (0; 1]$$

$$\mathbf{J} = \begin{bmatrix} \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\rlap{\mathbf{b}}\mathbf{S}^*) vec(\rlap{\mathbf{b}}\mathbf{S})} & \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\rlap{\mathbf{b}}\mathbf{S}^*) vec(\mathbf{A})} \\ \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\rlap{\mathbf{b}}\mathbf{a}^*) vec(\rlap{\mathbf{b}}\mathbf{S})} & \frac{\delta \mathbf{r}_1^H \mathbf{r}_1}{\delta vec(\rlap{\mathbf{b}}\mathbf{a}^*) vec(\rlap{\mathbf{b}}\mathbf{a})} \end{bmatrix} = \begin{bmatrix} \frac{\delta \mathbf{d}}{\delta vec(\rlap{\mathbf{b}}\mathbf{S})} & \frac{\delta \mathbf{d}}{\delta vec(\rlap{\mathbf{b}}\mathbf{a})} \end{bmatrix} \tag{2.66}$$

$$\mathbf{d} = \begin{bmatrix} -\mathbf{\Omega}_1^H(\mathbf{r}_1) + \mathbf{S}_{sel}(\mathbf{r}_3) \\ -\mathbf{\Omega}_2^H(\mathbf{r}_2) \end{bmatrix} \tag{2.67}$$

$$\mathbf{J} = \begin{bmatrix} \mathbf{\Omega}_1^H\mathbf{\Omega}_1 + \mathbf{S}_{sel} & \mathbf{\Omega}_1^H\mathbf{\Omega}_2 \\ \mathbf{\Omega}_2^H\mathbf{\Omega}_1 & \mathbf{\Omega}_2^H\mathbf{\Omega}_2 \end{bmatrix} \tag{2.68}$$

The following algorithm is defined in the next steps:

- Set the initial point θ randomly.
- Calculate the partial derivative $\mathbf{d}$(2.67) and Jacobian $\mathbf{J}$(2.68) in the given point θ
- Solve the system of linear equations (2.64)
- Renew the variable point with (2.65)
- Check the residual. If residual reduction is higher than limit, repeat process from step 2.

Joint solution converge faster in comparison with the approximated approach. Each iteration in the joint solution algorithm become computationally more expensive, but algorithm takes much fewer iterations.

2.4 Frequency selective channel estimation

2.4.1 Approximated channel estimation

In this section is considered model where channel has the multipath propagation components (2.69)(2.70). We use assumption of maximal delay between first and the last multipath propagation components equal to the T/T_s time samples. This assumption makes approach below only approximation of the channel estimation .

$$\mathbf{y} = \mathbf{H}\mathbf{x} \tag{2.69}$$

$$\mathbf{y}, \mathbf{x} \in \mathbb{C}^{T\times 1}\mathbf{H} \in \mathbb{C}^{T\times T}$$

$$\mathbf{x} = \mathbf{\Omega}_1 vec(\mathbf{S}) \tag{2.70}$$

We can implement the channel estimation approach without channel prefix in the start of the transmission block. We put the training symbols in the transmission block. Consider the matrix $\mathbf{H}$ in terms of parametric model. The matrix $\mathbf{H}$ has Toeplitz lower triangular structure(2.71)[Gar05]. Each column in the $\mathbf{H}$ corresponds to the multipath components for certain time sample. We assume that length of the channel is less than T/T_s. The matrix $\mathbf{H}$ has only T/T_s non-zero elements in the each column. Maximal length of the cyclic prefix to estimate the channel equal to the T/T_s samples in that case.

$$\mathbf{H} = \begin{bmatrix} h_1 & 0 & 0 & \cdots & 0 \\ h_2 & h_1 & 0 & \cdots & 0 \\ h_3 & h_2 & h_1 & \cdots & 0 \\ \vdots & & & & \\ h_T & h_{T-1} & h_1 & \cdots & h_1 \end{bmatrix} \tag{2.71}$$

$$h_i = 0 \; if \; i > T/T_s \tag{2.72}$$

The transmitter inserts this prefix as predefined symbols in the first column in the matrix $\mathbf{S}$ (2.73). The first column in the matrix $\mathbf{S}$ corresponds to the first time slot for each sub-carrier stream(2.74). The knowledge of the symbols in the first column doesn't define the first T/T_s time samples in the transmission block. The receiver doesn't know exactly the first T/T_s samples of received data, even if the first F symbols is known. Due to that fact the receiver will approximate the channel with the error [VJ14]. Define the matrix with training symbols(2.73).Define the matrix with unknown symbols(2.74). The training signal from the transmitter will be written in the following form . The received signal will be written as following(2.76). The receiver can construct the following array from the known training symbols(2.78) and define the convolution matrix from the known array.

$$\mathbf{S}_{rec} = \begin{bmatrix} s_{1,1} & 0 & 0 & \cdots & 0 \\ s_{2,1} & 0 & 0 & \cdots & 0 \\ \vdots \\ s_{F,1} & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{2.73}$$

$$\mathbf{S}_{tr} = \begin{bmatrix} 0 & s_{1,2} & s_{1,3} & \cdots & s_{1,T_s} \\ 0 & s_{2,2} & s_{2,3} & \cdots & s_{2,T_s} \\ \vdots \\ 0 & s_{F,2} & s_{F,3} & \cdots & s_{F,T_s} \end{bmatrix} \tag{2.74}$$

$$\mathbf{x}_{rec} = \mathbf{\Omega}_1 vec(\mathbf{S}_{rec}) \tag{2.75}$$

$$\mathbf{y} = \mathbf{H}\mathbf{\Omega}_1(vec(\mathbf{S}_{tr}) + vec(\mathbf{S}_{rec})) + \mathbf{n} \tag{2.76}$$

$$\mathbf{x}_{rec} = \mathbf{\Omega}_1 vec(\mathbf{S}_{rec}) = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_T \end{bmatrix}^T \tag{2.77}$$

$$\mathbf{X}_{cor} = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ x_2 & x_1 & 0 & \cdots & 0 \\ \vdots \\ x_T & x_{T-1} & x_{T-2} & \cdots & x_1 \end{bmatrix} \tag{2.78}$$

$$\mathbf{h}_{appx} = \mathbf{X}_{cor}^{*} \cdot \mathbf{H}\mathbf{y} \tag{2.79}$$

One approach to decrease channel estimation error is the adaptive α coefficient adjustment. The transmitter uses two different α coefficients for the working time slots and for the test time slots. We use guard interval with length one symbol to avoid ICI and decrease interference for the certain symbols. Symbols with decreased ICI used as the cyclic prefix and increase estimation precision for the described above approach. The second approach to decrease uncertainty and increase precision is optimization algorithm. Algorithm estimate use additional double side interference cancellation. The joint symbol and multipath components estimation leads to the decrease of uncertainty with symbol estimation, and decrease probability of the error with the multipath component estimation.

2.4.2 Channel estimation

The main equation of the received data (2.81) can be rewritten with respect to the unknown channel model (2.80).

$$\mathbf{y} = \mathbf{D}_{mod}\mathbf{h} + \mathbf{n} \tag{2.80}$$

$$\mathbf{y} = \mathbf{H}\mathbf{\Omega}_1 vec(\mathbf{S}) + \mathbf{n} \tag{2.81}$$

$$\mathbf{x} = \mathbf{\Omega}_1 vec(\mathbf{S}) = \begin{bmatrix} x_1 & x_2 & \cdots & x_T \end{bmatrix}^T \tag{2.82}$$

$$\mathbf{D}_{mod} = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ x_2 & x_1 & 0 & \cdots & 0 \\ \vdots & & & & \\ x_T & x_{T-1} & x_{T-2} & \cdots & x_{2+L} \end{bmatrix} \tag{2.83}$$

$$\mathbf{h} \in \mathbb{C}^{L+1\times 1} \mathbf{D}_{mod} \in \mathbb{C}^{T\times L+1}$$

$$\mathbf{r}_s = \mathbf{y} - \mathbf{D}_{mod}\mathbf{h} \tag{2.84}$$

Where the $\mathbf{h}$ is the channel coefficients with known length $L + 1$. The matrix $\mathbf{D}_{mod}$ is constructed via shifting and stacking of the transposed array $\mathbf{\Omega}_1 vec(\mathbf{S})$[LV06]. As shown in the (2.85) the receiver can estimate the channel is case if symbols are known[PP12]. It should be noted that transmitter must know the length of the channel . The least squares solution is calculated as following in the previous parts with the Wirtinger calculus (2.86). The receiver calculates the pseudo-inverse of the matrix (2.88). The solution for the least squares problem is presented in the (2.88).

$$\min_{\mathbf{h}} \mid\mid \mathbf{y} - \mathbf{D}_{mod}\mathbf{h} \mid\mid^2 = \min_{\mathbf{h}} \mathbf{r}_s^H \mathbf{r}_s \tag{2.85}$$

$$\frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta \mathbf{h}^*} = -\mathbf{D}_{mod}(\mathbf{y} - \mathbf{D}_{mod}^H \mathbf{h}) = 0 \tag{2.86}$$

$$\mathbf{h} = (\mathbf{D}_{mod}^H \mathbf{D}_{mod})^{-1} \mathbf{D}_{mod}^H \mathbf{y} \tag{2.87}$$

$$\mathbf{h}_{opt} = \mathbf{D}_{mod}^{+} \mathbf{y} \tag{2.88}$$

In the practical case, the transmission block is too large to them all with known symbols. The transmitter can fill number of the symbols with the known data and other as unknown to the receiver. The matrices $\mathbf{S}_{sel}$ and $\mathbf{S}_{selunk}$ are constructed in the similar way. The matrices are the selection matrices with ones in the points where symbol transmitted with certain column of the $\mathbf{\Omega}_1$.

$$\mathbf{x}_{rec} = \mathbf{\Omega}_1 vec(\mathbf{S}) = \mathbf{\Omega}_1 \mathbf{S}_{sel} vec(\mathbf{S}_{kn}) + \mathbf{\Omega}_1 (\mathbf{S}_{selunk}) vec(\mathbf{S}_{unk}) \tag{2.89}$$

$$vec(\mathbf{S}_{sel}) \in \mathbb{C}^{T \times Z} vec(\mathbf{S}_{selunk}) \in \mathbb{C}^{T \times T_s F - Z}$$

$$vec(\mathbf{S}) = \mathbf{S}_{sel} vec(\mathbf{S}_{kn}) + (\mathbf{S}_{selunk}) vec(\mathbf{S}_{unk}) \tag{2.90}$$

$$\mathbf{\Omega}_1 \mathbf{S}_{sel} = \mathbf{\Omega}_k \tag{2.91}$$

$$\mathbf{\Omega}_1(\mathbf{S}_{selunk}) = \mathbf{\Omega}_u \tag{2.92}$$

$$vec(\mathbf{S}_{kn}) \in \mathbb{C}^{Z\times 1} vec(\mathbf{S}_{unk}) \in \mathbb{C}^{FT_s - Z\times 1} \tag{2.93}$$

$$\mathbf{\Omega}_u \in \mathbb{C}^{T\times FT_s - Z} \mathbf{\Omega}_k \in \mathbb{C}^{T\times Z} \tag{2.94}$$

The receiver must find both symbols and the channel values and perform the semi-blind receiver solution. To make the statement of the task separated from the known and unknown symbol part, which also decrease inter-symbol interference the receiver rewrite the task in the following notation (2.101).

$$\mathbf{y} = \mathbf{H}\mathbf{\Omega}_k vec(\mathbf{S}_{kn}) + \mathbf{H}\mathbf{\Omega}_u vec(\mathbf{S}_{unk}) + \mathbf{n} \tag{2.95}$$

$$\mathbf{r}_s = \mathbf{y} - \mathbf{H}\mathbf{\Omega}_k vec(\mathbf{S}_{kn}) - \mathbf{H}\mathbf{\Omega}_u vec(\mathbf{S}_{unk}) \tag{2.96}$$

$$\mathbf{D}_{mod} = \mathbf{D}_k + \mathbf{D}_u \tag{2.97}$$

$$\mathbf{H}\mathbf{\Omega}_k vec(\mathbf{S}_{kn}) = \begin{bmatrix} kn_1 & kn_2 & \cdots & kn_T \end{bmatrix}^T \tag{2.98}$$

$$\mathbf{D}_k = \begin{bmatrix} kn_1 & 0 & 0\cdots & 0 \\ kn_2 & kn_1 & 0\cdots & 0 \\ \vdots & & & \\ kn_T & kn_{T-1} kn_{T-2}\cdots & kn_1 & \end{bmatrix} \tag{2.99}$$

$$\mathbf{y} = \mathbf{D}_k\mathbf{h} + \mathbf{D}_u\mathbf{h} + \mathbf{n} \tag{2.100}$$

$$\mathbf{r}_s = \mathbf{y} - \mathbf{D}_k\mathbf{h} - \mathbf{D}_u\mathbf{h} \tag{2.101}$$

We write the received data as the sum of two array: known and unknown. The receiver also can divide the modulation matrix into the two parts: with columns corresponding to the known symbols and with other columns of the matrix $\boldsymbol{\Omega}_1$. With such equation the unknown symbols will estimate more precisely in comparison with simple ALS approach. The receiver constructs the two residual equations and try to minimize them during the optimization task. The residual function based on the second norm to make the objective function convex. The optimization task can be solved with ALS and Newton algorithm.

$$\min_{vec(\mathbf{S}_{unk})\mathbf{h}} \mathbf{r}_s^H \mathbf{r}_s \tag{2.102}$$

$$\frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta \mathbf{h}^*} = -(\mathbf{D}_u + \mathbf{D}_k)^H (\mathbf{y} - \mathbf{D}_k \mathbf{h} - \mathbf{D}_u \mathbf{h}) = 0 \tag{2.103}$$

$$\frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta vec(\mathbf{S}_{unk})^*} = -(\mathbf{H}\boldsymbol{\Omega}_k)^H (\mathbf{y} - \mathbf{H}\boldsymbol{\Omega}_k vec(\mathbf{S}_{kn}) - \mathbf{H}\boldsymbol{\Omega}_u vec(\mathbf{S}_{unk})) = 0 \tag{2.104}$$

$$\mathbf{h}_{opt} = (\mathbf{D}_u + \mathbf{D}_k)^+ \mathbf{y} \tag{2.105}$$

$$vec(\mathbf{S}_{unk})_{opt} = (\mathbf{H}\boldsymbol{\Omega}_k)^+ (\mathbf{y} - \mathbf{H}\boldsymbol{\Omega}_k vec(\mathbf{S}_{kn})) \tag{2.106}$$

The objective function is following(2.102). We construct the partial derivative with respect to the unknown symbols and channel values. We used the Wirtinger calculus to find the derivative of the complex function. The derivative is written as the system of non-linear equations. We used ALS and Newton algorithms to solve the problem . The ALS algorithm solve the system of linear equations at the each iteration with respect to the each variable. The optimization process is explained below:

- Set the variables $\hat{\mathbf{h}}$ and $vec(\hat{\mathbf{S}}_{unk})$ as random values and zeros correspondingly.
- Solve the system of linear equations (2.105) for the $\hat{\mathbf{h}}$ array and renew values of the $\hat{\mathbf{h}}$

- Solve the system of linear equations (2.106) for the $vec(\hat{\mathbf{S}}_{unk})$ array and renew values of the $vec(\hat{\mathbf{S}}_{unk})$
- If the residual decrease is higher than tolerance repeat process from the step 2.

$$\mathbf{J}\theta = -\mathbf{d} \tag{2.107}$$

$$\theta_{k+1} = \theta_k - \mathbf{J}^{+}\mathbf{d} \tag{2.108}$$

$$\mathbf{J} = \begin{bmatrix} \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta \mathbf{h}^* \mathbf{h}^*} & \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta vec(\mathbf{S}^*)\mathbf{h}} \\ \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta \mathbf{h}^* vec(\mathbf{S})} & \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta vec(\mathbf{S}^*) vec(\mathbf{S})} \end{bmatrix} \tag{2.109}$$

$$\mathbf{d} = \begin{bmatrix} \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta \mathbf{h}^*} \\ \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta vec \mathbf{S}^*} \end{bmatrix} \tag{2.110}$$

$$\mathbf{d} = \begin{bmatrix} -(\dot{\mathbf{Đ}}_u + \mathbf{D}_k)^H (\mathbf{y} - \mathbf{D}_k \mathbf{h} - \dot{\mathbf{Đ}}_u \mathbf{h}) \\ -(\mathbf{H}\boldsymbol{\Omega}_k)^H (\mathbf{y} - \mathbf{H}\boldsymbol{\Omega}_k vec(\mathbf{S}_{kn}) - \mathbf{H}\boldsymbol{\Omega}_u vec(\mathbf{S}_{unk})) \end{bmatrix} \tag{2.111}$$

$$\mathbf{J} = \begin{bmatrix} (\mathbf{D}_k + \dot{\mathbf{Đ}}_u)^H (\mathbf{D}_k + \dot{\mathbf{Đ}}_u) & (\mathbf{D}_k + \dot{\mathbf{Đ}}_u)^H \mathbf{H}\boldsymbol{\Omega}_k \\ (\mathbf{H}\boldsymbol{\Omega}_k)^H (\mathbf{D}_k + \dot{\mathbf{Đ}}_u) & (\mathbf{H}\boldsymbol{\Omega}_k)^H \mathbf{H}\boldsymbol{\Omega}_k \end{bmatrix} \tag{2.112}$$

$$\theta = \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix} \tag{2.113}$$

The Newton algorithm solve the system of linear equations at each iteration for which given point [DES82]. We must write the equation of the Jacobian matrix (2.109) for the partial derivatives to determine the solved at each iteration equation. To write the Jacobian matrix we use property of the equality of the different notations of the same function in the same point. The resulting Jacobian matrix is written as follows (2.112). The Newton algorithm is explained below

- Set the variable θ in the certain way (2.113).
- Calculate the Jacobian matrix and derivative array in the given point θ
- Solve the system of linear equations (2.107) with respect to the θ
- Renew the θ with (2.108) equation.
- If the residual decrease is higher than tolerance repeat process from the step 2.

The Newton algorithm may be regularized and written in the more stable form with the Powell-Wolf stepsize rule[Fle00] and Levenberg-Marquardt regularization approach [Kel03].

2.5 Simulation results

In this section we present simulation results for the GFDM system and algorithms which was reviewed in the chapter 2.

2.5.1 Scenario 1.1

Table 2.1: GFDM simulation parameters.Scenario 1.1

Parameter	Variable	GFDM
Modulation scheme	μ	QPSK
Samples per symbol	T/T_s	32
Sub-carriers	F	32
Block size	T_s	15
Filter type		RRC
Roll-off-factor	α	0,0.3,0.5, 1
Channel	h	AWGN
Cyclic Prefix		No
Transmission		Uncoded

Performance of the GFDM system in comparison with different α coefficients are obtained through simulation. The parameters of the system are tabulated in Table (2.1). The GFDM system is simulated in the AWGN channel without coding. QPSK modulation scheme is used. The number of sub-carriers was $F = 32$, samples for each symbol was $T/T_s = F$ The block size was $T_s = 15$. The root-raised cosine filter was used with list of roll-of-factors. In the roll-of factor test we measured the SER for the different α for the Zero-Forced receiver and matched filter. The resulting SER plot is shown for the zero forced receiver at fig. (2.1) and for the matched filter receiver at fig. (2.2).

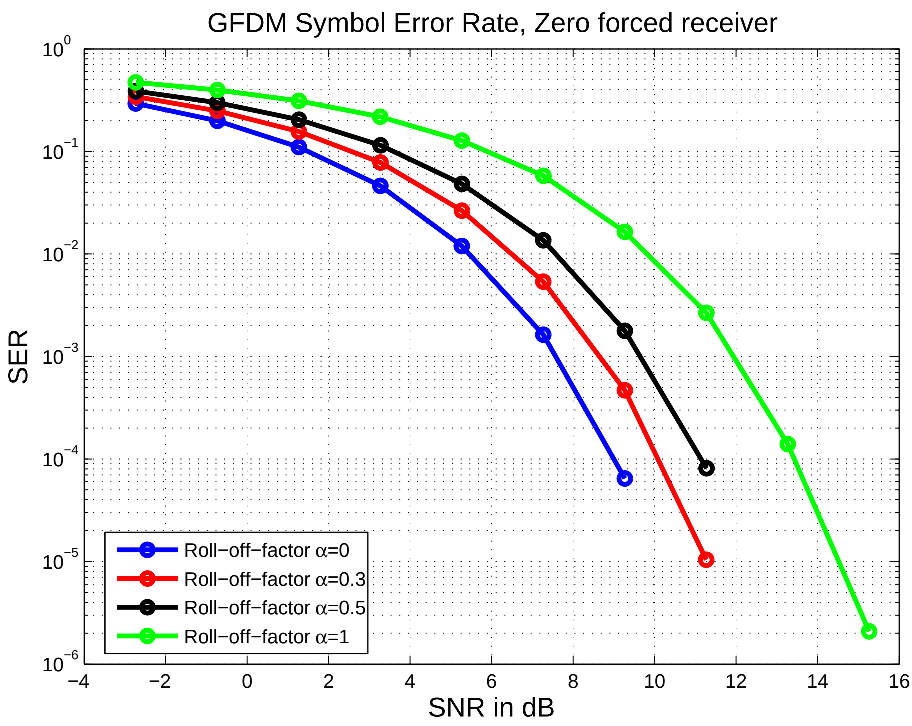

Figure 2.1: *SER dependency from α roll-off factor. Zero-forced receiver*

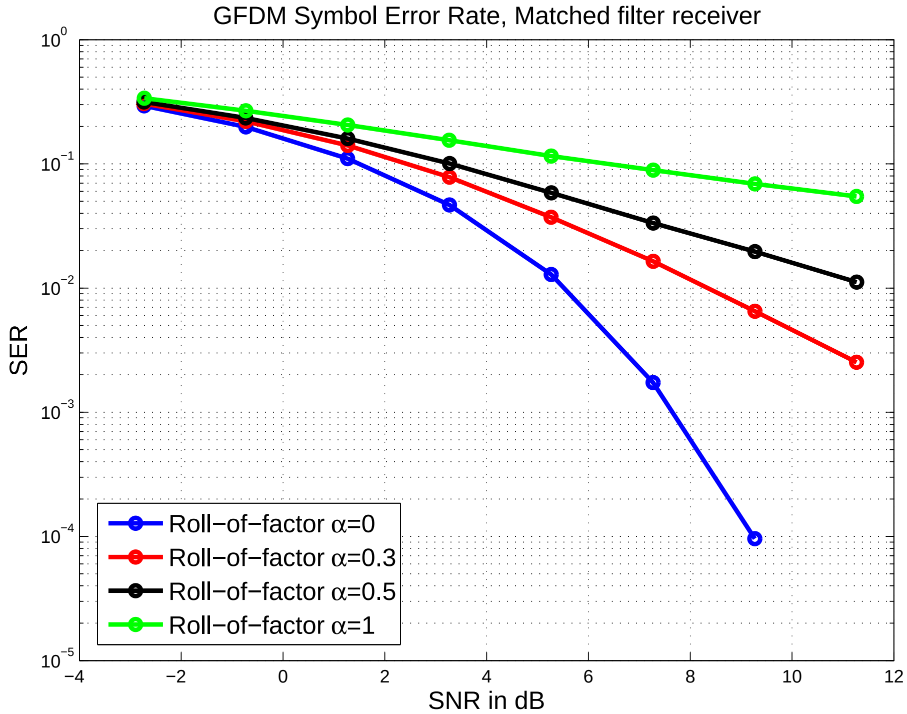

Figure 2.2: *SER dependency from α roll-off factor. Matched filter receiver*

2.5.2 Scenario 1.2

Performance results of the GFDM selection coefficient estimation and semi-blind receiver are obtained through simulation. The parameters of the system are tabulated in Table 2.2.

Table 2.2: GFDM simulation parameters.Scenario 1.2

Parameter	Variable	GFDM
Modulation scheme	μ	QPSK
Samples per symbol	T/T_s	32
Sub-carriers	F	32
Block size	T_s	15
Filter type		RRC
Roll-off-factor	α	0.5
Sub-carrier coefficients	$\mathbf{a}_i$	$unif(0, 1)$
Channel	h	AWGN
Cyclic Prefix		No
Transmission		Uncoded

The GFDM system is simulated with AWGN channel. QPSK modulation scheme is used. The number of sub-carriers was $F = 32$, samples for each symbol was $T/T_s = F$ The block size was $T_s = 15$. The root-raised cosine filter was used with roll-of-factor $\alpha = 0.5$. The selection coefficients was chosen as the random integer values in range from 0 to 1. The results of the GFDM performance is presented in the two plots, at first (2.3) is shown the SER in comparison with the case if receiver know the original $\mathbf{a}$ with the Zero-Forced receiver and with case if receiver doesn't know the $\mathbf{a}$. At second figure (2.4) is shown the reconstruction error for the $\mathbf{a}$.

$$r = \frac{|| \rlap{\mathbf{b}}\mathbf{a} - \mathbf{a} ||^2}{|| \mathbf{a} ||^2} \tag{2.114}$$

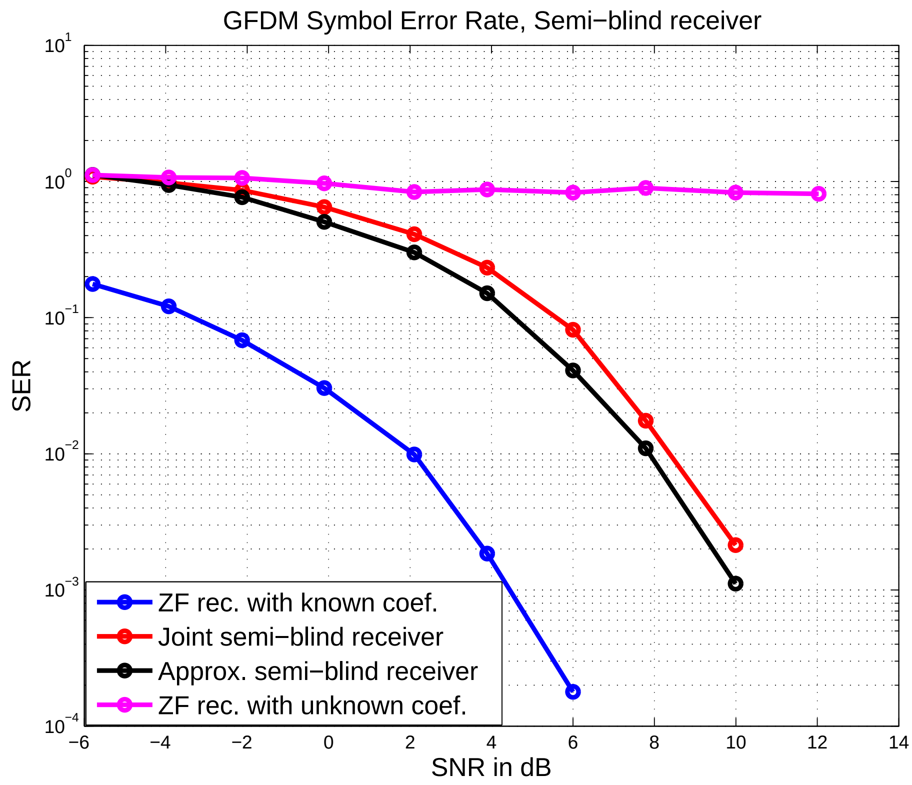

Figure 2.3: *SER comparison for the semi-blind and zero-forced receiver*

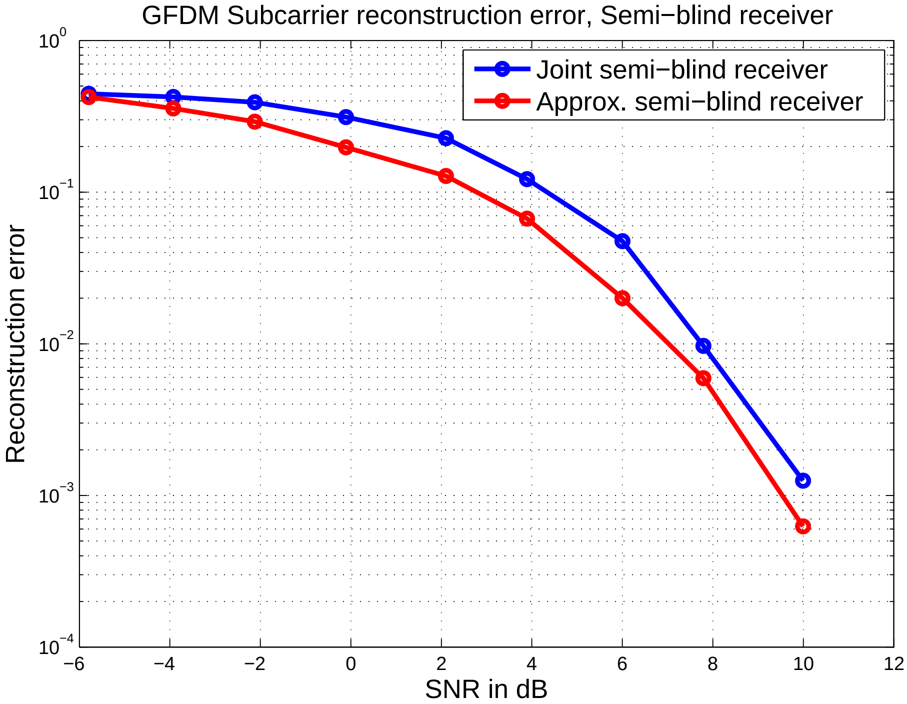

Figure 2.4: *Reconstruction error for the semi-blind receiver*

2.5.3 Scenario 1.3

There are additional plots for the joint and approximated algorithms comparison is shown at fig.(2.5) fig.(2.6) fig.(2.7). The comparison results are obtained through simulation. The simulation set-up is shown in the Table 2.3.The GFDM system is simulated with AWGN channel. QPSK modulation scheme is used. The parameters of the setup is the same as in the previous experiment. In the fig.(2.5) is shown the typical reconstruction error convergence of the both of the algorithms per iterations. As the real minimum point is shown values of the residual between generated matrices and received data. In the fig.(2.6) shown the time of the convergence of the algorithm with dependence for the SNR. In the fig.(2.7) shown the typical convergence of the residual (2.115) of the both of the algorithms per iterations.

$$\mathbf{r}_1^H \mathbf{r}_1 = (vec(\mathbf{y}) - \mathbf{\Omega}_1 \cdot vec(\boldsymbol{\mathfrak{S}}))^H (vec(\mathbf{y}) - \mathbf{\Omega}_1 \cdot vec(\boldsymbol{\mathfrak{S}})) \tag{2.115}$$

Table 2.3: GFDM simulation parameters.Scenario 1.3

Parameter	Variable	GFDM
SNR	$log(P_s/P_n)$	10
Samples per symbol	T/T_s	32
Sub-carriers	F	32
Block size	T_s	15
Filter type		RRC
Roll-off-factor	α	0.5
Sub-carrier coefficients	a_i	$unif0, 1$

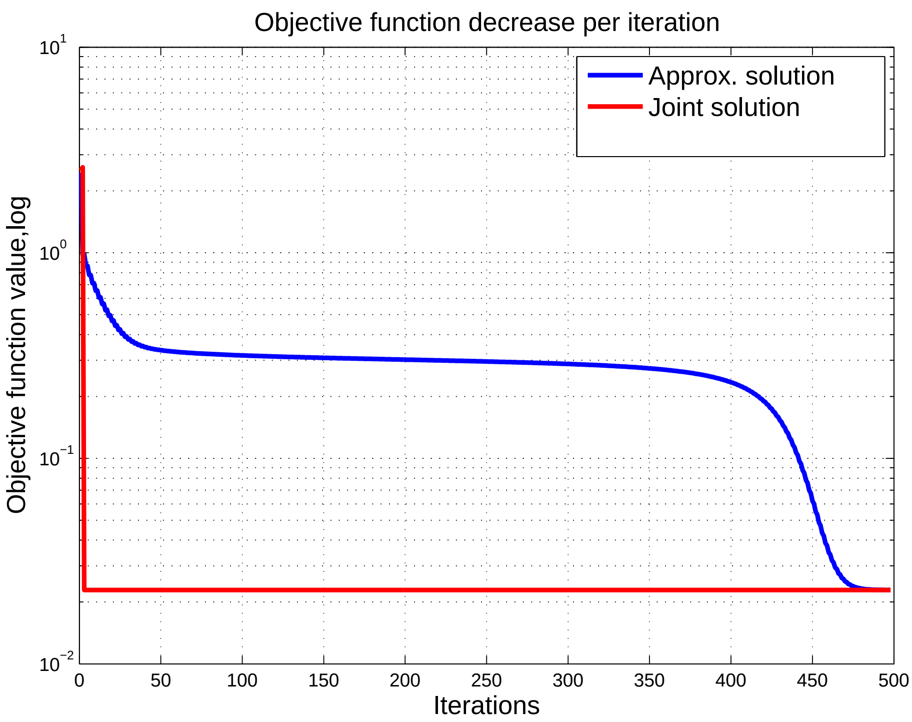

Figure 2.5: *Convergence of the semi-blind receiver with respect to the iterations*

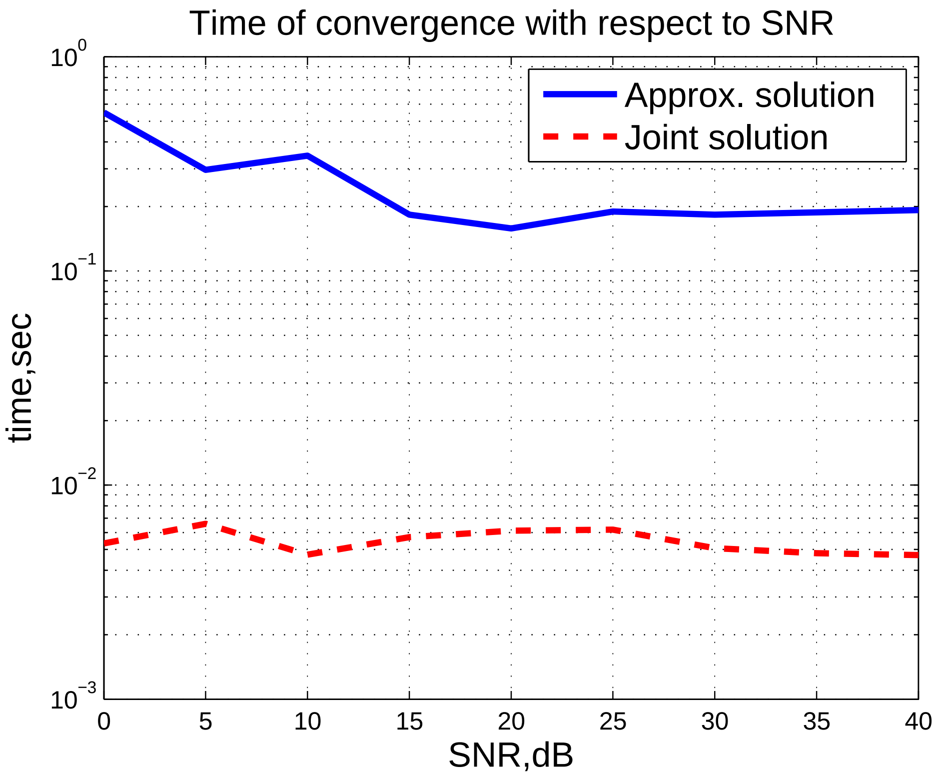

Figure 2.6: *Convergence time for the semi-blind receiver with respect to the SNR*

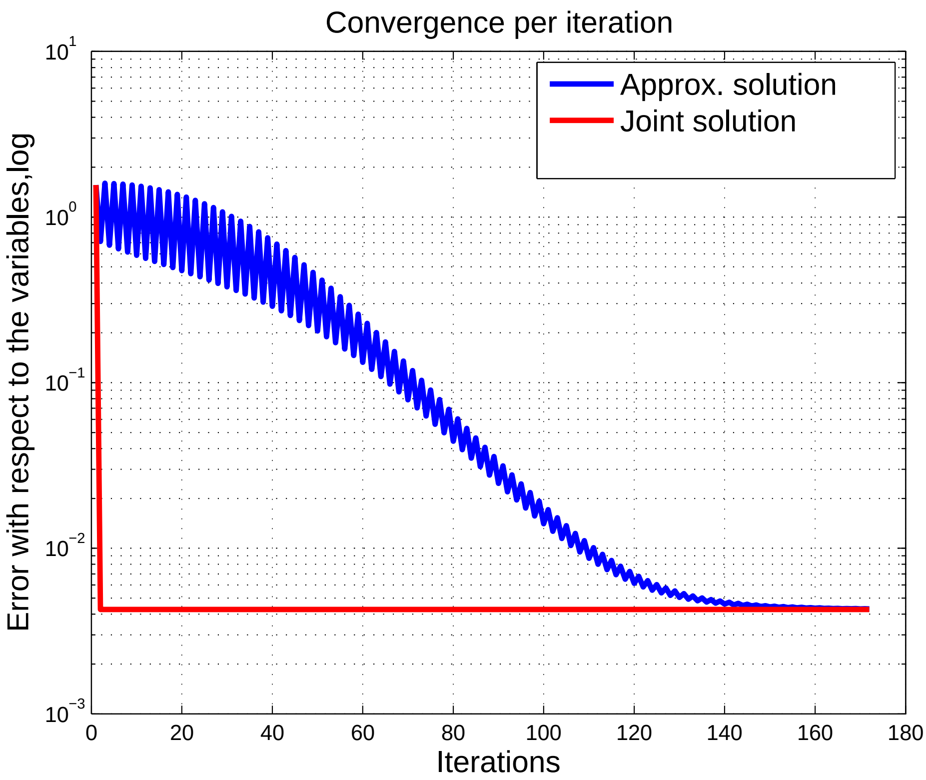

Figure 2.7: *Residual decrease for the semi-bind receiver with respect to the iterations*

2.5.4 Scenario 1.4

Table 2.4: GFDM simulation parameters.Scenario 1.4

Parameter	Variable	GFDM
Modulation scheme	μ	QPSK
Samples per symbol	T/T_s	32
Sub-carriers	F	32
Block size	T_s	15
Filter type		RRC
Roll-off-factor	α	1
Channel	h	Ped-A
Length of the channel	$L+1$	12
Cyclic Prefix		No
Transmission		Uncoded

The GFDM system semi-blind receiver is simulated in the frequency selective channel without coding. QPSK modulation scheme is used. The number of sub-carriers was $F = 32$, samples for each symbol was $T/T_s = F$ The block size was $T_s = 15$. The root-raised cosine filter was used with roll-of-factor $\alpha = 1$. The length of the channel equal to $L+1 = 12$. The values of the channel is presented in the table 2.4. The results of the GFDM performance are presented in the two plots, at first (2.8) is shown the SER in comparison with the Zero-Forcing FFT receiver and with case for different number of unknown symbols in one transmission block. In the simulation we define as the unknown symbol number of the non-training symbols in the $vec(\mathbf{S})$ At second figure (2.9) is shown the reconstruction error for the channel.

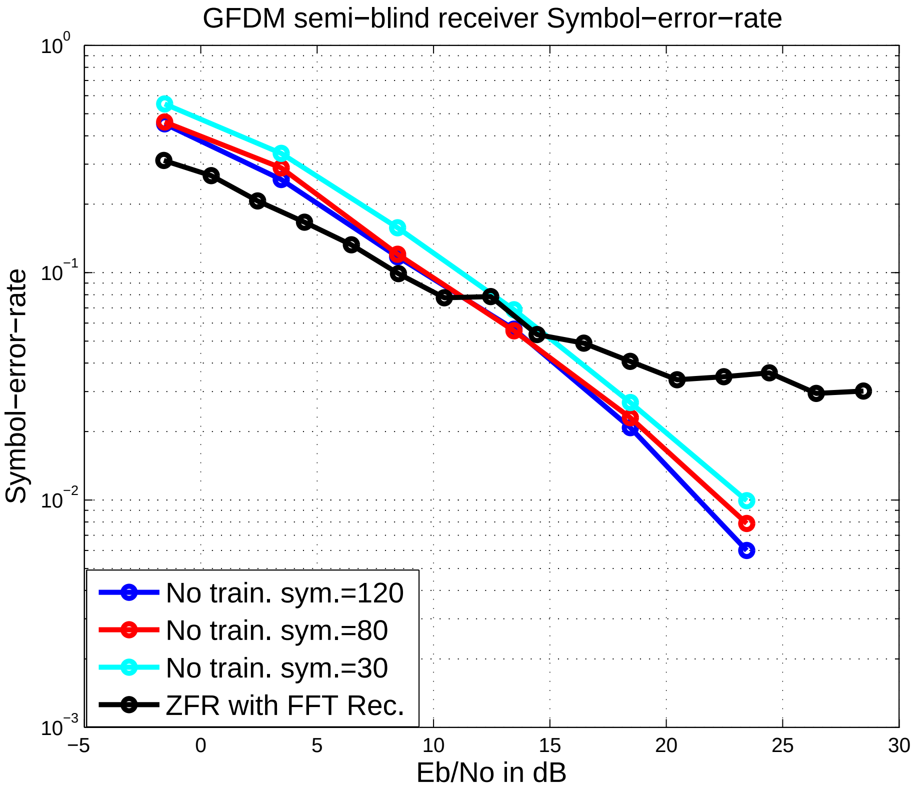

Figure 2.8: *SER for the semi-blind receiver with different number of known symbols*

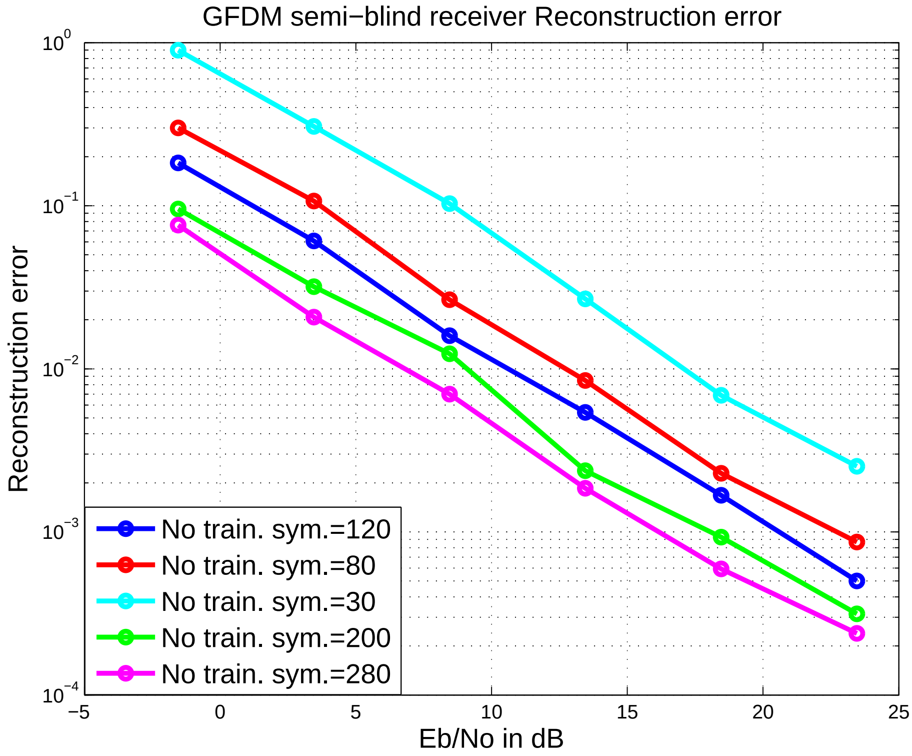

Figure 2.9: *Channel reconstruction error for the semi-blind receiver with different number of known symbols*

2.6 Conclusion

As conclusion for the first experiment we can say that zero forced receiver show much better results in the symbol detection in comparison with matched filter. The difference become significant when the $roll-off$ factor of the RRC filter increase. The ZF receiver has increased the SER slightly with α increasing which is shown at the fig.2.1. The MF receiver increase SER significantly with α increasing which is shown at fig.2.2. Explained results show, that ZF receiver should be preferred in the receiver. The ZF receiver achieve the better performance with the same self-interference ratio and allow decreasing the out-of-band radiation. In general, even if we consider the noise enhancement from the ZF receiver, the performance of the system is much better in case if we use the ZF receiver. The MF receiver is much more depend from the

$roll - off$ factor and has performance reduction due to this.

As conclusion for the experiment with frequency coefficient selection approach we can see that algorithm decrease performance in comparison with system where the frequency coefficients is known fig.2.3. Huge performance decrease come from the non-convenient coefficient selection way.We chose coefficient as 1 if the resulting $abs(a_1)$ value is higher than 0.5. There are better solutions to increase performance of the algorithm. We can use predefined information at the receiver. We can see that joint algorithm has slightly worse performance in symbol and frequency coefficients detection fig.2.3fig.2.4in comparison with approximated algorithm. This behaviour is explained from the initial point setup in the approximated algorithm. We take as the initially transmission coefficients equal to the zero. The approximated algorithm turn on less sub-carriers and make false negative errors. The false negative error lead to the better SER performance due to less estimated number of the symbols. The possible performance increase for the joint solution algorithm is the different weights for the objective function parts. The part for frequency coefficients estimations should have higher weight.

The next figures show advantages of the joint algorithm. As we can see from the fig. 2.5,fig.2.6,fig.2.7. The residual decrease show that joint algorithm converge much faster than approximated. The typical number of iteration of the joint algorithm equal to 2 and for the joint algorithm equal to the 200. So significant difference neglect the difference between the computationally expensive iterations in the joint algorithm. The fig. 2.6 confirm that fact. The joint algorithm converge independently from the SNR and take 100 times less time to converge. The approximated algorithm slightly depend from the SNR in the received data. The fig. 2.7 show that estimated values in the approximated algorithm decrease non-linearly. The shown fact mean that algorithm decrease at each iteration only one set of the variables, symbols or frequency coefficients. The overall conclusion is that approach is effective enough for the data estimation and there are possibility to increase performance with weights addition. Algorithm will show slightly better results if additional pre-processing will be used. The overall conclusion over the frequency coefficient algorithms we can say that spectrum sensing approach is applicable in the automatic way without a-priori information at the receiver. The receiver can estimate used subcarriers and estimate transmitted symbols with decrease in the performance. In the paper is explained very simple approach to coefficients estimation. Performance of the algorithm can be implemented in more efficient way without huge performance loss.

The next experiment show results of the semi-blind receiver. We estimated the unknown channel and the unknown symbols in the same time. We used the pedestrian-A type of frequency selective channel. The result for the SER if shown in the fig. 2.8. We can see from the results, algorithm show better performance in comparison with FFT receiver in the high SNR and slightly worse performance in the low SNR. The worse performance in the low SNR leads from the significant channel error estimation. We can see that performance in the FFT receiver has upper bound and FFT receiver doesn't allow to decrease the SER lower. This behavior leads from the rank of the channel matrix, which is not full. With the semi-blind receiver we can put the symbols in more convenient way to increase the performance of the algorithm. We can see that performance of the algorithm slightly depends from the number of training symbols. The difference from the number of used training symbols is not significant and equal to 2-3 dB. The fig.2.9 show the reconstruction error of the channel estimation. We can see that channel estimation error has significant dependence from the number of training symbols. There is trade-off between channel estimation and the number of received information. As conclusion, we can say that semi-blind receiver allow to estimate the channel and the symbols without channel prefix, only with the known symbols in data block. The semi-blind receiver outperforms the FFT receiver. The disadvantage of the semi-blind receiver is computational complexity.

3 Generalized frequency division multiplexing multiple input multiple output

3.1 System model

In this chapter we assume the GFDM MIMO system the flat-frequency channel[FKB09][Jan04] in that case the system is explained in the (3.1) equation with channel model a identity matrix without multipath propagation.

$$\mathbf{Y} = \mathbf{H}\mathbf{Z} + \mathbf{N} \tag{3.1}$$

$$\mathbf{H} \in \mathbb{C}^{M_r \times M_t} \mathbf{Z} \in \mathbb{C}^{M_t \times T} \mathbf{N}, \mathbf{Y} \in \mathbb{C}^{M_r \times T}$$

The GFDM system in MIMO AWGN case can be explained with the PARATUCK2 third order model. The difference between the SISO and MIMO case is significant[Jan04], but PARATUCK2 model allow to extend matrices to define the MIMO case inside of the same mathematical structure. The received data should have certain dimensionality $\mathbf{Y} \in \mathbb{C}^{M_r \times T}$ The symbol matrix $\mathbf{S}$ from the SISO system become three dimensional tensor $\mathcal{S} \in \mathbb{C}^{F \times T_s \times M_t}$ in the MIMO case. Additional dimension corresponds to the symbols which is transmitted from the different transmit antennas. The PARATUCK2 does not allow using tensors in the model. To overcome this constraint we transform the $\mathcal{X}$ into the stacked slices for each m_t (3.2).

$$\mathbf{S} = \begin{bmatrix} \mathcal{S}_{:,:,1} \\ \mathcal{S}_{:,:,2} \\ \vdots \\ \mathcal{S}_{:,:,M_t} \end{bmatrix} \tag{3.2}$$

$$\mathcal{S} \in \mathbb{C}^{F \times T_s \times M_t} \mathbf{S} \in \mathbb{C}^{F \cdot M_t \times T_s}$$

The $C^{[b]}$ matrix stay the same as in the SISO model and does not change with increasing of the transmit antennas (3.3). In the columns of the matrix stays the same time mixing filter which doesn't change from the number of the transmit and receive antennas. From the right hand side the $\mathbf{S}$ does not change and have number of the columns T_s which also satisfy the model.

$$\mathbf{C}^{[b]} \in \mathbb{C}^{T \times T_s} \tag{3.3}$$

The connection array $\mathbf{b}$ stay the same as in the SISO case (3.4). The necessary assumption for this is linear time invariant transmission channel in terms of the one transmission block. We fill the $\mathbf{b}$ array with all-ones as in the SISO case.

$$\mathbf{b} = \mathbf{1}_{T_s \times 1} \tag{3.4}$$

$$\mathbf{b} \in \mathbb{C}^{T_s \times 1}$$

The $\mathbf{C}^{[a]}$ matrix in the MIMO case is changed (3.5). Define this matrix in the MIMO case as $\mathbf{C}^{[a]'}$. Regarding to the $\mathbf{S}$ matrix, dimensions of the $\mathbf{C}^{[a]'}$ should be $T \times F \cdot M_t$. In comparison with the SISO model, in the MIMO case $\mathbf{S}$ should be multiplied with the $\mathbf{C}^{[a]}$ matrices. The new $\mathbf{C}^{[a]'}$ become stacked repeated version of the $\mathbf{C}^{[a]}$ matrix (3.6). Hadamard product between $\mathbf{S}$ and the $\mathbf{C}^{[a]'}$ become simple explanation of the sub-carrier modulation of the each sub-frequency at the each transmitter antenna. Sub-carriers in the each transmit antennas is equal, so $\mathbf{C}^{[a]}$ matrices for each m_t is also equal.

$$\mathbf{C}^{[a]'} = \begin{bmatrix}\mathbf{C}^{[a]} & \mathbf{C}^{[a]} & \cdots & \mathbf{C}^{[a]}\end{bmatrix} \in \mathbb{C}^{T\times M_t\cdot F} \tag{3.5}$$

$$\mathbf{C}^{[a]'} = (\mathbf{1}_{1\times M_t} \otimes \mathbf{C}^{[a]}) \in \mathbb{C}^{T\times M_t\cdot F} \tag{3.6}$$

The **a** array become the matrix and connect each m_r antenna with the each m_t transmit antenna and include every sub-carrier coefficient(3.7). Define sub-carrier coefficients as the $\mathcal{A}$ tensor with coefficients for each m_r, m_t, f. In the PARATUCK2 model **A** is the matrix, we must define the $\mathcal{A}$ as the matrix which will sum for every transmit antenna sub-carrier frequency with corresponding sub-carrier coefficient(3.8). Regarding to the **S** matrix, symbols is located sequentially for each sub-carrier at each transmit antenna. The **A** dimension is $M_r \times F \cdot M_t$. The first unfolding of the $\mathcal{A}$ have corresponding sub-carrier coefficients following the **S** symbols(3.9). This notation multiplies each sub-carrier coefficient **A** to each symbol from the **S**.

$$\mathcal{A} \in \mathbb{C}^{M_r\times F\times M_t} \tag{3.7}$$

$$\mathbf{A} = \mathcal{A}_{[1]} \tag{3.8}$$

$$\mathbf{A} \in \mathbb{C}^{M_r\times F\cdot M_t} \tag{3.9}$$

The overall model with explained matrices will be defined in the (3.10). This model define the received data. All operation, which was explained in the GFDM SISO part is also applicable in the GFDM MIMO case. It should be noted that the rank of the **A** is $min(M_r, F \cdot M_t)$ in the real case number of the receive antennas is much less than sub-carriers multiplied with transmit antennas. The rank of the matrix **A** show how many transmit coefficient possible to know in the system with known transmitted symbols.

$$\mathbf{HZ} = \mathbf{X} = \mathbf{A} \cdot (\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T)) \tag{3.10}$$

$$\mathbf{X} \in \mathbb{C}^{M_r \times T} \mathbf{A} \in \mathbb{C}^{M_r \times F \cdot M_t} \mathbf{C}^{[a]'} \in \mathbb{C}^{T \times F \cdot M_t}$$

$$\mathbf{S} \in \mathbb{C}^{F \cdot M_t \times T_s} \mathbf{C}^{[b]} \in \mathbb{C}^{T \times T_s} \mathbf{B} \in \mathbb{C}^{1 \times T_s}$$

3.2 Sub-carrier coefficient update approach

The spectrum sensing approach is applicable in the MIMO system too. One of the important properties in the MIMO system is diversity, which frequently achieved via spatial location of the transmitter and receiver antennas[Jan04]. It should be noted that for different transmitter-receiver antenna pair one sub-carrier may be occupied with another system, and not. Receiver must know the $\mathbf{A}$ to know how data is transmitted. Sub-carrier coefficients update is complex task due to the time mix and multi-frequency system. The number of the unknown variables in the system for MIMO case equal to the $M_r \cdot M_t \cdot F$. Number of the equations to solve equal to the $T \cdot M_r$ for each transmission block. The transmission coefficients can be found with specially structured known symbol matrix.

3.2.1 Approximated approach

The one way to find the matrix $\mathbf{A}$ with sub-carrier coefficients is semi-blind receiver. The number of time slots must equal to the number of the transmit antennas. The receiver may use approach from the previous chapter 2.3. Transmitter should construct special structure of the matrix $\mathbf{S}$. For each time slot the receiver can find the $M_r \cdot F$ transmission coefficients. The transmitter must fill at least M_t time slots with known symbols to find the all $\mathbf{A}$ matrix. The search algorithm for the matrix $\mathbf{A}$ is next:

- Transmitter put inside the transmission block specially structured symbols. Transmitter put known to the receiver symbol row to the one transmit antenna at the each time slot. In other words the symbols are sent from each sub-carrier frequency in the certain antenna. Transmitter repeats the same row, but from another transmit antenna at the next time slot. The transmitter repeat process M_t times for each antenna. The receiver must know the transmission order.

- The receiver has transmission block with M_t time slots length. The receiver uses semi-blind approach from the SISO case and assume received signal is sent from the SISO system. Semi-blind receiver approach estimates the sub-carrier coefficients for m_t-th transmit antenna.
- The receiver repeats process for another time slot, and estimates the sub-carrier coefficients for another transmit antenna. The receiver repeats process M_t times until estimates the all $\mathbf{A}$ matrix elements.

3.2.2 MIMO model based approach

The matrix $\mathbf{A}$ will be found with the equation (3.14) separation in the PARATUCK2 model. The model is written as the vectorized matrix $\mathbf{A}$ multiplication with the intermediate matrix $\mathbf{\Delta}$, as given in (3.15).

$$\mathbf{Y} = \mathbf{HZ} + \mathbf{N} \tag{3.11}$$

$$\mathbf{HZ} = \mathbf{X} = \mathbf{A} \cdot (\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \tag{3.12}$$

$$\mathbf{Y}, \mathbf{X}, \mathbf{N} \in \mathbb{C}^{M_r \times T} \mathbf{H} \in \mathbb{C}^{M_r \times M_t} \mathbf{Z} \in \mathbb{C}^{M_t \times T}$$

$$vec(\mathbf{X}) = vec(\mathbf{A} \cdot (\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T))) \tag{3.13}$$

The product of this two matrices explained in the vectorized form for both of the second matrices (3.14).The matrix $\mathbf{\Delta}$ constructed with property (3.14) and defined in (3.15). We used property (3.14) to find the sub-carrier coefficient matrix $\mathbf{A}$ from the received data. The overall equation is defined as (3.16).

$$vec(\mathbf{OP}) = (\mathbf{P}^T \otimes \mathbf{I}) \cdot vec(\mathbf{O}) \tag{3.14}$$

$$\mathbf{\Delta} = (\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T)) \tag{3.15}$$

$$vec(\hat{\mathbf{Y}}) = (\mathbf{\Delta}^T \otimes \mathbf{I}) \cdot vec(\hat{\mathbf{A}}) \tag{3.16}$$

We constructed the objective function to minimize the second norm of the vectorized residual explained via $vec(\mathbf{A})$ array (3.16). The function is non-holomorphic[Ste14] due to the complex values in the $vec(\mathbf{A})$. We have used Wirtinger calculus and have found the first partial derivative. We have equated first partial derivative to the zero[PP12]. The first partial derivative has global minimum point, because the second norm function is convex.

$$\mathbf{r}_5 = vec(\mathbf{Y}) - vec(\hat{\mathbf{Y}}) \tag{3.17}$$

$$\mathbf{\Delta} \in \mathbb{C}^{M_t \cdot F \times T} \mathbf{r}_5 \in \mathbb{C}^{M_r \cdot T \times 1}$$

$$\min_{vec(\mathbf{A})} \mathbf{r}_5^H \mathbf{r}_5 \tag{3.18}$$

$$\frac{\delta \mathbf{r}_5^H \mathbf{r}_5}{\delta vec(\hat{\mathbf{A}}^*)} = 0 \tag{3.19}$$

$$\frac{\delta \mathbf{r}_5^H \mathbf{r}_5}{\delta vec(\hat{\mathbf{A}}^*)} = \frac{\delta (vec(\mathbf{Y}) - vec(\hat{\mathbf{Y}}))^H (vec(\mathbf{Y}) - vec(\hat{\mathbf{Y}}))}{\delta vec(\hat{\mathbf{A}}^*)} \tag{3.20}$$

$$\frac{\delta \mathbf{r}_5^H \mathbf{r}_5}{\delta vec(\hat{\mathbf{A}}^*)} = (\mathbf{\Delta}^T \otimes \mathbf{I})^H (\mathbf{\Delta}^T \otimes \mathbf{I}) vec(\hat{\mathbf{A}}) - (\mathbf{I} \otimes \mathbf{\Delta})^H vec(\mathbf{Y}) = 0 \tag{3.21}$$

$$(\mathbf{\Delta}^T \otimes \mathbf{I})^H (\mathbf{\Delta}^T \otimes \mathbf{I}) vec(\hat{\mathbf{A}}) = (\mathbf{\Delta}^T \otimes \mathbf{I})^H vec(\mathbf{Y}) \tag{3.22}$$

$$vec(\hat{\mathbf{A}})_{opt} = ((\mathbf{\Delta}^T \otimes \mathbf{I})^H (\mathbf{\Delta}^T \otimes \mathbf{I}))^{-1} (\mathbf{\Delta}^T \otimes \mathbf{I})^H vec(\mathbf{Y}) \tag{3.23}$$

$$vec(\hat{\mathbf{A}})_{opt} = ((\mathbf{\Delta}^* \otimes \mathbf{I})(\mathbf{\Delta}^T \otimes \mathbf{I}))^{-1}(\mathbf{\Delta}^* \otimes \mathbf{I})vec(\mathbf{Y}) \tag{3.24}$$

The optimal solution in the second norm sense is written as here (3.25). We can simplify defined equation (3.25) with linear algebra properties. The simplified equation is defined as (3.27)

$$vec(\hat{\mathbf{A}})_{opt} = ((\mathbf{\Delta}^* \mathbf{\Delta}^T) \otimes \mathbf{I})^{-1}(\mathbf{\Delta}^* \otimes \mathbf{I})vec(\mathbf{Y}) \tag{3.25}$$

$$vec(\hat{\mathbf{A}})_{opt} = ((\mathbf{\Delta}^* \mathbf{\Delta}^T)^{-1} \otimes \mathbf{I}^{-1})(\mathbf{\Delta}^* \otimes \mathbf{I})vec(\mathbf{Y}) \tag{3.26}$$

$$vec(\hat{\mathbf{A}})_{opt} = ((\mathbf{\Delta}^* \mathbf{\Delta}^T)^{-1} \mathbf{\Delta}^* \otimes \mathbf{I})vec(\mathbf{Y}) \tag{3.27}$$

We applied the Kronecker product properties and have analysed the rank of the inverse matrix $((\mathbf{\Delta}^* \mathbf{\Delta}^T)^{-1} \otimes \mathbf{I}^{-1})(\mathbf{\Delta}^* \otimes \mathbf{I})$ to find number of the coefficients which is possible to estimate in the one transmission block (3.27). The rank of the Kronecker product equals to the product between the multiplied matrix ranks (3.28). The identity matrix has the maximal rank, so the rank of the matrix is defined by the second part of the equation (3.30). Consider the matrix (3.15), The size of the core matrix $\mathbf{S}$ is equal to the $M_t \cdot F \times T_s$, rank of the matrix $(\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T)$ is less of equal to the T_s. There is inequality which define the rank of the Hadamard product (3.32) as written in the [ZK06][MB06][LT06]. The matrix $\mathbf{C}^{[a]'T}$ has the rank equal to the $min(F, T)$ because it based on the stacked repetition of the $\mathbf{C}^{[a]T}$ matrix. The $\mathbf{C}^{[a]T}$ matrix has the full rank $min(F, T)$. The matrix (3.33) has the rank equal to the matrix $\mathbf{\Delta}$ rank. We have defined the system of the linear equations rank with equation (3.34).

$$rank(\mathbf{\Delta}^* \mathbf{\Delta}^T)^{-1} \mathbf{\Delta}^* \otimes \mathbf{I}) = (rank(\mathbf{I}))rank((\mathbf{\Delta}^* \mathbf{\Delta}^T)^{-1} \mathbf{\Delta}^*) \tag{3.28}$$

$$rank(\mathbf{I}) = M_r \tag{3.29}$$

$$rank((\mathbf{\Delta}^* \mathbf{\Delta}^T)^{-1} \mathbf{\Delta}^*)) = rank(\mathbf{\Delta}) \tag{3.30}$$

$$rank(\mathbf{\Delta}) \leq rank(\mathbf{C}^{[a]'T})rank((\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T))) \tag{3.31}$$

$$rank(\mathbf{C}^{[a]'T}) = min(T_s, F) \tag{3.32}$$

$$rank((\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T))) = rank(\mathbf{S}) \leq T_s \tag{3.33}$$

$$rank((\mathbf{\Delta}^H \mathbf{\Delta})^{-1} \mathbf{\Delta}^H)) \leq T_s \cdot min(T_s, F) \tag{3.34}$$

$$rank(((\mathbf{\Delta}^* \mathbf{\Delta}^T)^{-1} \mathbf{\Delta}^* \otimes \mathbf{I})) \leq T_s \cdot M_t \cdot min(T_s, F) \tag{3.35}$$

The transmitter can use the specially structured symbol matrix as explained in the semi-blind receiver for SISO case part. We will find the all matrix **A** via least squares solution.We consider all dependencies with the MIMO based approach. We don't make assumption about SISO case and hold precision of the estimation.

3.2.3 Decomposition based approach

The receiver can estimate sub-carrier selection coefficients with less number of unknowns. The tensor $\mathcal{A}$ from the PARATUCK2 model is explained as unfolding of tensor decomposition and decrease number of estimated variables.

$$\mathbf{Y} = \mathbf{HZ} + \mathbf{N} = \mathbf{X} + \mathbf{N} \tag{3.36}$$

$$\mathbf{HZ} = \mathbf{X} = \mathbf{A} \cdot (\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \tag{3.37}$$

$$\mathbf{A} = \mathcal{A}_{[1]} = \mathbf{A_A} \cdot (\mathbf{A_C} \diamond \mathbf{A_B})^T \tag{3.38}$$

$$\mathbf{X} = \mathbf{A}_A \cdot (\mathbf{A}_C \diamond \mathbf{A}_B)^T \cdot \mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \tag{3.39}$$

$$\mathcal{A} \in \mathbb{C}^{M_r \times F \times M_t}$$

where $\mathbf{A}_A \in \mathbb{C}^{M_r \times r} \mathbf{A}_B \in \mathbb{C}^{F \times r} \mathbf{A}_C \in \mathbb{C}^{M_t \times r}$ and therefore receiver decrease number of variables to solve the task(3.40) where we must find the N_1 variables. In case of the decomposition approach the number of the estimated values is expressed with another equation (3.41), where previous variables come in the sum, but we add new variable which we can adjust. Relation from r to possible decrease of necessary equations grows polynomially. Rank of the decomposition decrease number of the variables in case if r from the (3.42) will be higher than rank of the tensor decomposition.

$$N_1 = M_r \cdot M_t \cdot F \tag{3.40}$$

$$N_2 = r \cdot (M_r + M_t + F) \tag{3.41}$$

$$r = \frac{M_r \cdot M_t \cdot F}{M_r + M_t + F} \tag{3.42}$$

We have written equation in the vectorized residual form and have classical system of linear equations.

$$\mathbf{X} = \mathcal{A}_{[1]} \cdot \mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \tag{3.43}$$

$$\mathbf{X} = \mathbf{A}_A \cdot (\mathbf{A}_C \diamond \mathbf{A}_B)^T \cdot \mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \tag{3.44}$$

$$vec(\mathbf{A}_A \cdot (\mathbf{A}_C \diamond \mathbf{A}_B)^T \cdot \mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T))) \tag{3.45}$$

Possible to separate $vec()$ operation for the $\mathbf{A}_A$ and another part of the multiplication exploiting vec operation properties (3.46) which we can see from the equation (3.43) .

The receiver estimate the $\mathbf{A}_A$ matrix in the least squares sense from equation (3.50).

$$vec(\mathbf{OP}) = (\mathbf{P}^T \otimes \mathbf{I}) \cdot vec(\mathbf{O}) \tag{3.46}$$

$$vec(\mathbf{X}) = vec(\mathbf{A}_A \cdot (\mathbf{A}_C \diamond \mathbf{A}_B)^T \cdot \mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T))) \tag{3.47}$$

$$\mathbf{D}_{in} = (\mathbf{A}_C \diamond \mathbf{A}_B)^T \cdot \mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \tag{3.48}$$

$$vec(\mathbf{X}) = vec(\mathbf{A}_\mathbf{A} \cdot \mathbf{D}_{in}) \tag{3.49}$$

$$vec(\mathbf{X}) = (\mathbf{D}_{in}^T \otimes \mathbf{I}_{M_r}) \cdot vec(\mathbf{A}_\mathbf{A}) \tag{3.50}$$

The receiver separate Khatri-Rao product components from the equation (3.52) using another property of the *vec* operation (3.51). The receiver can find Khatri-Rao product and must separate $\mathbf{A}_B$ and the $\mathbf{A}_C$ matrix from the product.

$$vec(\mathbf{OPL}) = (\mathbf{L}^T \otimes \mathbf{O}) \cdot vec(\mathbf{P}) \tag{3.51}$$

$$vec(\mathbf{X}) = vec(\mathbf{A}_A \cdot (\mathbf{A}_C \diamond \mathbf{A}_B)^T \cdot \mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T))) \tag{3.52}$$

$$vec(\mathbf{X}) = ((\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T))^T \otimes \mathbf{A}_\mathbf{A}) vec((\mathbf{A}_C \diamond \mathbf{A}_B)^T) \tag{3.53}$$

$$\mathbf{D}_{sec} = ((\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \otimes \mathbf{A}_\mathbf{A}) \tag{3.54}$$

$$vec(\mathbf{X}) = \mathbf{D}_{sec} \cdot vec((\mathbf{A}_C \diamond \mathbf{A}_B)^T) \tag{3.55}$$

$$\mathbf{D}_{in} \in \mathbb{C}^{T \times M_r \times r} \mathbf{D}_{sec} \in \mathbb{C}^{T \cdot M_t \times M_t \cdot F \cdot r}$$

Possible to separate $\mathbf{A}_B$ and the $\mathbf{A}_C$ matrices from the Khatri-Rao product, using the decomposition of the Khatri-Rao product(3.56). The receiver doesn't estimate over the Khatri-Rao product, but estimates the vectorized product version (3.55). The receiver separate Khatri-Rao product components with the Hadamard product, but doesn't reveal necessary matrices in the matrix algebra. We introduce two other equations (3.57) (3.59) to reveal $\mathbf{A}_B$ and the $\mathbf{A}_C$ in the clear form.

Matrices inside of the Khatri-Rao product is separable with vectorization operation following the equations (3.56) (3.57). We can estimate each of the matrix and decrease number of the variables in comparison with matrix based approach. We can separate the (3.56) with additional presented operations.

$$vec((\mathbf{A_C} \diamond \mathbf{A_B})^T) = diag(\mathbf{K}_1 \cdot vec(\mathbf{A_C}^T)) \cdot \mathbf{K}_2 \cdot vec(\mathbf{A_B}^T) \tag{3.56}$$

$$\mathbf{L}_B = diag(\mathbf{K}_1 \cdot vec(\mathbf{A_C}^T)) \cdot \mathbf{K}_2 \in \mathbb{C}^{M_t \cdot F \cdot r \times F \cdot r} \tag{3.57}$$

$$vec((\mathbf{A_C} \diamond \mathbf{A_B})^T) = diag(\mathbf{K}_2 \cdot vec(\mathbf{A_B}^T))\mathbf{K}_1 \cdot vec(\mathbf{A_C}^T) \tag{3.58}$$

$$\mathbf{L}_C = diag(\mathbf{K}_2 \cdot vec(\mathbf{A_B}^T))\mathbf{K}_1 \in \mathbb{C}^{M_t \cdot F \cdot r \times M_t \cdot r} \tag{3.59}$$

$$\mathbf{K}_1 = \begin{bmatrix} \mathbf{I}_r & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{I}_r & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & & & \\ \mathbf{I}_r^F & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_r & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_r & \cdots & \mathbf{0} \\ & & & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I}_r \end{bmatrix} = \mathbf{I}_{M_t} \otimes (\mathbf{1}_{F \times 1} \otimes \mathbf{I}_r) \tag{3.60}$$

$$\mathbf{K}_2 = \begin{bmatrix} \mathbf{I}_{F\cdot r} \\ \mathbf{I}_{F\cdot r} \\ \vdots \\ \mathbf{I}_{F\cdot r} \end{bmatrix} = \mathbf{1}_{M_t \times 1} \otimes \mathbf{I}_{F\cdot r} \tag{3.61}$$

We have constructed the objective function to estimate minimum point of the residual between estimated and the received data(3.62). We have rewritten the objective function with respect to the each of the variable set(3.64). The objective function is non-holomorphic [ADKM+15] and we have used the Wirtinger calculus to estimate the partial derivatives(3.70) from the each of the variables set (3.71).

$$\min_{\begin{bmatrix} vec(\hat{\mathbf{A}}_{\mathbf{A}}) \\ vec(\hat{\mathbf{A}}_{\mathbf{B}}) \\ vec(\hat{\mathbf{A}}_{\mathbf{C}}) \end{bmatrix}} (vec(\mathbf{Y}) - vec(\hat{\mathbf{X}}))^H (vec(\mathbf{Y}) - vec(\hat{\mathbf{X}})) \tag{3.62}$$

$$\mathbf{r}_4 = (vec(\mathbf{Y}) - vec(\hat{\mathbf{X}})) \tag{3.63}$$

$$vec(\mathbf{X}) = \mathbf{\Gamma}_1 vec(\hat{\mathbf{A}}_{\mathbf{A}}) = \mathbf{\Gamma}_2 vec(\hat{\mathbf{A}}_{\mathbf{B}}^T) = \mathbf{\Gamma}_3 vec(\hat{\mathbf{A}}_{\mathbf{C}}^T) \tag{3.64}$$

$$\mathbf{\Gamma}_1 = (\mathbf{D}_{in}^T \otimes \mathbf{I}_{M_r}) \tag{3.65}$$

$$\mathbf{\Gamma}_2 = \mathbf{D}_{sec} \cdot \mathbf{L}_B \tag{3.66}$$

$$\mathbf{\Gamma}_3 = \mathbf{D}_{sec} \cdot \mathbf{L}_C \tag{3.67}$$

$$\mathbf{\Gamma}_1 \in \mathbb{C}^{T\cdot M_r \times r\cdot M_r} \mathbf{\Gamma}_2 \in \mathbb{C}^{T\cdot M_r \times r\cdot F} \mathbf{\Gamma}_3 \in \mathbb{C}^{T\cdot M_r \times r\cdot M_t}$$

$$\mathbf{r}_4 = vec(\mathbf{Y}) - \mathbf{\Gamma}_1 vec(\boldsymbol{\lambda}_{\mathbf{A}}) = vec(\mathbf{Y}) - \mathbf{\Gamma}_2 vec(\boldsymbol{\lambda}_{\mathbf{B}}^T) \tag{3.68}$$

$$\mathbf{r}_4 = vec(\mathbf{Y}) - \mathbf{\Gamma}_3 vec(\boldsymbol{\lambda}_{\mathbf{C}}^T) \tag{3.69}$$

$$\min_{\begin{bmatrix} vec(\boldsymbol{\lambda}_{\mathbf{A}}) \\ vec(\boldsymbol{\lambda}_{\mathbf{B}}) \\ vec(\boldsymbol{\lambda}_{\mathbf{C}}) \end{bmatrix}} \mathbf{r}_4^H \mathbf{r}_4 \tag{3.70}$$

$$\delta \mathbf{r}_4^H \mathbf{r}_4 = \begin{bmatrix} \frac{\delta \mathbf{r}_4^H \mathbf{r}_4}{\delta vec(\boldsymbol{\lambda}_{\mathbf{A}}^*)} \\ \frac{\delta \mathbf{r}_4^H \mathbf{r}_4}{\delta vec(\boldsymbol{\lambda}_{\mathbf{B}}^*)} \\ \frac{\delta \mathbf{r}_4^H \mathbf{r}_4}{\delta vec(\boldsymbol{\lambda}_{\mathbf{C}}^*)} \end{bmatrix}^T = 0 \tag{3.71}$$

$$\frac{\delta \mathbf{r}_4^H \mathbf{r}_4}{\delta vec(\boldsymbol{\lambda}_{\mathbf{A}}^*)} = -\mathbf{\Gamma}_1^H (vec(\mathbf{Y}) - \mathbf{\Gamma}_1 vec(\boldsymbol{\lambda}_{\mathbf{A}})) \tag{3.72}$$

$$\frac{\delta \mathbf{r}_4^H \mathbf{r}_4}{\delta vec(\boldsymbol{\lambda}_{\mathbf{B}}^*)} = -\mathbf{\Gamma}_2^H (vec(\mathbf{Y}) - \mathbf{\Gamma}_2 vec(\boldsymbol{\lambda}_{\mathbf{B}}^T)) \tag{3.73}$$

$$\frac{\delta \mathbf{r}_4^H \mathbf{r}_4}{\delta vec(\boldsymbol{\lambda}_{\mathbf{C}}^*)} = -\mathbf{\Gamma}_3^H (vec(\mathbf{Y}) - \mathbf{\Gamma}_3 vec(\boldsymbol{\lambda}_{\mathbf{C}}^T)) \tag{3.74}$$

The receiver must solve three system of linear equation to find the transmission coefficients. We use the ALS algorithm. The ALS algorithm works in the certain order:

- Set the initial point $\boldsymbol{\lambda}_{\mathbf{A}}, \boldsymbol{\lambda}_{\mathbf{B}}, \boldsymbol{\lambda}_{\mathbf{C}}$ as the random complex valued matrices
- Solve the equation $vec(\mathbf{Y}) - \mathbf{\Gamma}_1 vec(\boldsymbol{\lambda}_{\mathbf{A}})$ with given $\boldsymbol{\lambda}_{\mathbf{B}}, \boldsymbol{\lambda}_{\mathbf{C}}$ and update $\boldsymbol{\lambda}_{\mathbf{A}}$ matrix
- Solve the equation $vec(\mathbf{Y}) - \mathbf{\Gamma}_2 vec(\boldsymbol{\lambda}_{\mathbf{B}}^{\mathbf{T}})$ with given $\boldsymbol{\lambda}_{\mathbf{A}}, \boldsymbol{\lambda}_{\mathbf{C}}$ and update $\boldsymbol{\lambda}_{\mathbf{B}}$

matrix

- Solve the equation $vec(\mathbf{Y}) - \mathbf{\Gamma}_3 vec(\hat{\mathbf{X}}^{\mathbf{T}}_{\mathbf{C}})$ with given $\hat{\mathbf{X}}_{\mathbf{A}}, \hat{\mathbf{X}}_{\mathbf{B}}$ and update $\hat{\mathbf{X}}_{\mathbf{C}}$ matrix.
- Check the reconstruction error $\frac{||vec(\mathbf{Y}) - vec(\hat{\mathbf{X}})||^2}{||vec(\mathbf{Y})||^2}$. If decrease of the reconstruction error is higher than tolerance, repeat from step 2.

The ALS algorithm is very unstable to the ill-conditioned matrices. Possible to add regularization parameters for the algorithm using the Levenberg-Marquardt based improvement.

3.2.3.1 Rank analysis

Consider the possible ranks of the generating matrices $\mathbf{\Gamma}_1, \mathbf{\Gamma}_2, \mathbf{\Gamma}_3$ to analyse the decrease size of the task to estimate the all sub-carrier selection coefficients. Two generating matrices consist from two parts corresponding to the symbols and the Khatri-Rao product separation[RL03].

$$vec(\hat{\mathbf{X}}) = \mathbf{\Gamma}_1 vec(\hat{\mathbf{X}}_{\mathbf{A}}) = \mathbf{\Gamma}_2 vec(\hat{\mathbf{X}}^T_{\mathbf{B}}) = \mathbf{\Gamma}_3 vec(\hat{\mathbf{X}}^T_{\mathbf{C}}) \tag{3.75}$$

$$\mathbf{\Gamma}_1 = (\mathbf{D}^T_{in} \otimes \mathbf{I}_{M_r}) \tag{3.76}$$

$$\mathbf{\Gamma}_3 = \mathbf{D}_{sec} \cdot \mathbf{L}_C \tag{3.77}$$

Consider the the matrix $\mathbf{\Gamma}_1$. The matrix is based on the Kronecker product of the two matrices. The Kronecker product rank equal to the (3.78), where identity matrix has full rank[AB00]. The rank depends only from the first part of the equation (3.79).

$$\mathbf{\Gamma}_1 = (\mathbf{D}^T_{in} \otimes \mathbf{I}_{M_r}) \tag{3.78}$$

$$rank(\mathbf{O} \otimes \mathbf{P}) = rank(\mathbf{O}) rank(\mathbf{P}) \tag{3.79}$$

$$rank(\mathbf{O} \odot \mathbf{P}) \leq rank(\mathbf{O})rank(\mathbf{P}) \tag{3.80}$$

$$rank(\mathbf{OP}) \leq min(rank(\mathbf{O}), rank(\mathbf{P})) \tag{3.81}$$

$$rank(\mathbf{\Gamma}_1) = rank(\mathbf{D}_{in}^T)rank(\mathbf{I}_{M_r}) \tag{3.82}$$

$$\mathbf{D}_{in}^T = (\mathbf{A}_C \diamond \mathbf{A}_B)^T \cdot \mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T))^T \tag{3.83}$$

$$rank(\mathbf{C}^{[a]'}) = min(F, T) \tag{3.84}$$

$$rank((\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) = min(T_s, T) \tag{3.85}$$

$$rank(\mathbf{S}) \leq min(F \cdot M_t, T_s) \tag{3.86}$$

$$rank((\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T)) \leq min(F \cdot M_t, T_s, T) \tag{3.87}$$

$$rank(\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T)) \leq (min(F \cdot M_t, T_s, T) \cdot min(F, T)) \tag{3.88}$$

$$rank(\mathbf{D}_{in}) \leq (min(F \cdot M_t, T_s, T) \cdot min(F, T)) \cdot M_r \tag{3.89}$$

The upper bound of the rank (3.88) is equal to the $min(M_t \cdot F, T)$ due to the size of the resulting matrix. The T is the highest value from the list, so we can neglect them, because it will be higher that any other value in the list(3.89). The real rank does not depend from the multiplication with rank of the $\mathbf{C}^{[a]'}$ matrix due to the symbol filling of the matrix. Consider the Khatri-Rao product [LMV00]. Row subspace doesn't change from the product due to the problem statement $r \leq F \cdot M_t$, following reveal upper bound of the Khatri-Rao product. Result is shown in the equation . If the rank of is full and equal to the T_s the system is bounded with rank r, which allow decreasing rank of the matrix $\mathbf{S}$ from T_s to r and find the $\hat{\mathbf{A}}_{\mathbf{A}}$.

$$rank(\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{b}^T)^T)) \leq (min(F \cdot M_t, T_s) \cdot F) \tag{3.90}$$

$$rank(\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T)) \leq (min(F \cdot M_t, T_s)) \tag{3.91}$$

$$rank((\hat{\mathbf{A}}_C \diamond \hat{\mathbf{A}}_B)^T) \leq r \tag{3.92}$$

$$rank(\mathbf{D}_{in}) \leq min(r, F \cdot M_t, T_s) \tag{3.93}$$

Next step is $\mathbf{\Gamma}_2$ rank analysis. We doesn't write the overall explanation of the analysis and write only the upper bounds of the rank for each matrix.

$$rank(\mathbf{\Gamma}_2) \leq min((M_t \cdot r), (F \cdot r), F \cdot M_t, T_s)) \cdot min(M_r, r) \tag{3.94}$$

$$rank(\mathbf{\Gamma}_3) \leq min((M_t \cdot r), (F \cdot r), F \cdot M_t, T_s)) \cdot min(M_r, r) \tag{3.95}$$

3.2.4 Semi-blind receiver for MIMO

In the previous subsection was solved task with the transmission coefficient search for the case if the number of the M_t antennas is equal to the T_s. Due to high size of transmission blocks, the the M_t is not obligatory equal to the T_s. To increase efficiency of the system we can implement the semi-blind receiver from the SISO approach to the MIMO case, with known symbols.

$$\mathbf{r}_1 = vec(\mathbf{Y}) - \mathbf{\Omega}_1 \cdot vec(\mathbf{S}^T) = \mathbf{r}_2$$

$$\mathbf{r}_2 = vec(\mathbf{Y}) - \mathbf{\Omega}_2 \cdot vec(\mathbf{A}^T)$$

$$\mathbf{r}_3 = \mathbf{q} - \mathbf{A}_{sel} vec(\mathbf{S});$$

$$\mathbf{J}\theta_\delta = \mathbf{d} \tag{3.96}$$

$$\theta_{k+1} = \theta_k + \theta_\delta \tag{3.97}$$

Due to the same PARATUCK2 third order model the algorithm is almost the same. The semi-blind task statement will be written in the same form as in the section. The modulation matrices will be changed due to the new structure of the signal . The equations is changed due to the transposed version of the vectorized symbol matrix and hand vectorized received matrix $\mathbf{Y}$. The all other equations is the same as in the joint semi-blind receiver it the SISO case. The start block of the known symbols increase from the one column, to the M_t columns, for each transmit antenna, and other symbols in the transmission block are filled with data symbols.

$$\mathbf{\Omega}_1 = \begin{bmatrix} (\mathbf{C}^{[a]'T} \diamond \mathbf{C}^{[b]T}) \diamond (\mathbf{A}_{1,:} \otimes \mathbf{B}) \\ (\mathbf{C}^{[a]'T} \diamond \mathbf{C}^{[b]T}) \diamond (\mathbf{A}_{2,:} \otimes \mathbf{B}) \\ \vdots \\ (\mathbf{C}^{[a]'T} \diamond \mathbf{C}^{[b]T}) \diamond (\mathbf{A}_{M_r,:} \otimes \mathbf{B}) \end{bmatrix} \tag{3.98}$$

$$\mathbf{\Delta} = (\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T)) \tag{3.99}$$

$$\mathbf{\Omega}_2 = (\mathbf{I}_{M_r} \otimes \mathbf{\Delta})^T \tag{3.100}$$

$$\mathbf{d} = \begin{bmatrix} -\mathbf{\Omega}_1^H(\mathbf{r}_1) + \mathbf{A}_{sel}(\mathbf{r}_3) \\ -\mathbf{\Omega}_2^H(\mathbf{r}_2) \end{bmatrix} = 0 \tag{3.101}$$

$$\mathbf{J} = \begin{bmatrix} \mathbf{\Omega}_1^H\mathbf{\Omega}_1 + \mathbf{A}_{sel} & \mathbf{\Omega}_1^H\mathbf{\Omega}_2 \\ \mathbf{\Omega}_2^H\mathbf{\Omega}_1 & \mathbf{\Omega}_2^H\mathbf{\Omega}_2 \end{bmatrix} \tag{3.102}$$

The algorithm is explained below:

- Set the $\mathbf{S^T}$ and $\mathbf{A}$ as the zero arrays.
- Solve the system of linear equations (3.96) in the LS sense
- Update (3.97) the θ variable.
- Check the tolerance and calculate objective function decrease. If tolerance is less, repeat from the step 2

3.3 The system model for frequency selective channel

In this section, we assume the GFDM MIMO system with frequency selective channel and with additive white Gaussian noise at the receiver (3.103)[Jan04] Explained model is written in the two forms: matrix and vectorized forms.We assume the zero guard interval before transmission block, and zero interference between different blocks. The main equation of the MIMO system with frequency selective channel is written as following [LV06] equation (3.103).

$$\mathbf{Y}_{:,k} = \sum_{l=0}^{L} \mathcal{H}_{:,:,l} \mathbf{X}_{:,k-l} + \mathbf{N}_{:,k} \tag{3.103}$$

$$\mathbf{Y} \in \mathbb{C}^{M_r \times T} \mathbf{N} \in \mathbb{C}^{M_r \times T} \mathbf{X} \in \mathbb{C}^{M_t \times T} \mathcal{H} \in \mathbb{C}^{M_r \times M_t \times L+1}$$

Where the $\mathbf{X}_{:,k}$ is the k-th column of the transmitted data block. The size of each column equal of the transmitted antenna size. The $\mathcal{H}$ is the third dimensional tensor and $\mathbf{H}_{:,:,i}$ is i-th slice in the third dimension from the tensor $\mathcal{H}$. The $\mathbf{N}_{:,k}$ is the column of the noise matrix We assume that noise has Gaussian distribution, is uncorrelated and has variance δ^2. This model consider the MIMO case and doesn't include any modulation of the data. The GFDM system also has modulation of the symbols, which we should consider in the investigation. We can write modulated data in the two forms, as the transmitted matrix(3.104) and as the vectorized notation(3.107). The model is the same as the GFDM MIMO flat frequency model but with changes in the matrix **A**.

$$\mathbf{X} = \mathbf{A} \cdot (\mathbf{C}^{[a]'T} \odot (\mathbf{S} \cdot (\mathbf{C}^{[b]} \diamond \mathbf{B})^T)) \tag{3.104}$$

$$\mathbf{X} \in \mathbb{C}^{M_t \times T} \tag{3.105}$$

$$\mathbf{A} = (\mathbf{I} \otimes \mathbf{1}_{1\times F}) \in \mathbb{C}^{M_t \times M_t F} \tag{3.106}$$

$$vec(\mathbf{X}^T) = \mathbf{\Upsilon}_1 \cdot vec(\mathbf{S}^T) \tag{3.107}$$

$$\mathbf{\Upsilon}_1 = \begin{bmatrix} (\mathbf{C}^{[a]'T} \diamond \mathbf{C}^{[b]T}) \diamond (\mathbf{A}_{1,:} \otimes \mathbf{B}) \\ (\mathbf{C}^{[a]'T} \diamond \mathbf{C}^{[b]T}) \diamond (\mathbf{A}_{2,:} \otimes \mathbf{B}) \\ \vdots \\ (\mathbf{C}^{[a]'T} \diamond \mathbf{C}^{[b]T}) \diamond (\mathbf{A}_{M_t,:} \otimes \mathbf{B}) \end{bmatrix} \tag{3.108}$$

$$vec(\mathbf{X}^T) \in \mathbb{C}^{M_t T \times 1} \mathbf{\Upsilon}_1 \in \mathbb{C}^{M_t T \times F M_t T_s} vec(\mathbf{S}^T) \in \mathbb{C}^{F M_t T_s \times 1} \tag{3.109}$$

Where the symbol matrix **S** is the same as in the GFDM MIMO flat frequency model and constructed from the third dimensional symbol tensor $\mathcal{S} \in \mathbb{C}^{F \times T_s \times M_t}$. Also it should be noted, we consider case with $T_s F = T$

$$\mathbf{S} = \begin{bmatrix} \mathcal{S}_{:,:,1} \\ \mathcal{S}_{:,:,2} \\ \vdots \\ \mathcal{S}_{:,:,M_t} \end{bmatrix} \tag{3.110}$$

$$\mathcal{S} \in \mathbb{C}^{F \times T_s \times M_t} \mathbf{S} \in \mathbb{C}^{F \cdot M_t \times T_s}$$

The $C^{[b]}$,$\mathbf{C}^{[a]'}$ matrices, $\mathbf{b}$ array stay the same as in the GFDM MIMO flat frequency model .The connection array stay the same as in the previous model.

$$\mathbf{C}^{[b]} \in \mathbb{C}^{T \times T_s}$$

$$\mathbf{b} = \mathbf{1}_{T_s \times 1} \in \mathbb{C}^{T_s \times 1} \tag{3.111}$$

$$\mathbf{C}^{[a]'} = (\mathbf{1}_{1 \times M_t} \otimes \mathbf{C}^{[a]}) \in \mathbb{C}^{T \times M_t \cdot F} \tag{3.112}$$

The $\mathbf{A}$ matrix changes in comparison with the flat-frequency model. In the original case $\mathbf{A}$ showed connection between receive and transmit antenna. Now it should show transmitted signal. Because in the model we separate each sub-carrier, the $\mathbf{A}$ must sum up all the sub-carriers for each transmit antenna. The following $\mathbf{A}$ from (3.113) will sum the sub-carriers for the transmit antennas.

$$\mathbf{A} = \mathbf{I}_{M_t} \otimes \mathbf{1}_{1 \times F} \in \mathbb{C}^{M_t \times F \times M_t} \tag{3.113}$$

The defined two models allow consider two equation of the frequency selective channels: in the classical (3.114) MIMO form and in the vectorized form. The classical form (3.114) is written as the sum of different matrix products. The classical form doesn't work in the channel estimation algorithm. The vectorized form (3.115) allow writing vectorized received data as the product of the modulation matrix and block Toeplitz convolution matrix. The structure of the matrix $\mathbf{H}$ is presented in the (3.116), where the $\mathbf{H_{x,y}}$ is the convolution matrix for the data which is transmitted from the y-th transmit antenna and received with x-th receive antenna (3.117). We assume that receiver knows the length of the channel and it equals to the $L+1$ time samples.

$$\mathbf{Y}_{:,k} = \sum_{l=0}^{L} \mathcal{H}_{:,:,l} \mathbf{X}_{:,k-l} + \mathbf{N}_{:,k} \tag{3.114}$$

$$\mathbf{Y} \in \mathbb{C}^{M_r \times T} \mathbf{N} \in \mathbb{C}^{M_r \times T} \mathbf{X} \in \mathbb{C}^{M_t \times T} \mathcal{H} \in \mathbb{C}^{M_r \times M_t \times L+1}$$

$$vec(\mathbf{Y}^T) = \mathbf{H} \cdot \mathbf{\Upsilon}_1 \cdot vec(\mathbf{S}^T) = \mathbf{H}vec(\mathbf{X}^T) \tag{3.115}$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{1,1} & \mathbf{H}_{1,2} & \cdots & \mathbf{H}_{1,M_t} \\ \mathbf{H}_{2,1} & \mathbf{H}_{2,2} & \cdots & \mathbf{H}_{2,M_t} \\ \vdots & & & \\ \mathbf{H}_{M_r,1} & \mathbf{H}_{M_r,2} & \cdots & \mathbf{H}_{M_r,M_t} \end{bmatrix} \tag{3.116}$$

$$\mathbf{H} \in \mathbb{C}^{M_r T \times M_t T}$$

$$\mathbf{H}_{x,y} = \begin{bmatrix} \mathcal{H}_{x,y,1} & 0 & 0 & \cdots & 0 \\ \mathcal{H}_{x,y,2} & \mathcal{H}_{x,y,1} & 0 & \cdots & 0 \\ \mathcal{H}_{x,y,3} & \mathcal{H}_{x,y,2} & \mathcal{H}_{x,y,1} & \cdots & 0 \\ \vdots & & & & \\ \mathcal{H}_{x,y,T} & \mathcal{H}_{x,y,T-1} & \mathcal{H}_{x,y,T-2} & \cdots & \mathcal{H}_{x,y,1} \end{bmatrix} \tag{3.117}$$

$$\mathbf{H}_{1,1} \in \mathbb{C}^{T \times T}$$

We can rewrite equation (3.117) with respect to the vectorized channel matrix. We can write the channel values in the following form (3.118)

$$\mathbf{h} = \begin{bmatrix} vec(\mathcal{H}_{(:,:,1)}) \\ vec(\mathcal{H}_{(:,:,2)}) \\ \vdots \\ vec(\mathcal{H}_{(:,:,L+1)}) \end{bmatrix} \tag{3.118}$$

$$\mathbf{h} \in \mathbb{C}^{(L+1)M_r M_t \times 1}$$

The equation allow writing vectorized received data with respect to the array $\mathbf{h}$.The matrix $\mathbf{D}_{mod}$ is constructed in the special way, similar to the simple convolution matrix. To show the construction process we define the transmitted signal (3.121). The construction is similar to the convolution matrix but for the matrix instead array. After convolution construction the Kronecker product is used to extend matrix for all receive antennas.

$$vec(\mathbf{Y}) = \mathbf{D}_{mod}\mathbf{h} \tag{3.119}$$

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,T} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,T} \\ \vdots & & & \\ x_{M_t,1} & x_{M_t,2} & \cdots & x_{M_t,T} \end{bmatrix} \tag{3.120}$$

$$\mathbf{D}_{mod} = \begin{bmatrix} x_{1,1} & \cdots & x_{M_t,1} & 0 & \cdots & 0 & \cdots & 0 \\ x_{1,2} & \cdots & x_{M_t,2} & x_{1,1} & \cdots & x_{M_t,1} & \cdots & 0 \\ x_{1,3} & \cdots & x_{M_t,3} & x_{1,2} & \cdots & x_{M_t,2} & \cdots & 0 \\ \vdots & & & & & & & \\ x_{1,T} & \cdots & x_{M_t,T} & x_{1,T-1} & \cdots & x_{M_t,T-1} & \cdots & x_{M_t,1} \end{bmatrix} \otimes \mathbf{I}_{M_r} \tag{3.121}$$

$$\mathbf{D}_{mod} \in \mathbb{C}^{TM_r \times (L+1)M_tM_r} \tag{3.122}$$

3.3.1 Least squares solution for the channel estimation

The equation with respect to the transmitted data allow us to construct the optimization problem for the channel vector in the second norm sense(3.123). To find the minimal residual point we should find the partial derivative of the function and equate it to zero. We used the Wirtinger calculus to find the partial derivative with respect to the $\mathbf{h}^*$ (3.124), because of the complex valued function. The residual function is convex, and there is only point where partial derivative equal to the zero and this point is global minimum[CJ13]. We must solve the system of linear equations to find the solution of equation (3.125). The least squares solution become (3.125).

$$\min_{\mathbf{h}}(vec(\mathbf{Y}) - \mathbf{D}_{mod}\mathbf{h})^H(vec(\mathbf{Y}) - \mathbf{D}_{mod}\mathbf{h}) = \min_{\mathbf{h}} \mathbf{r}_m^H \mathbf{r}_m \qquad (3.123)$$

$$\frac{\delta \mathbf{r}_m^H \mathbf{r}_m}{\delta \mathbf{h}^*} = -\mathbf{D}_{mod}(vec(\mathbf{Y}) - \mathbf{D}_{mod}\mathbf{h}) = 0 \qquad (3.124)$$

$$\mathbf{h}_{opt} = \mathbf{D}_{mod}^{+} vec(\mathbf{Y}) \qquad (3.125)$$

3.3.2 The semi-blind receiver for channel estimation

The receiver may find the unknown symbol with unknown channel matrix with additional constraints. The approach of the semi-blind receiver must base on the ALS algorithm, because of the different residual functions for each of the unknown array.The main equations for the solution become two residual functions: for the symbols(3.126) and for the channel coefficients(3.127).

$$\mathbf{r}_{ms} = vec(\mathbf{Y}^T) - \mathbf{H}\boldsymbol{\Upsilon}_1 vec(\mathbf{S}^T) \qquad (3.126)$$

$$\mathbf{r}_{mh} = vec(\mathbf{Y}^T) - \mathbf{D}_{mod}\mathbf{h} \qquad (3.127)$$

We considered three approaches of the semi-blind receiver:

- Simplified semi-blind receiver
- Semi-blind receiver with interference cancellation.
- Newton based semi-blind receiver.

The both of them is explained below. Also possible to use the Newton algorithm based approach as in the SISO model here, but to simplify the model and show performance of the both algorithm in the simulation was shown only ALS based algorithm.

3.3.3 Simplified semi-blind receiver

The simplified semi-blind receiver is constructed as the ALS algorithm with typical implementation. The additional information from the known symbols is also renewed in the iteration, but ignored in the update step. The algorithm is explained below:

- Set the $\mathbf{S^T}$ as the $\mathbf{S^T}_{known}$. The unknown values equated to zero
- Solve the system of linear equations (3.123) in the LS sense (3.128) and update the $\mathbf{h}$ variable.
- Solve the system of linear equations (3.123) in the LS sense (3.129) and update the $vec(\mathbf{S}^T)$ variable only for unknown symbols.
- Check the tolerance and calculate objective function decrease. If tolerance is less, repeat from the step 2

The equation for the variables update is shown in (3.128) and (3.129) where $\mathbf{S}_{sel}$ is the selection matrix for the unknown symbols matrix elements.

$$\mathbf{h}_{opt} = \mathbf{D}^{+}_{mod} vec(\mathbf{Y}) \tag{3.128}$$

$$vec(\mathbf{S}^T)_{opt} = (\mathbf{I} - \mathbf{S}_{sel}) \cdot vec(\mathbf{S}^T_{kn})\mathbf{S}_{sel} \cdot \boldsymbol{\Upsilon}_1^{+}\mathbf{H}^{+} vec(\mathbf{Y}) \tag{3.129}$$

$$\mathbf{S}_{sel} = \mathbf{I} - diag(vec(\mathbf{S}^T_{known}))^{+} diag(vec(\mathbf{S}^T_{known})) \in \mathbb{C}^{M_tFT_s \times M_tFT_s} \tag{3.130}$$

3.3.4 Semi-blind receiver with interference cancellation

The normal semi-blind receiver use one important property of the received data, the known and unknown symbols possible to separate as the sum between them. The transmitted data can be represented as sum of two components (3.131) known and unknown. The known and unknown symbol vectors are constructed via multiplication with selection matrix and possible to change the modulation matrix.

$$vec(\mathbf{X}^T) = \boldsymbol{\Upsilon}_1(vec(\mathbf{S}^T_{kn}) + vec(\mathbf{S}^T_{unk})) \tag{3.131}$$

$$vec(\mathbf{X}^T) = \mathbf{\Upsilon}_1 vec(\mathbf{S}_{kn}^T) + \mathbf{\Upsilon}_1 vec(\mathbf{S}_{unk}^T) \tag{3.132}$$

$$vec(\mathbf{Y}^T) = \mathbf{H}\mathbf{\Upsilon}_1 vec(\mathbf{S}_{kn}^T) + \mathbf{H}\mathbf{\Upsilon}_1 vec(\mathbf{S}_{unk}^T) + vec(\mathbf{E}^T) \tag{3.133}$$

The receiver can't separate symbols in time domain spatially, but can decrease the interference between known and unknown symbols to archive lower SNR. As the changed objective function for the symbols matrix become (3.134) function

$$\mathbf{r}_{mi} = vec(\mathbf{Y}^T) - \mathbf{H}\mathbf{\Upsilon}_1 vec(\mathbf{S}_{unk}^T) - \mathbf{H}\mathbf{\Upsilon}_1 vec(\mathbf{S}_{kn}^T) \tag{3.134}$$

$$vec(\mathbf{S}_{unk}^T)_{opt} = \mathbf{\Upsilon}_1^+ \mathbf{H}^+ (vec(\mathbf{Y}^T) - \mathbf{H}\mathbf{\Upsilon}_1 vec(\mathbf{S}_{kn}^T)) \tag{3.135}$$

$$\mathbf{h}_{opt} = \mathbf{D}_{mod}^+ vec(\mathbf{Y}) \tag{3.136}$$

The algorithm to semi-blind receiver is written as following:

- Set the $\mathbf{S^T}$ as the $\mathbf{S^T}_{kn}$. The unknown values equated to zero
- Solve the system of linear equations (3.134) in the LS sense (3.135) and update the $\mathbf{h}$ variable.
- Solve the system of linear equations (3.134) in the LS sense (3.136) and update the $vec(\mathbf{S}_{unk}^T)$.
- Check the tolerance and calculate objective function decrease. If tolerance is less, repeat from the step 2

3.3.4.1 Regularization for the semi-blind receiver

We can add the regularization term in the objective function of semi-blind receiver to make semi-blind receiver more stable in the noise environment. The objective function for the channel estimation will be written as following to previous part. The additional regularization term possible to write in the two notations: vectorized and convolution forms. In the paper [LV06] proposed two algorithms to set the matrix $\mathbf{V}$ for the regularization term:

- Subspace approach
- Smoothing least squares approach

We don't considered this approach n this thesis, but they will come to the thesis as the future work model to raise performance of the semi-blind receiver.

3.3.5 Newton based semi-blind receiver

The semi-blind receiver task can be solved via Newton based algorithm. For this objective function should be written in the certain way and we can see that the task can be solved in the similar way as in the SISO case.

$$\min_{vec(\mathbf{S}_{unk})\mathbf{h}} \mathbf{r}_{mi}^H \mathbf{r}_{mi} \tag{3.137}$$

$$\frac{\delta \mathbf{r}_{mi}^H \mathbf{r}_{mi}}{\delta \mathbf{h}^*} = -(\mathbf{D}_u + \mathbf{D}_k)^H (\mathbf{y} - \mathbf{D}_k \mathbf{h} - \mathbf{D}_u \mathbf{h}) = 0 \tag{3.138}$$

$$\frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta vec(\mathbf{S}_{unk})^*} = -(\mathbf{H}\boldsymbol{\Omega}_k)^H (\mathbf{y} - \mathbf{H}\boldsymbol{\Omega}_k vec(\mathbf{S}_{kn}) - \mathbf{H}\boldsymbol{\Omega}_u vec(\mathbf{S}_{unk})) = 0 \tag{3.139}$$

The objective function will be written as following(3.137). The equation $\mathbf{r}_s$ can be written in the two forms which are equal. The optimal point for the objective function will be point where partial derivative will be equal to the zero (3.138). This point is one because of the objective function convexity. We construct the partial derivative with respect to the unknown symbols and channel values. We used the Wirtinger calculus to find the derivative of the complex function (3.139). The following equations is equated to zero and we solved the system of non-linear equations. To solve the system we used as written above Newton algorithm.

$$\mathbf{J}\theta = -\mathbf{d} \tag{3.140}$$

$$\theta_{k+1} = \theta_k - \mathbf{J}^{+}\mathbf{d} \tag{3.141}$$

$$\mathbf{J} = \begin{bmatrix} \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta \mathbf{h}^* \mathbf{h}^*} & \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta vec(\mathbf{S}^*)\mathbf{h}} \\ \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta \mathbf{h}^* vec(\mathbf{S})} & \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta vec(\mathbf{S}^*) vec(\mathbf{S})} \end{bmatrix} \tag{3.142}$$

$$\mathbf{d} = \begin{bmatrix} \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta \mathbf{h}^*} \\ \frac{\delta \mathbf{r}_s^H \mathbf{r}_s}{\delta vec \mathbf{S}^*} \end{bmatrix} \tag{3.143}$$

$$\mathbf{d} = \begin{bmatrix} -(\mathbf{D}_u + \mathbf{D}_k)^H (\mathbf{y} - \mathbf{D}_k \mathbf{h} - \mathbf{D}_u \mathbf{h}) \\ -(\mathbf{H}\boldsymbol{\Omega}_k)^H (\mathbf{y} - \mathbf{H}\boldsymbol{\Omega}_k vec(\mathbf{S}_{kn}) - \mathbf{H}\boldsymbol{\Omega}_u vec(\mathbf{S}_{unk})) \end{bmatrix} \tag{3.144}$$

$$\mathbf{J} = \begin{bmatrix} (\mathbf{D}_k + \mathbf{D}_u)^H (\mathbf{D}_k + \mathbf{D}_u) & (\mathbf{D}_k + \mathbf{D}_u)^H \mathbf{H}\boldsymbol{\Omega}_k \\ (\mathbf{H}\boldsymbol{\Omega}_k)^H (\mathbf{D}_k + \mathbf{D}_u) & (\mathbf{H}\boldsymbol{\Omega}_k)^H \mathbf{H}\boldsymbol{\Omega}_k \end{bmatrix} \tag{3.145}$$

$$\theta = \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix} \tag{3.146}$$

The Newton algorithm calculate at each iteration solution for the system of linear equations which constructed at the each iteration point [TA79]. We must write the equation of the Jacobian matrix (3.142) for the partial derivatives to determine the solved at each iteration equation. To write the Jacobian matrix we use property of the equality of the different notations of the same function in the same point. The resulting Jacobian matrix is written as follows (3.145).Unfortunately the Newton algorithm in this statement of the problem is significantly unstable. The typical conditional number of the Jacobian matrix higher than 100. Higher conditional number increase the error in the each iteration. To overcome this problem we propose the simple approach to regularize the problem at each iteration. The explanation of the regularization is written with the Newton algorithm and is explained below. Regularization allow to make algorithm more stable and converge them even if the Jacobian is very ill-conditioned.

- Set the variable θ in the certain way (3.146).
- Calculate the Jacobian matrix and derivative array in the given point θ
- Set the $\lambda = 1$.
- Calculate $\mathbf{J} = \mathbf{J} + \mathbf{I}\lambda$.
- $\lambda = \lambda * 2$.
- Check the Jacobian matrix condition number. If condition number is higher than 15 go to the step 4.
- Solve the system of linear equations (3.140) with respect to the θ
- Renew the θ with (3.141) equation.
- If the residual decrease is higher than tolerance repeat process from the step 2.

3.4 Simulation results

In this section we present simulation results for the GFDM MIMO system and algorithms which was reviewed in the chapter 3.

3.4.1 Scenario 2.1

The performance results of the $\mathcal{A}$ search algorithm are obtained through simulation. The parameters of the system are tabulated in Table 3.1. The GFDM system is simulated in the AWGN channel without coding. QPSK modulation scheme is used. The number of sub-carriers was $F = 16$, samples for each symbol was $T/T_s = F$ The block size was $T_s = 5$. The root-raised cosine filter was used with roll-off factor $\alpha = 0.5$. The selection coefficients was chosen mode two products between identity matrix in the first and third dimension with random integer values array. The results of the GFDM performance is presented in the two plots, at first is shown the reconstruction error for the $\mathbf{A}$ fig. 3.1, at second is shown the SER in comparison with the case if $\mathbf{a} = \mathbf{1}$ with the zero-forced receiver fig. 3.2.

Table 3.1: GFDM simulation parameters.Scenario 2.1

Parameter	Variable	GFDM
Transmit antennas	M_t	2
Receive antennas	M_r	2
Modulation scheme	μ	QPSK
Samples per symbol	T/T_s	16
Sub-carriers	F	16
Block size	T_s	5
Filter type		RRC
Roll-off-factor	α	0.5
Sub-carrier coefficients	$\mathbf{a}_i$	$unif(0,1)$
Channel	h	Flat-fading
Cyclic Prefix		No
Transmission		Uncoded
Decomposition rank	$rank$	25

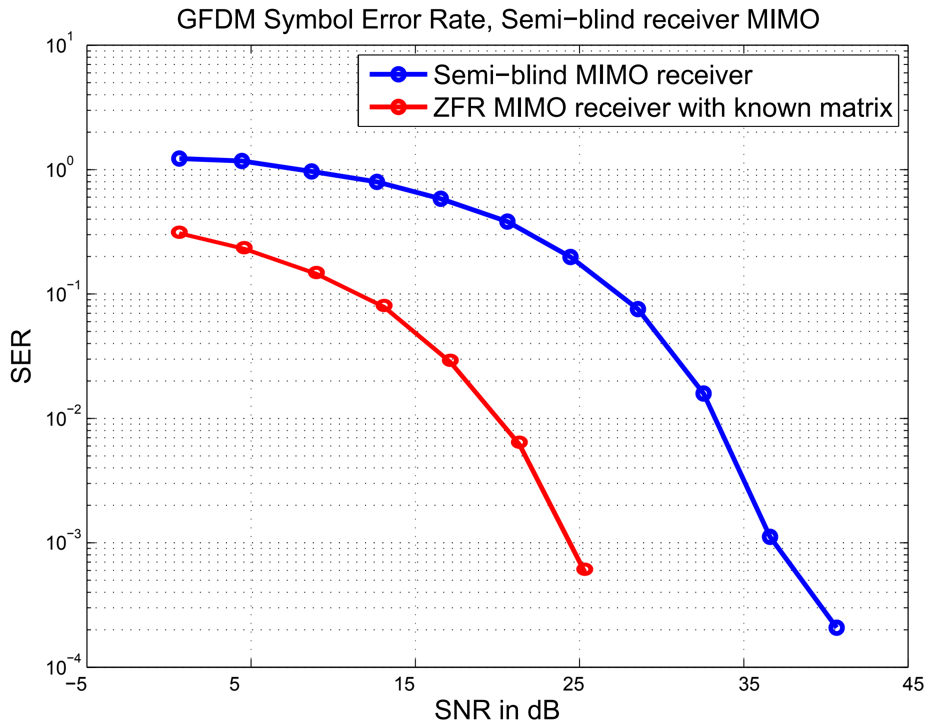

Figure 3.1: *SER dependency for the semi-blind MIMO receiver and ZF receiver with known channel*

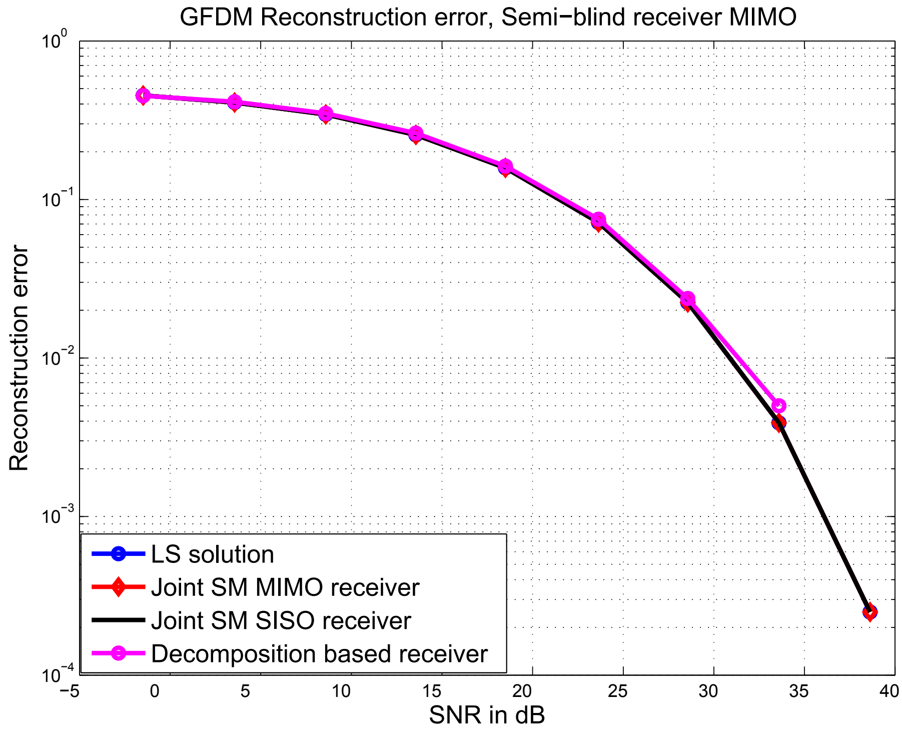

Figure 3.2: *Reconstruction error for the sub-carrier estimation approaches*

3.4.2 Scenario 2.2

Table 3.2: GFDM simulation parameters.Scenario 2.2

Parameter	Variable	GFDM
Modulation scheme	μ	QPSK
Samples per symbol	T/T_s	16
Sub-carriers	F	16
Block size	T_s	5
Transmit antennas	M_t	2
Receive antennas	M_r	2
Filter type		RRC
Roll-off-factor	α	1
Length of the channel	$L+1$	12
Channel	h	Ped-A
Cyclic Prefix		No
Transmission		Uncoded

The GFDM system semi-blind receiver in the MIMO system is simulated in the frequency selective channel without coding. QPSK modulation scheme is used. The channel was generated in the two steps. At the first step was generated flat frequency channel matrix $\mathbf{H}$ with zero-mean circularly symmetric complex Gaussian random values. At the second step for each values of the matrix $\mathbf{H}$ generated the tensor $\mathcal{H}$ with time filter coefficients for each transmission link. At the end each slice of the tensor $\mathcal{H}$ is element-wise multiplied with $\mathbf{H}$. The number of the receive and transmit antennas was $M_r = M_t = 3$. The number of sub-carriers was $F = 8$, samples for each symbol was $T/T_s = F$ The block size was $T_s = 3$. The root-raised cosine filter was used with roll-of-factor $\alpha = 1$. The length of the channel equal to $L + 1 = 4$. The experiment setup is presented in the table 3.2. The results of the GFDM performance is presented in the two plots, at first (3.3) is shown the SER in comparison with the case if receiver know the original channel and use the Zero-Forced receiver and with case for different number of unknown symbols in one transmission block. At second figure (3.4) is shown the reconstruction error for the channel values (3.147).

$$R_e = \frac{|| \mathbf{A} - \hat{\mathbf{A}} ||^2}{|| \mathbf{A} ||^{@}} \tag{3.147}$$

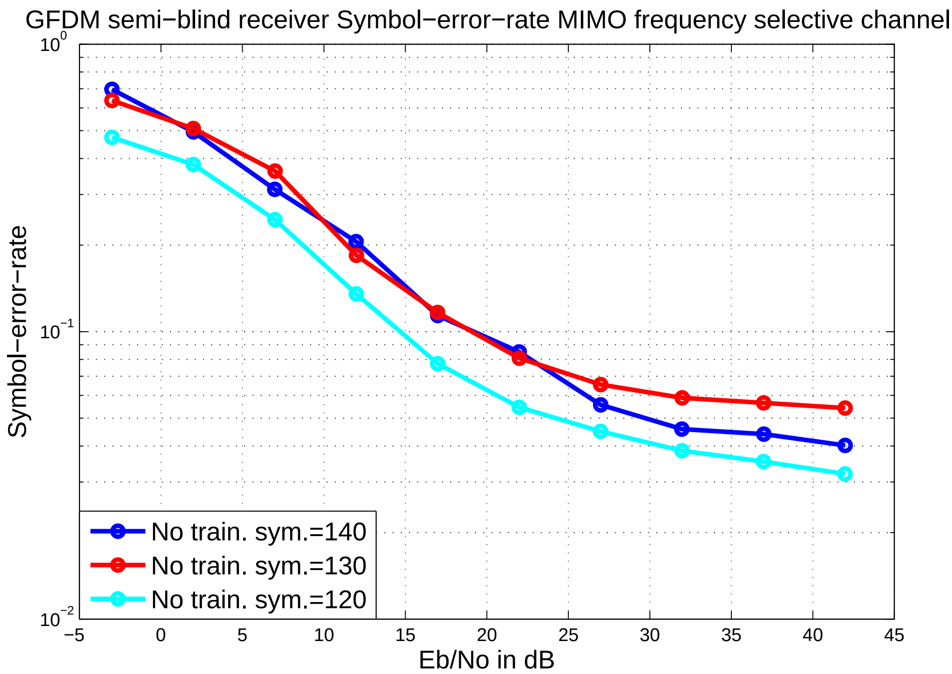

Figure 3.3: *SER for the semi-blind MIMO receiver with different number of known symbols*

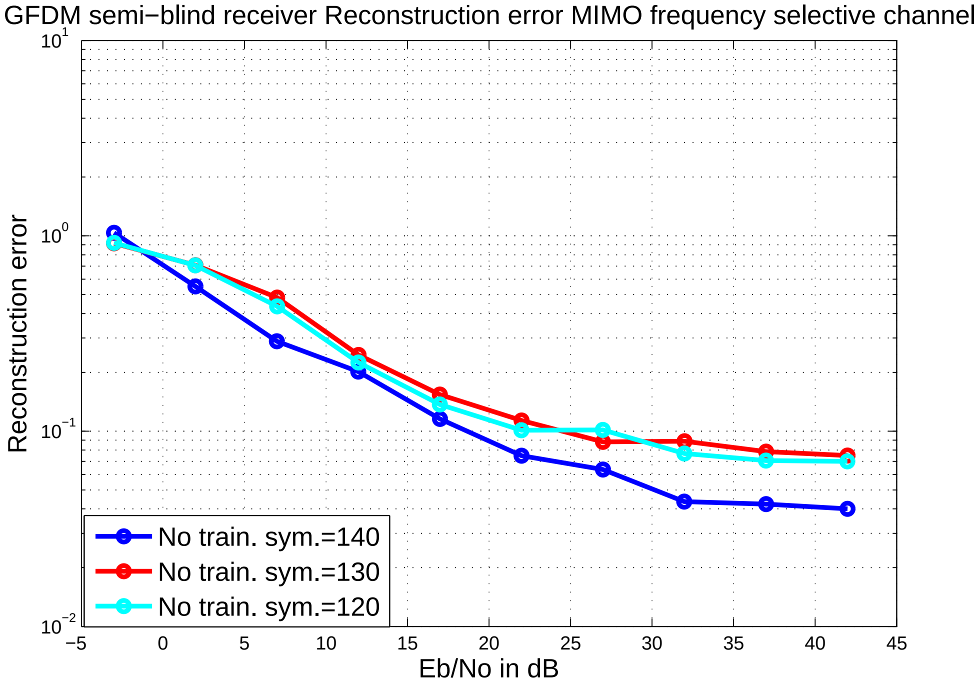

Figure 3.4: *Channel reconstruction error for the semi-blind MIMO receiver with different number of known symbols*

3.4.3 Scenario 2.3

The GFDM system Newton based semi-blind receiver in the MIMO system is simulated in the frequency selective channel without coding. QPSK modulation scheme is used. The channel was generated in the two steps. At the first step was generated flat frequency channel matrix $\mathbf{H}$ with zero-mean circularly symmetric complex Gaussian random values. At the second step for each values of the matrix $\mathbf{H}$ generated the tensor $\mathcal{H}$ with time filter coefficients for each transmission link. At the end each slice of the tensor $\mathcal{H}$ is element-wise multiplied with $\mathbf{H}$. The number of the receive and transmit antennas was $M_r = M_t = 3$. The number of sub-carriers was $F = 8$, samples for each symbol was $T/T_s = F$ The block size was $T_s = 3$. The root-raised cosine filter was used with roll-of-factor $\alpha = 1$. The length of the channel equal to $L + 1 = 4$. The experiment setup is presented in the table 3.2. The results of the GFDM performance is presented in the two plots, at first (3.5) is shown the SER in comparison with the case if receiver know the original channel and use the Zero-Forced receiver and with case for different number of unknown symbols in one transmission block. At second figure (3.6) is shown the reconstruction error for the channel values. The reconstruction error is defined in the (3.148)

$$R_e = \frac{\| \mathbf{h} - \hat{\mathbf{h}} \|^2}{\| \mathbf{h} \|^{@}} \qquad (3.148)$$

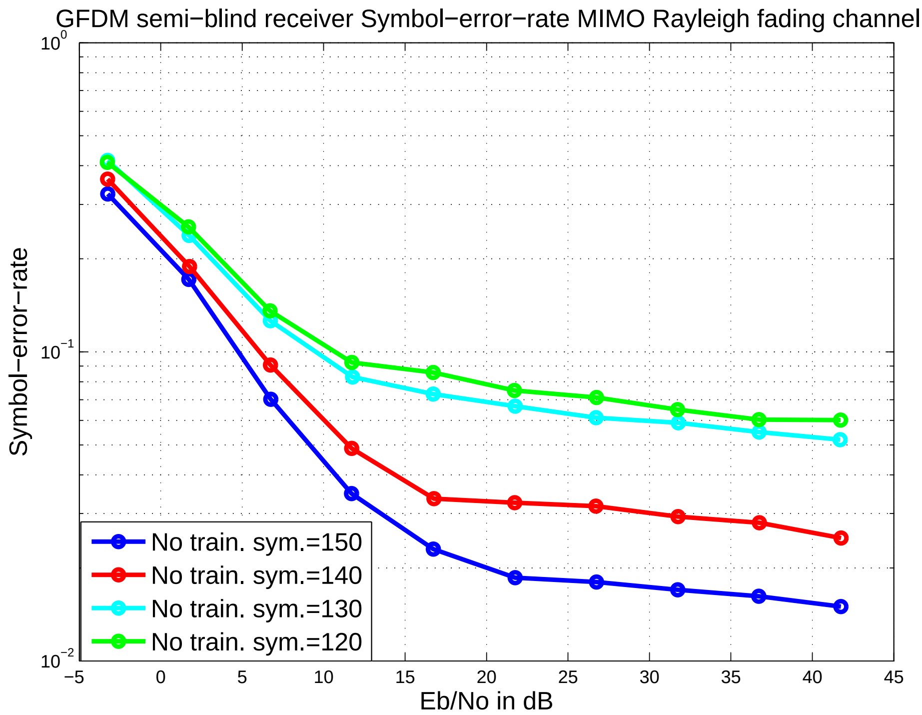

Figure 3.5: *SER for the semi-blind MIMO receiver with different number of known symbols*

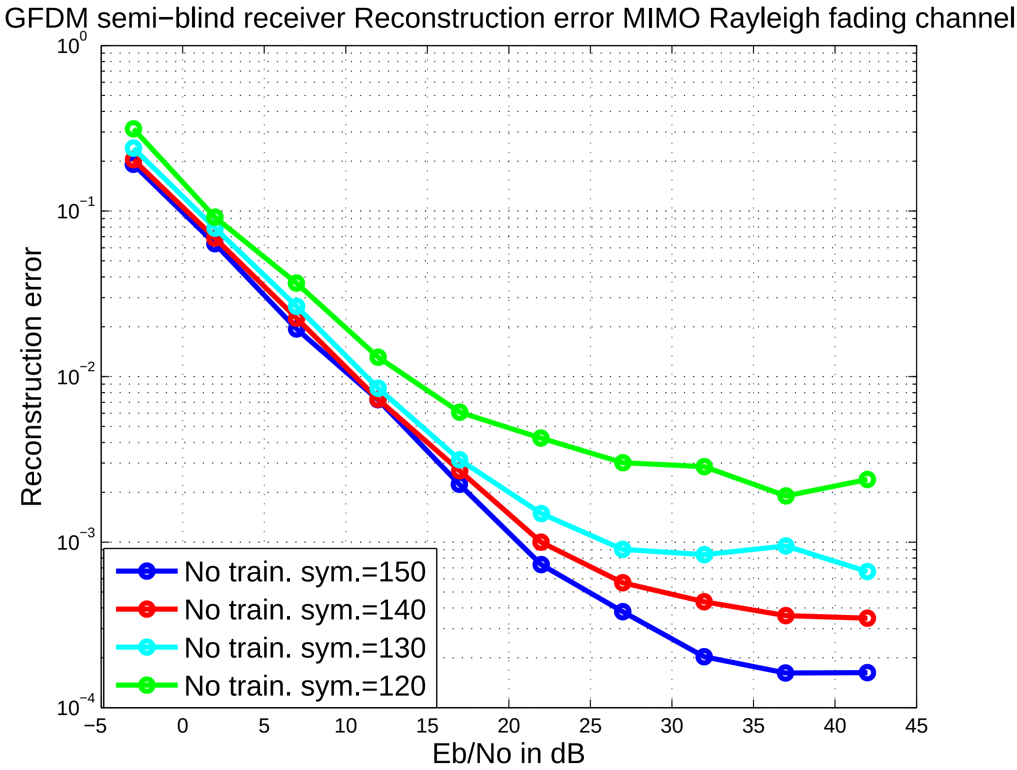

Figure 3.6: *Channel reconstruction error for the semi-blind MIMO receiver with different number of known symbols*

3.5 Conclusion

As conclusion for the second experiment with frequency coefficient selection approach in the MIMO case, we can see that algorithm decrease performance for 5-6 dB in comparison with system where the frequency coefficients known fig. 3.2. Decrease in performance come from the non-convenient coefficient selection way. The coefficients were chosen as 1 if the resulting $abs(a_1)$ value is higher that 0.5. There are better solutions to increase performance of the algorithm with the predefined information at the receiver.

We explained three frequency coefficients estimation approaches. As we can see from the result at the fig.3.3 all of them has the same result. Even the approximated solution based on the SISO model show the same performance. Due to that fact there is no difference with algorithm to choose. It should be noted that the joint MIMO algorithm allow to also the symbol estimation. Other algorithms can't estimate the symbols in the received data and work only if transmission block equal to the number of symbols in each subcarrier. There is one additional disadvantage of the Decomposition based approach, which doesn't converge with some initial points. Due to that fact decomposition based approach is less reliable than other algorithms. The overall conclusion over the frequency selection coefficient algorithms we can say that spectrum sensing approach is applicable in the MIMO case too. The system may use different subcarrier antennas for each transmit antenna and the receiver can estimate them. The performance loss is similar to the SISO case, but number of the time-slots decrease significantly in comparison with the SISO case. In the paper is explained very simple approaches to coefficients estimation. Performance of the algorithm can be implemented in more efficient way without huge performance loss. We doesn't use any pre-processing and post-processing steps to increase performance.

Next experiment is the semi-blind receiver channel and the symbol estimation over frequency selective channel with Rayleigh fading. The pedestrian-A channel was assumed in the each of the antennas. The experiment was done for the number of possible known symbols which have solution for the problem statement. The result for the SER if shown in the fig. 3.3. We can see from the plot, that performance of the algorithm is weak. With high SNR the algorithm has only 10^{-2} SER, which is very high value. This behavior is explained with the very bad conditioning of the channel matrix. The additional fact is the low performance of the Rayleigh fading channel itself. We can implement the more convenient ways of the algorithm to avoid such low performance. There is additional investigation about typical real channel should be done. We can see from the fig. the channel estimation has the upper bound of precision. We can say that algorithm has the same performance from the 30 dB. There is significant dependency from the channel estimation precision as we can see here. The upper bound of the channel estimation doesn't allow increasing performance with higher SNR. As conclusion for the ALS based semi-blind receiver we can say that algorithm isn't provide full performance and they pre-processing and post-processing steps should be applied to increase the performance. This behavior will be investigated in the future work. The last experiment show performance of the Newton based semi-blind receiver over the frequency selective Rayleigh fading channel. The fig.(3.5) show the performance of the algorithm in the symbol estimation. As we can see here the algorithm has the upper bound in the performance and doesn't allow to decrease the SER lower than 10^{-2}. This behavior doesn't connect with error in the channel estimation, because the error in the channel estimation is upper bounded with much higher SNR and have precision near to the 10^{-3}/ The main reason for this behavior is the high condition number of the Jacobian matrix. To implement the more precise and regularized solution the BFGS of Levenberg-Marquardt algorithm must be applied. We can see from the results, that there is significant dependency between the number of the training symbols and performance of the system. The number of the training symbols is high due to the high number of estimated channel variables.The fig.(3.6) show the performance of the algorithm in the channel estimation. We can see that precision of the algorithm is high and algorithm estimates the channel with precision up to 10^{-3} reconstruction error. There are disadvantages of the algorithm which connected with bad conditioning and the computational complexity. The additional approaches must be investigated to overcome them. As conclusion for this algorithm we can propose two statements:

- Performance of the algorithm weak for the symbol estimation
- Performance of the algorithm applicable for the channel estimation
- The algorithm must use the regularizing approach.

4 Summary and future work

In this work generalized frequency division multiplexing was discussed and investigated. In the first part is concerned the introduction into the multiplexing scheme and into the tensor based processing. We show the main equations of the tensor based processing in the second section. We discussed the PARATUCK2 model, which become the main model for the transmitted signal. In addition, the PARATUCK2 model allow to simplify the modulation matrix as the Khatri-Rao product between two matrices, where the first is the Fourier transform matrix and the second is time filter matrix for symbols in block. There is one similar model of data explanation, which called wavelets. In this part was considered the main filters, which are used in the GFDM system. There is strong influence on the system performance from the $roll-off$ factor of the time filter. This influence is shown in the simulation results from second part. We considered the raised cosine and root-raised cosine filters.

In the second part, we considered the GFDM system with one transmit and one receive antenna. We modelled the transmitted signal with the PARATUCK2 model in the vectorized form. We considered the two approached in the GFDM receiver side: subcarrier coefficients estimation approach and channel estimation approach. In the first approach we assumed that transmitter turn off number of subcarriers and doesn't transmit data from them. The receiver knows the first symbol on each subcarrier and try to estimate which subcarrier is turned on. We developed algorithm which estimates the symbols and make decision about turned on coefficients. Algorithm allow using the spectrum sensing in the automatic way at the receiver without predefined subcarriers. The second approach we consider in the model with channel. We assumed that there is frequency selective channel between receiver and transmitter. We developed algorithm, which allow to estimate the channel and the symbols in one algorithm. This algorithm called semi-blind receiver. We have made simulations for the all of this algorithms. The simulations show, that results have good performance. There is one disadvantage of the optimization based algorithms, all of them are computationally expensive. The developed algorithms have the same disadvantage. The disadvantage become significant, if the transmission block size become large. The matrix inversion calculation become the hard task. There is additional iterative approaches so called ISTA" and "FISTA" which allow decreasing computational price. The semi-blind receiver outperforms the FFT receiver with known channel. This result shows the high performance of the semi-blind receiver. In the last part we extended the model for the MIMO case. The high number of the antennas make the model very complex for modeling. We developed the subcarrier selection approach for the MIMO. We assumed the same situation as in the SISO case, but used two predefined options: the overall transmission block is sent for subcarrier analysis, and the semi-blind approach. We developed both algorithms and tested results via simulations. The algorithm allow to estimate the turned on subcarriers automatically without information from the transmitter. The performance of the algorithms is lower than in known case. The performance decrease is explained with the huge number of estimated coefficients. The next algorithm which we assumed is frequency selective channel estimation for the MIMO system. We assumed Rayleigh fading channel with frequency selection for each transmitter-receiver pair. We developed two algorithms based on the Newton optimization and ALS based optimization algorithm.We have made simulations for the all of theese algorithms. The simulations don't allow making prediction about performance of the algorithm, due to the complex model of the receiver part. There is one disadvantage of the optimization based algorithms, all of them are computationally expensive. The developed algorithms have this disadvantage and the computational complexity become the very strong problem because the MIMO model increase amount of data significantly. The deeper analysis necessary for the optimization algorithms overview. As the conclusion over all thesis we can say that optimization algorithm allow increasing performance of the system and estimate the symbols in more precise way. The more precise estimation of the MIMO frequency selective model will be done in the future work. In the future work we will consider the different channel types in the MIMO model and ways to explain the physical channel. The multiplication between the Rayleigh fading and frequency selective channel is quite complex and must be simplified.

Bibliography

[AB00] Andersson, C. A. ; Bro, R.: The N-way Toolbox for MATLAB. In: *Chemometrics & Intelligent Laboratory Systems. 52 (1):1-4, 2000.* (2000). – http://www.models.life.ku.dk/source/nwaytoolbox/

[ADKM+15] Al-Dweik, Arafat ; Kalbat, Fatma ; Muhaidat, Sami ; Filio, Oscar ; Ali, Shirook M.: Robust MIMO-OFDM System for Frequency-Selective Mobile Wireless Channels. In: *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, VOL. 64, NO. 5,* (MAY 2015)

[AFX13] Almeida, André L. F. ; Favier, Gérard ; Ximenes, Leandro R.: Space-Time-Frequency (STF) MIMO Communication Systems With Blind Receiver Based on a Generalized PARATUCK2 Model. In: *IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 61, NO. 8,* (APR, 2013)

[AMDG13] Alves, Bruno M. ; Mendes, L.L. ; D.A.Guimaraes ; Gaspar, I.S.: Performance GFDM over frequency selective channels. In: *Revista Telecemunicationes vol 15* (Dec 2013)

[BSG97] Bro, R. ; Sidiropoulos, N. ; Giannakis, G.: PARAFAC: Tutorial and applications. In: *Chemometrics Group, Food Technology, Royal Veterinary & Agricultural University* (1997)

[BSG99] Bro, R. ; Sidiropoulos, N. ; Giannakis, G.: A fast least squares algorithm for separating trilinear mixtures. In: *ICA99- Int. Workshop on Independent Component Analysis and Blind Separation* (1999)

[CJ13] Comon, Pierre ; Jutten, Christian ; Comon, P. (Hrsg.) ; Jutten, C. (Hrsg.): *Handbook of Blind Source Separation*. Academic Press is an imprint of Elsevier, 2013

[Cox08] Cox, Donald: Fundamental limitations on increasing data rate in wireless systems. In: *IEEE Communacation magazine* (2008)

[DES82] Dembo, R.S ; Einsenstat, S.C. ; Steihaug, T.: Inexact Newton methods. In: *SIAM J. Sci. Comput., 19(1), pp. 400408,* (1982)

[Deu04] Deufelhard, P.: Newton methods for nonlinear problems: affne invariance and adaptive algorithms. In: *Springer* (2004)

[DMLF12] Datta, R. ; Michailow, N. ; Lentmaier, M. ; Fettweis, G.: GFDM Interference Cancellation for Flexible Cognitive Radio PHY Design. In: *Vehicular Technology Conference (VTC Fall), 2012 IEEE* (2012)

[DS83] Dennis, J.E. ; Schanabel, JR; R.: Numerical methods of unconstrained optimization and nonlinear equations. In: *Prentice-Hall,Englewood-Cli s* (1983)

[FKB09] Fettweis, Gerhard ; Krondorf, Marco ; Bittner, Steffen: GFDM - Generalized Frequency Division Multiplexing. In: *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th* (2009)

[Fle00] Fletcher, R.: Practical methods of optimization. In: *Wiley* (2000)

[FMD16] Farhang, Arman ; Marchetti, Nicola ; Doyle, Linda E.: Low-Complexity Modem Design for GFDM. In: *IEEE Transactions on Signal Processing* (2016)

[Gar05] Garin, Lionel J.: The “Shaping Correlator”, Novel Multipath Mitigation Technique Applicable to GALILEO BOC(1,1) Modulation Waveforms in High Volume Markets. In: *The European Navigation Conference GNSS 2005* (2005)

[HJ07] Horn, R. A. ; Johnson:, C. R.: Topics in matrix analysis. In: *Cambridge Univ. Press* (2007)

[Jan04] Jankiraman, Mohinder.: *Space-Time-Codes and MIMO systems.* Artech House universal personal communications series), 2004

[KB09] Kolda, Tamara G. ; Bader, Brett W.: Tensor Decompositions and Applications. In: *SIAM Review* 51 (2009), Nr. 3, S. 455–500

[Kel03] Kelley:, C. T.: Solving nonlinear equations with Newton methods. In: *SIAM, Philadelphia* (2003)

[LMV00] Lathauwer, Lieven de ; Moor, Bart de ; Vanderwalle, Joos: A multilinear singular value decomposition. In: *SIAM J. MATRIX ANAL. APPL. Vol. 21, No. 4, pp. 1253–1278* (2000)

[LT06] LIU, SHUANGZHE ; TRENKLER, GÄOTZ: HADAMARD, KHATRI-RAO, KRONECKER AND OTHER MATRIX PRODUCTS. In: *INTERNATIONAL JOURNAL OF INFORMATION AND SYSTEMS SCIENCES* (2006)

[LV06] LEUS, GEERT ; VEEN, ALLE-JAN VAN D.: Channel Estimation. In: *Delft University of Technology, Dept. EEMCS, Mekelweg 4, 2628 CD Delft, The Netherlands* (2006)

[MB06] Michele, Heather ; Bush, Clyburn: KHATRI-RAO PRODUCTS AND CONDITIONS FOR THE UNIQUENESS OF PARAFAC SOLUTIONS FOR IxJxK ARRAYS. In: *University of Kentucky* (2006)

[MMF14] Matthe, M. ; Michailow, N. ; Fettweis, G.: Influense of Pulse shaping on bit-error rate Performance and out of band radiation of GFDM. In: *ICC 14 WS 5G* (2014)

[MMI14] Matthe, M. ; Michailow, N. ; I.Gaspar: GFDM for 5G cellular networks. In: *IEEE transactions on commenications vol 62* (2014)

[MMM+15] Matthé, Maximilian ; Mendes, Luciano L. ; Michailow, Nicola ; Zhang, Dan ; Fettweis, Gerhard: Widely Linear Estimation for Space-Time-Coded GFDM in Low-Latency Applications. In: *IEEE Transactions on Communications* (2015)

[PP12] Petersen, Kaare B. ; Pedersen, Michael S.: *The Matrix Cookbook*. Petersen & Pedersen,, 2012

[RL03] R, Bro ; L., Kiers H. A.: A new efficient method for determining the number of components in parafac models. In: *J. Chemometrics, 17: 274–286 doi: 10.1002/cem.801* (2003)

[Skl01] Sklar, Bernard: Digital Communcations Fundamental and applications. In: *Prentice Hall 2 edition* (2001)

[Ste14] Steinwandt, Jens: Tutorial on Convex Optimization for Engineers. In: *Communications Research Laboratory* (2014)

[TA79] Tihonov, A.N. ; Asenin, V. Y.: Ways to solve Ill-posed tasks. In: *Nauka* (1979)

[VJ14] Vilaipornsawai, Usa ; Jia, Ming: Scattered-pilot channel estimation for GFDM. In: *2014 IEEE Wireless Communications and Networking Conference (WCNC)* (2014)

[ZK06] ZHOUR, ZEYAD A. ; KILICMAN, ADEM: MATRIX EQUALITIES AND INEQUALITIES INVOLVING KHATRI-RAO AND TRACY-SINGH SUMS. In: *Journal of Inequalities in Pure and Applied Mathematics* (2006)

List of Figures

List of Tables

Thesen

1. With tensor based PARATUCK2 model we can simplify the modulation matrix for the GFDM model.
2. The spectrum sensing can work can work without predefining at the receiver side which subcarrier are used.
3. The channel estimation is done without cshannel prefix. The ALS and Newton based algorithms allow to implement the semi-blind receiver inside of the transmission block.
4. The PARATUCK2 model also defines the MIMO model and all algorithms may be extended in the MIMO space.

Ilmenau, den 31. 07. 2016 Bulat Valeev

Erklärung

Die vorliegende Arbeit habe ich selbstständig ohne Benutzung anderer als der angegebenen Quellen angefertigt. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Quellen entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer oder anderer Prüfungen noch nicht vorgelegt worden.

Ilmenau, den 31. 08. 2016 Bulat Valeev