Master's Thesis

Frequency Offset Estimation for Multiuser Systems Based on a Low Dimensional Approximation of the Likelihood Function

Juan Carlos Bucheli Garcia

Department of Electrical and Information Technology,
Faculty of Engineering, LTH, Lund University, 2016.

Frequency Offset Estimation for Multiuser Systems Based on a Low Dimensional Approximation of the Likelihood Function

Juan Carlos Bucheli Garcia
wir14jbu@student.lu.se

Department of Electrical and Information Technology
Lund University

Advisor: Ove Edfors

Examiner: Fredrik Rusek

May 20, 2016

Printed in Sweden
E-huset, Lund, 2016

Abstract

This work corresponds to the theoretical foundations and practical proof of the master's degree project in frequency offset estimation for multiuser systems based on a low dimensional approximation of the likelihood function.

The project aims at the development of a low complexity estimator which can be relied upon. This way, higher frequency offset estimation rates can be achieved allowing for the use of less stable oscillators. Particularly, it is intended for the estimation of fractional (with respect to the OFDM sub-carrier spacing) frequency offsets.

The document is structured in a way it is expected to be understood by any reader with a background in Communications Engineering. Problem formulation and a review of the theoretical concepts used here are given at the beginning. Afterwards, the benchmark of the project (the ML estimator) is explained in detail with its main advantages and drawbacks. Later, a low dimension approximation in order to cope with ML's weaknesses is illustrated. Finally, a simulator is developed giving the outcomes required in order to be able to conclude about the proposed method.

Conclusions and future work are of relevance as there is a practical motivation for this project coming from the industry. Hence, further modifications in order to improve the results even more are envisaged. These were not developed thoroughly due to time constraint reasons in the development of the project but leave an open door for future research.

Acknowledgments

This project is the result of several months of work as one of the requirements to obtain my Master of Science degree. It would have not been possible without these two years of studies and experiences at Lund University and particularly the department of Electrical and Information Technology (EIT), which I am totally grateful to. Having this in mind, I wanna thank specially professors Ove Edfors, Fredik Rusek and PhD student Joao Vieira, all from EIT.

To begin with, I want to thank Ove for his support in this project and, since Spring 2015, when he allowed me to get the feeling of what it is to work in a huge project as the Massive MIMO project being carried out in the department. Apart from learning about that cutting edge technology, such an opportunity also helped me grow self confidence about how much I hope I will be able to give some day to research.

Likewise, I want to give special thanks to Fredrik Rusek for his invaluable support during all this process. Considering my work is just an extension of his, and that basically without his continuous supervision and guidance this would not have been possible, I think the least I can do is let him know how grateful I am. I really admire him and I hope some day I get to be as good as he is at what he does.

In addition, I want to thank Joao Vieira for trusting his 'gut' instinct since the first time he gave a good recommendation of me in EIT. I really think because of people like him, who selflessly support others who might really use some help and trust, is that you can expect this greedy and competitive world to be better.

Finally, but not less important, I wanna thank my friends and family for their support all along this process. Specially, my parents who blindly trust me and to whom I basically owe everything I am and everything I will ever be.

Table of Contents

List of Figures

Chapter 1
Introduction

One of the most important sub-components of any wireless device is its electronic oscillator. Due to the cascaded architecture of receivers, almost every performance measure in the communication chain has a dependency on the frequency synchronicity of the receiver's oscillator with respect to the transmitter node. In OFDM, for example, frequency synchronicity is important as it assures orthogonality among sub-carriers to avoid inter-carrier interference [1].

In particular, quasi-stable oscillators can be assumed to depend on temperature or other slow processes. Unfortunately, in practice oscillators, and especially cheap ones, are not steady enough to be considered robust by themselves. Therefore, a closed-loop control system based on estimation and compensation of frequency offset at the receiver should be relied upon.

The current project is based on the industry requirement of decreasing fabrication cost for wireless enabled devices. It aims at the study, design and simulation of a low complexity frequency offset estimation algorithm. More specifically, we are interested in the scenario of multiple user systems where orthogonal frequency division multiple access (OFDMA) allows for simultaneous use of the channel. An approach by using the structure of the signal due to the cyclic prefix on single user systems is found in [2] mainly for AWGN channels. On the other hand, some different techniques in order reduce the effect of inter-carrier interference due to frequency offset in multiuser systems are discussed in [3].

As the requirement is to be able to give a fast estimate of the frequency offset, based on existing work on a low complexity maximum likelihood (ML) frequency offset estimator for the single user scenario [4], this project extends such an approximation to the multiple user case. Specifically, this allows for the control of the computational complexity of the ML estimator at the expense of reducing estimation performance.

Some of the issues considered in the project are the complexity and the performance of the estimator when the number of user deviates from one. Particularly, the main reason why computational complexity plays an important role is that the rate at which the process should be repeated is generally high. Also, as this process is carried out at the receiver, it should be able not only to be implemented on a resource-constrained platform, but also in as brief time as possible.

At the end, the root mean square (RMS) estimation error via Monte-Carlo simulation was the main measure used to draw comparisons between performance of the studied and the proposed method in this project.

Chapter 2

Problem Formulation

The objective of this project is the assessment, design and evaluation of a multiple user frequency offset estimator to be carried out on communication receiver ends. Taking into account several performance and complexity considerations, and based on earlier work in the field for estimation of a single frequency offset [4], a maximum likelihood estimator is envisaged.

Due to the multi-user characteristic of the problem at hand, some complex operations are not possible to be avoided compared to the single user case. Precisely, big matrix inversion and determinant computations are now required. Furthermore, the increase of the number of degrees of freedom in the optimization problem required by maximum likelihood estimation becomes a bigger challenge as multidimensional optimization algorithms are now needed.

As it will be clear in Section 4, it is required to know the following in order to be able to give an estimate through the presented solution:

- System parameters such as the pilot symbols and their distribution for each user.
- Environment estimates such as the power delay profile of the channel and background noise power.
- One received OFDM symbol.

We then aim for the development of a low dimension approximation to a multiple user joint frequency offset maximum likelihood estimator in order to be able to control the performance-complexity trade-off.

2.1 Communication System Model

In order to carry out simulation of the mentioned algorithm, a general enough system model must be given. Particularly, different assumptions and choices regarding channel parameters and communication system considerations (as modulation technique among others) should be made. In light of this, the whole system model and channel features are, for the sake of analysis and numerical evaluation, described below:

- Frequency offset estimation is always supported by the receiver side with no information regarding this process being handled or added by the transmitter (different reference signals already existing on most communication standards are given use). This is particularly useful as the transmitter does not need to be modified at all.
- For the purpose of analysis, two users (or base stations in case of down-link) are attempted as the main case scenario in this work. A comparison to independent estimation (by assuming an unwanted user as interference) is presented against joint estimation for a number of up to two transmitters.
- The multiple access scheme for which the solution is presented is OFDMA (Orthogonal Frequency Division Multiple Access). This is particularly applicable for today's requirements and trends as shown on the specifications of 4G and early expectations of the upcoming to be released standards.
- For the sake of simulation, only the frequency offset estimation process is carried out at the receiver side. It is not required to perform any channel estimation nor information decoding of the received signal in order to estimate the performance of the proposed algorithm.
- A SISO (Single Input Single Output) system model is used.
- A flat power delay profile (with long coherence time of the channel compared to the OFDM period) along the channel length is considered.
- A Gaussian channel is assumed and characterized by the so mentioned PDP.
- Additive white Gaussian noise (AWGN) is assumed at the receiver.
- An interleaved (between both users), in frequency, pilot symbol distribution is used with independent uniformly distributed symbols as coded pilot symbol data.
- The frequency offset is assumed to be constant over one OFDM period.

The model for the received signal can be expressed either with or without user data contained in the OFDM symbol used to estimate the frequency offset as shown in Section 2.2 and Section 2.3.

2.2 Complete Signal Model (CSM)

The first and more general case contains user data and pilot symbols altogether, the model for the received signal is:

$$\boldsymbol{s}_{\mathrm{CSM}} \stackrel{\mathrm{def}}{=} \sum_{k=1}^{2} D(\epsilon_k) H_k Q \left[\boldsymbol{p}_k + \boldsymbol{d}_k\right] + \boldsymbol{n}. \tag{2.1}$$

Throughout the thesis, vectors are denoted by bold lowercase letters, matrices by uppercase letters and the sub-index k goes from 1 to 2, the number of users at the transmitter end.

Furthermore, $\boldsymbol{s}_{\mathrm{CSM}}$ in (2.1) is an N_{FFT} sized column vector representing the complex base-band representation of the perfectly removed CP (Cyclic Prefix) received time signal. There, the SCM subscripts were used for the sake of uniqueness in the definition but in general it will be referred to $\boldsymbol{s}$ and its definition should be known by context. Besides, H_k is an $N_{\mathrm{FFT}} \times N_{\mathrm{FFT}}$ circular convolution matrix with the impulse response from user k to the receiver on its columns; Q is the N_{FFT} sized inverse Fourier transform matrix and $\boldsymbol{n}$ is an N_{FFT} sized random vector of additive white Gaussian noise with $N_0 I_{N_{\mathrm{FFT}}}$ covariance matrix. Also, $D(\epsilon_k)$ is an $N_{\mathrm{FFT}} \times N_{\mathrm{FFT}}$ complex valued diagonal matrix with the sample by sample phase shift introduced by the normalized (to the OFDM sub-carrier spacing) frequency offset ϵ_k, specifically

$$\big[D\big(\epsilon_k\big)\big]_{l,l} = e^{-\jmath 2\pi \epsilon_k \frac{l-1}{N_{\mathrm{FFT}}}} \qquad \forall l = 1, 2, ..., N_{\mathrm{FFT}},$$

where $\boldsymbol{p}_k$ and $\boldsymbol{d}_k$ are N_{FFT} sized column vectors with the in-phase and quadrature representation of the pilot symbols and data from user k, respectively.

As shown in Figure 2.1, the pilot symbols and data are interleaved in frequency for the current OFDM symbol period. Nevertheless, it should be noted that the method developed here works for any arbitrarily defined pilot/data frequency and time distribution.

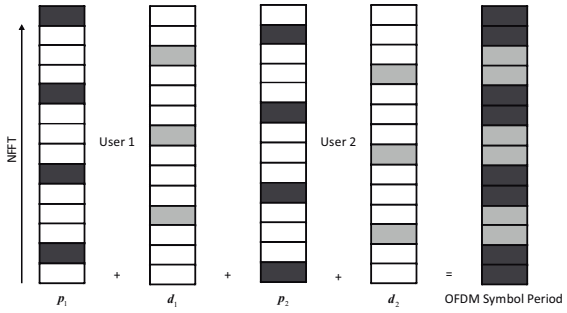

Figure 2.1: Complete Model Representation for the Location of Pilot Symbols and User Data

Besides, the constellation is assumed to be centered on the in-phase and quadrature origin. In the same way, both the AWGN and the channel taps are ZMCSCG described only by their covariance [5]. Hence, the expectation of the received signal is the N_{FFT} sized column vector of zeros $\mathbb{E}\{\boldsymbol{s}\} = \boldsymbol{0}$.

As its importance will be clear later, the covariance matrix for the received signal given by $\Sigma \stackrel{\mathrm{def}}{=} \mathbb{E}\{\boldsymbol{s}\,\boldsymbol{s}^H\}$ is

$$\Sigma = \sum_{k=1}^{2} D\big(\epsilon_k\big) Q P_k Q^H \mathrm{diag}\big(\mathbf{p_h}\big) Q P_k^H Q^H D\big(\epsilon_k\big)^H + \mathbb{E}_{\boldsymbol{d}} \Big\{ \sum_{k=1}^{2} D\big(\epsilon_k\big) Q D_k Q^H \mathrm{diag}\big(\mathbf{p_h}\big) Q D_k^H Q^H D\big(\epsilon_k\big)^H \Big\} + I_{N_{\mathrm{FFT}}} N_0, \quad (2.2)$$

where $\mathbf{p}_{\boldsymbol{h}}$ is an N_{FFT} sized vector containing the power delay profile of the transmission channel; P_k and D_k are diagonal matrices formed by the pilot symbols and data from user k, respectively (not to confuse with $D(\epsilon_k)$). The derivation of the covariance follows.

Covariance Derivation of (2.2)

An expression for the covariance matrix can be obtained by knowing (2.1). Particularly, the following is assumed to be known:

1. The circular matrix convolution representation from user k can be diagonalized by $H_k = Q\Delta_k Q^H$ where Q is the inverse Fourier transform matrix and Δ_k is a matrix with the channel frequency response of the mentioned channel on its main diagonal.
2. The channel transfer function vector representation can be written as $\boldsymbol{\delta}_k = Q^H \boldsymbol{h}_k$ given that Q is the (unitary) inverse Fourier Transform matrix and $\boldsymbol{h}_k$ is the channel impulse response vector.
3. Channels from different users are assumed to be uncorrelated to each other: $\mathbb{E}\{\boldsymbol{h}_i \boldsymbol{h}_j^H\} = 0_{n,n} \forall i \neq j$.
4. Using 2. the channel transfer function auto-correlation matrix can be expressed as: $\mathbb{E}\{\boldsymbol{\delta}_k \boldsymbol{\delta}_k^H\} = \mathbb{E}\{Q^H \boldsymbol{h}_k \boldsymbol{h}_k^H Q\} = Q^H \mathrm{diag}(\mathbf{p}_{\boldsymbol{h}}) Q$.

By expanding (2.1) and using the previous assumptions we get

$$\boldsymbol{s} = \sum_{k=1}^{2} \left\{ D(\epsilon_k) H_k Q\, \boldsymbol{p}_k + D(\epsilon_k) H_k Q\, \boldsymbol{d}_k \right\} + \boldsymbol{n},$$

$$\boldsymbol{s} = \sum_{k=1}^{2} \left\{ D(\epsilon_k) Q P_k\, \boldsymbol{\delta}_k + D(\epsilon_k) Q D_k\, \boldsymbol{\delta}_k \right\} + \boldsymbol{n}, \qquad \text{(Using 1.)}$$

$$\boldsymbol{s} = \sum_{k=1}^{2} \left\{ D(\epsilon_k) Q P_k Q^H\, \boldsymbol{h}_k + D(\epsilon_k) Q D_k Q^H\, \boldsymbol{h}_k \right\} + \boldsymbol{n}, \qquad \text{(Using 2.)}$$

$$\begin{aligned} \Sigma = \mathbb{E}_{\boldsymbol{h}} \Big\{ \sum_{k=1}^{2} D(\epsilon_k) Q P_k Q^H \boldsymbol{h}_k \boldsymbol{h}_k^H Q P_k^H Q^H D(\epsilon_k)^H \Big\} \qquad \text{(Using 3.)} \\ + \mathbb{E}_{\boldsymbol{d}} \left\{ \mathbb{E}_{\boldsymbol{h}} \Big\{ \sum_{k=1}^{2} D(\epsilon_k) Q D_k Q^H \boldsymbol{h}_k \boldsymbol{h}_k^H Q D_k^H Q^H D(\epsilon_k)^H \Big| \boldsymbol{d} \Big\} \right\} \\ + I_{N_{\mathrm{FFT}}} N_0, \end{aligned}$$

$$\therefore \Sigma = \sum_{k=1}^{2} D(\epsilon_k) Q P_k Q^H \operatorname{diag}(\mathbf{p_h}) Q P_k^H Q^H D(\epsilon_k)^H \quad \text{(Using 4.)}$$
$$+ \mathbb{E}_{\boldsymbol{d}} \Big\{ \sum_{k=1}^{2} D(\epsilon_k) Q D_k Q^H \operatorname{diag}(\mathbf{p_h}) Q D_k^H Q^H D(\epsilon_k)^H \Big\} + I_{N_{\mathrm{FFT}}} N_0.$$

It can be shown by evaluating the explicit expectation over $\boldsymbol{d}$ in (2.2) that the resulting covariance matrix is

$$\Sigma = \sum_{k=1}^{2} D(\epsilon_k) Q P_k Q^H \operatorname{diag}(\mathbf{p_h}) Q P_k^H Q^H D(\epsilon_k)^H$$
$$+ \sum_{k=1}^{2} D(\epsilon_k) Q f_k\Big(Q^H \operatorname{diag}(\mathbf{p_h}) Q\Big) Q^H D(\epsilon_k)^H + I_{N_{\mathrm{FFT}}} N_0, \quad (2.3)$$

assuming independent and identically distributed user data with $\mathbb{E}\{|d_{k,n}|^2\} = 1$, where n stands for the indices corresponding to user data (see Figure 2.1) and k for the indexed user. Furthermore, $f_k(M)$ with $M \in \mathcal{C}^{N_{\mathrm{FFT}} \times N_{\mathrm{FFT}}}$ is a function which outputs a sparse matrix with the n^{th} diagonal element taken from M with n corresponding to all user data indices associated to user k. The following figure shows how $f_k(M)$ would look like according to Figure 2.1 counting indices starting from the bottom.

$$f_k\left(\begin{bmatrix} m_{1,1} & m_{1,2} & m_{1,3} & m_{1,4} & m_{1,5} & \cdots & m_{1,14} \\ m_{2,1} & m_{2,2} & m_{2,3} & m_{2,4} & m_{2,5} & \cdots & \vdots \\ m_{3,1} & m_{3,2} & m_{3,3} & m_{3,4} & m_{3,5} & \cdots & \vdots \\ m_{4,1} & m_{4,2} & m_{4,3} & m_{4,4} & m_{4,5} & \cdots & \vdots \\ m_{5,1} & m_{5,2} & m_{5,3} & m_{5,4} & m_{5,5} & \cdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ m_{14,1} & \cdots & \cdots & \cdots & \cdots & \cdots & m_{14,14} \end{bmatrix}\right) = \begin{cases} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 & 0 & \cdots & \vdots \\ 0 & 0 & m_{4,4} & 0 & 0 & \cdots & \vdots \\ 0 & 0 & 0 & \ddots & 0 & \cdots & \vdots \\ 0 & 0 & 0 & 0 & m_{8,8} & \cdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \end{bmatrix}, & k=1 \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & 0 & 0 & \cdots & \vdots \\ 0 & 0 & m_{3,3} & 0 & 0 & \cdots & \vdots \\ 0 & 0 & 0 & \ddots & 0 & \cdots & \vdots \\ 0 & 0 & 0 & 0 & m_{7,7} & \cdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \end{bmatrix}, & k=2 \end{cases}$$

Figure 2.2: Defined function $f_k(M)$ for Figure 2.1

2.3 Reduced Signal Model (RSM)

As for the purpose of evaluating the presented algorithm there is no need in adding user data, a simplified version of the received signal model is considered instead.

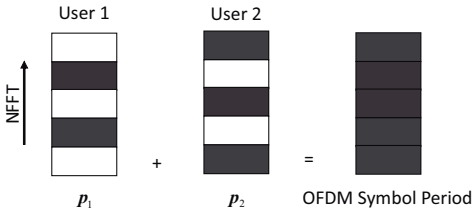

Figure 2.3: Model of Interest Representation for the Location of Pilot Symbols

According to this model, no uncertain user data is present and all frequency slots are used for the purpose of channel estimation (and frequency offset estimation) as shown in Figure 2.3. The model for the received signal is basically the same as the one in Section 2.2 with no user data,

$$\boldsymbol{s}_{\mathrm{RSM}} \overset{\mathrm{def}}{=} \sum_{k=1}^{K} D\big(\epsilon_k\big) H_k Q\, \boldsymbol{p}_k + \boldsymbol{n}. \quad (2.4)$$

The assumption of no user data entails an important abbreviation on the covariance matrix compared to the expression in (2.3), particularly:

$$\Sigma = \sum_{k=1}^{2} D\big(\epsilon_k\big) Q P_k Q^H \mathrm{diag}\big(\mathbf{p_h}\big) Q P_k^H Q^H D\big(\epsilon_k\big)^H + I_{N_{\mathrm{FFT}}} N_0. \quad (2.5)$$

This will be the model used from now on. Most importantly, the power delay profile normalization is chosen such that varying N_{FFT} does not affect the SNR as it will be shown in the following derivation.

System SNR Derivation

Using the expression for the covariance matrix (2.5), via its trace in order to compute the average received signal power, the noise and signal contributing parts can be identified

$$\underbrace{\mathrm{tr}\left(\sum_{k=1}^{2} D\big(\epsilon_k\big) Q P_k Q^H \mathrm{diag}\big(\mathbf{p_h}\big) Q P_k^H Q^H D\big(\epsilon_k\big)^H\right)}_{\text{Users contribution}} + \underbrace{\mathrm{tr}\big(I_{N_{\mathrm{FFT}}} N_0\big)}_{\text{Noise contribution}}.$$

By definition in this project, the system SNR is going to be the ratio of the signal power from all users to the noise power, specifically

$$\mathrm{SNR} \overset{\mathrm{def}}{=} \frac{\text{Signal Power from All Users}}{\text{Noise Power}} = \frac{\mathrm{tr}\Big(\mathrm{diag}\big(\mathbf{p_h}\big)\Big)}{N_{\mathrm{FFT}}} \frac{1}{N_0},$$

where $|p_k|^2 = 1$ is assumed. In order to make the SNR independent of N_{FFT} it should be enforced $\sum \mathbf{p_h} = N_{\mathrm{FFT}}$, giving an SNR equal to the reciprocal of the noise power (i.e. $\mathrm{SNR} = \frac{1}{N_0}$).

Chapter 3
Estimation Theory

One big problem usually faced in engineering is the requirement of estimating a parameter given some measurements. This is the study of estimation theory and for this project, in practical terms, it becomes quite important in order to build a function of the measured received signal that give a good estimate of the frequency offset.

As in most of problems faced in engineering, there is a trade-off between complexity and performance. It is very important to consider how well the estimator performs considering these two aspects based on some criteria. Particularly, computational complexity for the current problem is a big issue to consider as it is vital to give an estimate as fast as possible having in mind the coherence time of the frequency offset.

With this in mind, some estimation methods will be mentioned briefly in this chapter following the maximum likelihood method (which was the estimation method given use in this project).

3.1 Least Squares Estimation

This estimation method in its linear and nonlinear versions are popular in the literature but no statistical assumptions are made. Assuming the model

$$\boldsymbol{x} = f(\boldsymbol{\theta}) + \boldsymbol{n}, \tag{3.1}$$

where $\boldsymbol{x}$ is the vector of measurements, $\boldsymbol{\theta}$ is the vector of parameters to estimate and $\boldsymbol{n}$ is vector of noise in the measurements. Least squares is essentially a problem of minimizing the square norm of the residuals ($\boldsymbol{r}$) given by

$$\boldsymbol{r} = \boldsymbol{x} - f(\boldsymbol{\theta}). \tag{3.2}$$

The number of measurements vs. the number of parameters defines if the system is under-determined or over-determined which, among others, governs the confidence of the estimate. As this method does not minimize the residuals in any statistical sense, it is not optimal when information about the noise process is available (even though under some conditions it does tend to the same result as some statistically optimal methods [6]). Henceforth, it is not considered for the problem at hand.

3.2 Method of Moments

This estimation method is known for its simplicity. If there is a model for the parameters to estimate in terms of some moments of the underlying statistical process, it is a matter of approximating the k^{th} moment using

$$m^k = \frac{1}{T}\sum_{t=1}^{\mathsf{T}} x_t^k, \tag{3.3}$$

where the sub-index in x_t numerates each realization and k is the power to which the measurement is raised in order to obtain an estimate for the k^{th} moment. In (3.3) it is assumed that the measurements are scalars but it generalizes to higher dimensions according to the model.

The disadvantage and reason this method is not considered for this project is that it is not always efficient [6].

3.3 Maximum Likelihood (ML) Estimation

This method has several properties that make it an appropriate choice as the estimator for the presented problem. It is sometimes problematic to implement due to computational complexity but it achieves a small estimation error when a large set of independent observations is available.

The ML estimator is built by considering the likelihood function. The likelihood function is the joint probability density function (denoted p) of the measurements with swapped arguments (i.e. the measurements given and parameters to evaluate), namely

$$\lambda\big(\boldsymbol{\theta};\boldsymbol{x}\big) = p\big(\boldsymbol{x}|\boldsymbol{\theta}\big), \tag{3.4}$$

where the notation was the same as the one used in Section 3.1 and the dependency of the likelihood function on the current measurement was made explicit. In some cases and due to the exponential behavior of many distributions, the logarithm (as it is a monotonic function of the argument) is considered instead by:

$$\lambda\big(\boldsymbol{\theta};\boldsymbol{x}\big) = \log\Big(p\big(\boldsymbol{x}|\boldsymbol{\theta}\big)\Big). \tag{3.5}$$

From now on, we will refer to this as the likelihood function unequivocally. By considering the likelihood function, a measure of how well some measurements ($\boldsymbol{x}$) fit the assumed model (embedded in p) is obtained. Thus, the ML estimator is

$$\hat{\boldsymbol{\theta}}^{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} \lambda\big(\boldsymbol{\theta};\boldsymbol{x}\big), \tag{3.6}$$

where it can be seen that the estimate is equivalent up to an additive constant (in terms of the parameters) and scaling (if scaled by a negative it should be stated a minimization problem) in the likelihood function.

Also important to mention is the possibility of several local maxima (multimodal behavior) on the likelihood function. Because of this, it is of great importance to find the global optima in (3.6). Local optimization methods could bring wrong estimates when applied to the likelihood function.

Chapter 4

Frequency Offset Maximum Likelihood Estimation: The ML Estimator for the Presented Problem and its Complexity

In order to build the estimator for the specific problem at hand, some considerations about the received signal model should be made. Recalling the received signal model of Section 2.3, the underlying random behavior in $\boldsymbol{s}$ is due to the additive white Gaussian noise and the stochastic channel.

By using some linear algebra properties like the diagonalization of a circular matrix by the DFT matrix (see covariance derivation of Section 2.2), the received signal in (2.4) can be expressed as

$$\boldsymbol{s} = \sum_{k=1}^{K} D\big(\epsilon_k\big) Q P_k Q^H \boldsymbol{h}_k + \boldsymbol{n}, \tag{4.1}$$

where P_k is the diagonal matrix formed by the pilot symbols from user k, and $\boldsymbol{h}_k$ is the N_{FFT} sized column vector with the channel impulse response realization from user k to the receiver.

As any linear transformation of a multivariate Gaussian is also Gaussian, and the addition of two multivariate Gaussian also is, the measurements for an OFDM period (N_{FFT} samples not including any cyclic prefix) are jointly Gaussian. This allows us to build the likelihood function by means of

$$p\big(\boldsymbol{s}|\epsilon_1,\epsilon_2\big) = \frac{1}{\pi^{N_{\mathrm{FFT}}}|\Sigma\big(\epsilon_1,\epsilon_2\big)|} \exp\Big(- \boldsymbol{s}^H \Sigma(\epsilon_1,\epsilon_2)^{-1} \boldsymbol{s} \Big), \tag{4.2}$$

$$\log\Big(p\big(\boldsymbol{s}|\epsilon_1,\epsilon_2\big)\Big) = -N_{\mathrm{FFT}}\log(\pi) - \log\big(|\Sigma\big(\epsilon_1,\epsilon_2\big)|\big) - \boldsymbol{s}^H \Sigma(\epsilon_1,\epsilon_2)^{-1}\boldsymbol{s},$$

$$\lambda\big(\epsilon_1,\epsilon_2;\boldsymbol{s}\big) = -N_{\mathrm{FFT}}\log(\pi) - \log\big(|\Sigma\big(\epsilon_1,\epsilon_2\big)|\big) - \boldsymbol{s}^H \Sigma(\epsilon_1,\epsilon_2)^{-1}\boldsymbol{s}, \tag{4.3}$$

where $\Sigma(\epsilon_1,\epsilon_2)$ is the covariance matrix of the received signal (2.5) and the fact that, in this case, $\boldsymbol{s}$ turns out to be zero-mean circularly symmetric complex Gaussian (ZMCSCG) was used to express the probability density function p [5].

Finally, the ML frequency offset estimator can be mathematically expressed as

$$\begin{aligned}\left(\hat{\epsilon}_1^{\mathrm{ML}}, \hat{\epsilon}_2^{\mathrm{ML}}\right) &= \arg\min_{\epsilon_1,\epsilon_2} \Big\{ -\lambda\big(\epsilon_1, \epsilon_2; \boldsymbol{s}\big) \Big\}, \\ &= \arg\min_{\epsilon_1,\epsilon_2} \Big\{ \log\left(\left|\Sigma\big(\epsilon_1, \epsilon_2\big)\right|\right) + \boldsymbol{s}^H \Sigma(\epsilon_1, \epsilon_2)^{-1} \boldsymbol{s} \Big\}, \end{aligned} \tag{4.4}$$

where the dependency of the covariance matrix on the frequency offsets was made explicit for clarity. It can be seen that the task of proceeding further with (4.4) analytically is rather complicated and hence a solution relies on numerical evaluation. Equation (4.4) along with the expression for the covariance matrix (2.5) show that for the receiver in order to give an ML estimate it is required to know:

- The Power Delay Profile (PDP) of the channel.
- The pilot symbol time and frequency distribution for every user.
- An estimate of the environment's noise power.
- One received OFDM symbol (namely $\boldsymbol{s}$).

Afterwards, by applying a numerical nonlinear constrained optimization method, the estimates of the frequency offsets can be obtained. This last step, which for an offline computer would be pretty easy to solve, is on the other hand a heavy task for real time architectures like the ones generally used in receiver ends. More importantly, depending on the coherence time of the frequency offset, it might be required to repeat this process quite often (every few OFDM symbol periods in practice for unstable oscillators) so not only a good enough but also fast estimate should be available for the receiver to do compensation.

4.1 Joint ML Estimation Drawbacks

The reason why the computation of the estimator is resource consuming is associated to the nonlinear optimization problem presented in (4.4) (recall that the optimization is carried out over ϵ_1 and ϵ_2) and the computations over the covariance matrix (inverse and determinant).

First, and as mentioned before, the existence of several local optima (multimodal behavior) impose a condition of not relying in the solution to any arbitrary starting point when using iterative optimization algorithms such as gradient descent (even though this would not be gradient descent's biggest drawback when applied to the likelihood in equation 4.3). This suggests the use of a more complete optimization method with, for example, an exhaustive search initialization step. Of course, this sort of solutions would imply extra evaluations of the likelihood function and probably the under-utilization of the information contained in these initialization evaluations.

Lastly, the difficulty associated to the likelihood function evaluation can be understood as follows: the computational complexity of the inverse and determinant of the $N_{\mathrm{FFT}} \times N_{\mathrm{FFT}}$ matrix covariance matrix in (4.4) is $O\big(N_{\mathrm{FFT}}^3\big)$ in big O

notation, hence the time and resources requirements grow fast with N_{FFT} . When considering the optimization algorithm, the number of minimum evaluations in order to obtain a decent estimate is of great importance.

These two main issues are going to be addressed in the next chapter through a technique called functional decomposition. The following section is a small note on user-independent compared to user-joint maximum likelihood technique applied to the estimation problem to solve in this project.

4.2 Notes on User-Independent Estimation

Up to this point only joint estimation has been considered. In this section, a small overview on what a maximum likelihood independent estimation solution would be like, the advantages and disadvantages compared to joint estimation are given. As it would be to expect, joint estimation overcomes independent estimation both performance-wise and complexity-wise comparatively speaking.

To begin, let's consider the model for the received signal. The excess term in (2.4) due to user 2 (when estimating frequency offset corresponding to user 1) can very well be regarded as interference and posed as a problem of independent estimation with the received signal model

$$\boldsymbol{s}_1 = D\big(\epsilon_1\big)H_1 Q\,\boldsymbol{p}_1 + \boldsymbol{i} + \boldsymbol{n}, \tag{4.5}$$

where the received signal shown in (2.4) is now explicitly expressing the term of interest as if we were to estimate the frequency offset for user 1.

In the same way, the received signal model can be expressed in terms of ϵ_2 regarding the contribution from user 1 as interference by:

$$\boldsymbol{s}_2 = D\big(\epsilon_2\big)H_2 Q\,\boldsymbol{p}_2 + \boldsymbol{i} + \boldsymbol{n}. \tag{4.6}$$

The unwanted terms in (4.6) are the contribution from the other user $\boldsymbol{i}$ (regarded as colored Gaussian noise) and the additive white Gaussian noise (AWGN) term $\boldsymbol{n}$. The information contained in the interference term $\boldsymbol{i}$ would of course not be used to improve the performance of the current estimator which now would not be joint.

In the case of user-independent estimation, the covariance matrix for the reduced model (see Section 2.3) can be expressed as

$$\begin{aligned}\Sigma_k\big(\epsilon_k\big) &= \mathbb{E}\big\{\boldsymbol{s}_k\,\boldsymbol{s}_k^H\big\} \\ &= D\big(\epsilon_k\big)QP_kQ^H\mathrm{diag}\big(\mathbf{p_h}\big)QP_k^HQ^HD\big(\epsilon_k\big)^H \\ &\qquad + \Big[\mathrm{Q}P_{\bar{k}}Q^H\mathrm{diag}\big(\mathbf{p_h}\big)QP_{\bar{k}}^HQ^H\Big]\circ F + I_{N_{\mathrm{FFT}}}N_0, \end{aligned}\tag{4.7}$$

where ϵ_k stands for the current frequency offset to estimate, P_k to a diagonal matrix with the frequency representation for the pilot symbols associated to user k and $P_{\bar{k}}$ to the frequency representation for the pilot symbols associated to the user considered as interference for the independent estimation problem at hand. Furthermore, $A \circ B$ stands for the Hadamard product (i.e. component-wise product)

between matrices A and B. Lastly, F is a matrix with each component defined as $F_{n,m} = \operatorname{sinc}\left(\frac{n-m}{N_{\text{FFT}}}\right)\ \forall n = 1, 2, ..., N_{\text{FFT}},\ m = 1, 2, ..., N_{\text{FFT}}$.

The derivation for the covariance matrix follows from (2.2) as shown below.

Covariance Derivation of (4.7)

Assume the following to be known:

1. $\operatorname{diag}(\boldsymbol{a}) B \operatorname{diag}(\boldsymbol{a})^H = B \circ \boldsymbol{a}\boldsymbol{a}^H$ where $\boldsymbol{a}$ is any complex vector, B any matrix congruent size and $A \circ B$ is the Hadamard product.
2. $\mathbb{E}_{\epsilon_{\bar{k}}}\left\{ \left[D\left(\epsilon_{\bar{k}}\right)\right]_{n,n} \left[D\left(\epsilon_{\bar{k}}\right)\right]^*_{m,m} \right\} = \mathbb{E}_{\epsilon_{\bar{k}}}\left\{ e^{-\jmath 2\pi \epsilon_{\bar{k}} \frac{n-m}{N_{\text{FFT}}}} \right\} = \operatorname{sinc}\left(\frac{n-m}{N_{\text{FFT}}}\right)$ assuming a uniformly distributed frequency offset.

Starting from (2.2), the covariance of interest to estimate frequency offset from user k is

$$
\begin{aligned}
\Sigma_k\left(\epsilon_k\right) &= \mathbb{E}_{\epsilon_{\bar{k}}}\left\{\Sigma\right\} \\
&= D\left(\epsilon_k\right) Q P_k Q^H \operatorname{diag}\left(\mathbf{p}_{\boldsymbol{h}}\right) Q P_k^H Q^H D\left(\epsilon_k\right)^H \\
&\quad + \mathbb{E}_{\epsilon_{\bar{k}}}\left\{ D\left(\epsilon_{\bar{k}}\right) Q P_{\bar{k}} Q^H \operatorname{diag}\left(\mathbf{p}_{\boldsymbol{h}}\right) Q P_{\bar{k}}^H Q^H D\left(\epsilon_{\bar{k}}\right)^H \right\} + I_{N_{\text{FFT}}} N_0 \\
&= D\left(\epsilon_k\right) Q P_k Q^H \operatorname{diag}\left(\mathbf{p}_{\boldsymbol{h}}\right) Q P_k^H Q^H D\left(\epsilon_k\right)^H \\
&\quad + Q P_{\bar{k}} Q^H \operatorname{diag}\left(\mathbf{p}_{\boldsymbol{h}}\right) Q P_{\bar{k}}^H Q^H \circ \mathbb{E}_{\epsilon_{\bar{k}}}\left\{ \tilde{\boldsymbol{d}}(\epsilon_{\bar{k}}) \tilde{\boldsymbol{d}}(\epsilon_{\bar{k}})^H \right\} + I_{N_{\text{FFT}}} N_0.
\end{aligned}
$$

Finally, obtaining

$$
\begin{aligned}
\therefore \Sigma_k\left(\epsilon_k\right) = D\left(\epsilon_k\right) Q P_k Q^H \operatorname{diag}\left(\mathbf{p}_{\boldsymbol{h}}\right) Q P_k^H Q^H D\left(\epsilon_k\right)^H \\
+ Q P_{\bar{k}} Q^H \operatorname{diag}\left(\mathbf{p}_{\boldsymbol{h}}\right) Q P_{\bar{k}}^H Q^H \circ F + I_{N_{\text{FFT}}} N_0,
\end{aligned}
$$

where $\tilde{\boldsymbol{d}}(\epsilon_k)$ is the column vectored diagonal of the matrix $D\left(\epsilon_k\right)$ and $\bar{k}$ is the index of the user considered as an interference.

It is important to consider that computational complexity is increased through independent estimation compared to joint estimation. Particularly, by using independent estimation, the joint estimation is divided into several smaller (but still expensive) problems including a matrix inversion and a one dimension optimum search each.

The main reason why independent estimation is not convenient for the current problem is that the inverse and determinant covariance matrix are still, as in joint estimation, required to be computed. Nevertheless, performance is very inferior to the joint ML estimator as it will be shown in Section 6.

Some combination of independent and joint estimation (by using one or the other each time) can be considered according to the allowed frequency offset and conditions as the signal to noise ratio. Although, this is out of the scope of this work and becomes a problem of platform specific resource optimization.

Chapter 5

How to Deal with Multi-Modality and Complexity

There are several methods in the literature where optimization problems as the one shown in (4.4) are addressed. Derivative-free (which are the most common when facing unreliable, costly or hard to compute gradient [7] [8]) and common iterative algorithms are in general not giving efficient use to the expensive evaluations in a joint fashion but rather trying to find a path towards the optima. This basically means every sample taken before the current iteration is not used anymore.

Furthermore, the fact that local optima are possible makes the idea of using uniform likelihood evaluations over the parameter's domain attractive. Hence, if there is a method which relies on the so mentioned uniform likelihood evaluations in an efficient way (in order to extract as much information as possible from the available measurements) it would make these expensive calculations of the likelihood function worth the resources (time and computation wise).

Given that the mathematical expression for the function to optimize is well known, a good characterization of the likelihood function could be achieved. Therefore functional decomposition (a technique which can be closely related to different applications as image compression and feature extraction [9] [10]) can be applied. This non-iterative technique could not only give efficient joint use to every evaluation of the likelihood function but ideally get a better estimate of the global optima compared to iterative optimization methods applied directly onto the likelihood function.

5.1 Functional Decomposition

Functional decomposition is a broad term that has theoretical applications in different areas. In the specific case of this project it is used not only referring to the identification of composing elements for the likelihood function but also to the ability of finding its representation (viewed as a random process indexed by the frequency offsets) as a linear combination of these composing elements.

Basically the idea is to be able to approximate the likelihood function as

$$\widehat{\lambda}(\epsilon_1, \epsilon_2) = \sum_{l=1}^{L} \alpha_l \Phi_l(\epsilon_1, \epsilon_2). \tag{5.1}$$

In equation (5.1), L is the number of basis functions, $\Phi_l\big(\epsilon_1, \epsilon_2\big)$ the l^{th} function in the base and α_l is the l^{th} decomposition coefficient. Particularly, the idea is to find a representative set of basis functions such that the number of decomposition coefficients L is small with, what is commonly referred to, a small loss [11].

If such a basis was found, the α coefficients could be obtained and the optimization algorithm in (4.4) could be carried out over the likelihood approximation (instead of over the computationally heavy true likelihood function) given that the basis gives a low enough approximation error.

Because the likelihood function can be regarded as a random process of the multivariate sampled received signal $\boldsymbol{s}$, the α coefficients can be seen as random variables on the specific realization of the likelihood process.

5.2 The Optimum Base

The Karhunen-Loève basis [12] would be the optimal base in the sense of minimizing the mean square approximation error (MMSE). Such a basis requires to know the covariance function of the likelihood process, namely

$$K_\lambda\big(\epsilon_1, \epsilon_2, \tilde{\epsilon}_1, \tilde{\epsilon}_2\big) = \mathbb{E}_{\boldsymbol{s}}\{\lambda\big(\epsilon_1, \epsilon_2\big)\, \lambda\big(\tilde{\epsilon}_1, \tilde{\epsilon}_2\big)\} - \overline{\lambda}\big(\epsilon_1, \epsilon_2\big)\overline{\lambda}\big(\tilde{\epsilon}_1, \tilde{\epsilon}_2\big), \tag{5.2}$$

where $\overline{\lambda}(\epsilon_1, \epsilon_2)$ stands for the expectation of $\lambda(\epsilon_1, \epsilon_2)$. Because of the difficulty on finding an analytic expression for the likelihood covariance, its estimate (through the covariance matrix of the discretized likelihood process) is considered instead.

Therefore, by discretizing the (ϵ_1, ϵ_2) space and vector mapping the likelihood function, the discretized covariance can be represented and estimated by

$$\Sigma_{\boldsymbol{\lambda}} = \underbrace{\mathbb{E}\big\{\boldsymbol{\lambda}\,\boldsymbol{\lambda}^{\mathsf{T}}\big\}}_{R_{\boldsymbol{\lambda}}} - \mathbb{E}\{\boldsymbol{\lambda}\}\mathbb{E}\{\boldsymbol{\lambda}\}^{\mathsf{T}}, \tag{5.3}$$

$$\Sigma_{\boldsymbol{\lambda}} \approx \frac{1}{N-1}\sum_{n=1}^{N} \boldsymbol{\lambda}_n\, \boldsymbol{\lambda}_n^{\mathsf{T}} - \frac{1}{N^2}\sum_{n=1}^{N} \boldsymbol{\lambda}_n \sum_{m=1}^{N} \boldsymbol{\lambda}_m^{\mathsf{T}}, \tag{5.4}$$

where (5.3) shows the definition for the covariance ($\Sigma_{\boldsymbol{\lambda}}$) and correlation ($R_{\boldsymbol{\lambda}}$) matrices assuming $\boldsymbol{\lambda}$ as the column version of the likelihood.

Furthermore, (5.4) shows how to approximate the likelihood covariance matrix previously defined by knowing N realizations of the likelihood process. Ideally, the number of realizations to approximate the covariance matrix should be as big as possible in order to achieve a good estimate.

When dealing with the discretized version of the covariance function, the problem of finding the optimum basis (in the MMSE approximation sense as already mentioned) coincides with some well-known results of PCA (principal component analysis).

The idea of PCA applied here would be to find the set of basis functions $\Phi_l(\epsilon_1, \epsilon_2)\, \forall l = 1, 2, ..., L$ that give statistically uncorrelated decomposition coefficients α_l. Uncorrelatedness means there is no statistical linear dependence among coefficients (i.e. $\mathbb{E}\{\alpha_i \alpha_j\} = \mathbb{E}\{\alpha_i\}\mathbb{E}\{\alpha_j\}\, \forall i, j = 1, 2, ..., L$) as a measure of how well the obtained basis does the job of extracting and separating information from

the likelihood process. If it was the case that the likelihood was jointly Gaussian, then it would also imply independence of the decomposition coefficients which is a stronger (and more desirable) condition.

Having a small set of basis functions is desirable and achieved through PCA which ultimately gives a discretized version of the Karhunen-Loève basis. The intuitive reason behind this being desirable is that by identifying the most contributing components of the likelihood function, more information can be extracted from every evaluation of the likelihood.

Hence, the smaller the Karhunen-Loève basis the lesser the number of evaluations required in order to achieve a small approximation (and hence estimation) error. This means that every evaluation of the likelihood would be worth the required resources if the Karhunen-Loève approximation was used.

In order to obtain a numerical representation for the set of basis functions, tens of thousands of likelihood realizations are generated using the system model presented in Section 2.1. Afterwards, the covariance matrix is estimated using (5.4) and the Karhunen-Loève basis is approximated via PCA by obtaining the first Eigen-vectors (corresponding to the higher Eigen-values) of the covariance matrix.

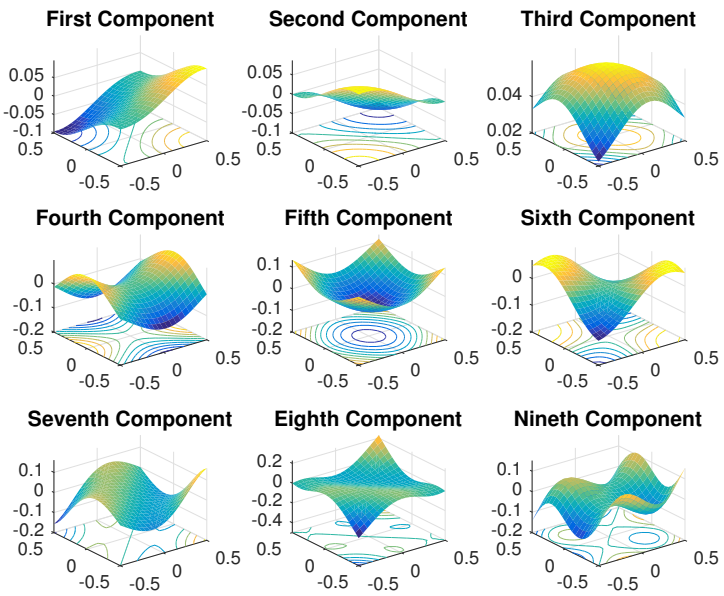

Figure 5.1: First 9 Components of Karhunen-Loève Basis for Estimated Covariance.

Figure 5.1 shows the first nine components of the Karhunen-Loève basis approximation. Some analytic expressions for the first components can be derived in order to obtain a somewhat similar basis function set, but after the fifth component this is not an easy task.

As the idea of using numerical instead of analytic approximations of the Karhunen-Loève basis does not seem as attractive, a different set of basis functions should be selected.

5.3 Selecting a Suitable Base

As already mentioned, the Karhunen-Loève basis is unfeasible to obtain since an analytical expression for the covariance matrix is hard to find. Furthermore, a numerical solution for the basis selection would not be as desirable.

In general, any set of functions spanning the space of functions created by the Karhunen-Loève Basis would serve for the purpose of approximation. The issue is how to find the smallest one among all possible sets. More importantly, a measure Ξ of how well a set of basis function approximates the likelihood should be defined. Hence, in order to achieve so, the ξ value derivation follows.

Derivation of Error Performance Measure ξ

As a first approach, the error performance measure is going to be the normalized approximation residual power in decibels and its derivation follows according to:

1. Define the likelihood energy: $E_\lambda \stackrel{\text{def}}{=} \iint_\Omega \lambda\big(\epsilon_1,\epsilon_2\big)^2 \, d\epsilon_1 \, d\epsilon_2$.
2. Recall that $\widehat{\lambda}\big(\epsilon_1,\epsilon_2\big) = \sum_{l=1}^{L} \alpha_l \Phi_l\big(\epsilon_1,\epsilon_2\big)$ (5.1).
3. Assume an orthogonal set of basis functions, i.e. $\iint_\Omega \Phi_i\big(\epsilon_1,\epsilon_2\big)\Phi_j\big(\epsilon_1,\epsilon_2\big) \, d\epsilon_1 \, d\epsilon_2 = 0 \quad \forall i \neq j$.
4. Assume a normalized set of basis functions, i.e. $\iint_\Omega \Big(\Phi_i\big(\epsilon_1,\epsilon_2\big)\Big)^2 \, d\epsilon_1 \, d\epsilon_2 = 1 \quad \forall i$.
5. Assuming the previous conditions 3 and 4, the decomposition coefficients are defined: $\alpha_l = \iint_\Omega \lambda\big(\epsilon_1,\epsilon_2\big)\Phi_l\big(\epsilon_1,\epsilon_2\big) \, d\epsilon_1 \, d\epsilon_2$.
6. Define $R_\lambda\big(\epsilon_1,\epsilon_2,\tilde{\epsilon}_1,\tilde{\epsilon}_2\big) \stackrel{\text{def}}{=} \mathbb{E}\Big\{\lambda\big(\epsilon_1,\epsilon_2\big)\,\lambda\big(\tilde{\epsilon}_1,\tilde{\epsilon}_2\big)\Big\}$ as the correlation of the likelihood function and $R_{\boldsymbol{\lambda}}$ as its matrix representation.

Let's begin by defining and expanding the intermediate error measure Ξ:

$$\begin{aligned}\Xi &\stackrel{\text{def}}{=} \iint_\Omega \Big(\lambda\big(\epsilon_1,\epsilon_2\big) - \widehat{\lambda}\big(\epsilon_1,\epsilon_2\big)\Big)^2 \, d\epsilon_1 \, d\epsilon_2 \\ &= \iint_\Omega \Big(\lambda\big(\epsilon_1,\epsilon_2\big)^2 - 2\lambda\big(\epsilon_1,\epsilon_2\big)\widehat{\lambda}\big(\epsilon_1,\epsilon_2\big) + \widehat{\lambda}\big(\epsilon_1,\epsilon_2\big)^2\Big) \, d\epsilon_1 \, d\epsilon_2,\end{aligned}$$

$$\Xi = \mathbb{E}_\lambda - 2\sum_{l=1}^{L} \alpha_l \iint_\Omega \lambda(\epsilon_1,\epsilon_2)\Phi_l(\epsilon_1,\epsilon_2)\, d\epsilon_1\, d\epsilon_2 + \iint_\Omega \Big[\sum_{l=1}^{L} \alpha_l \Phi_l(\epsilon_1,\epsilon_2)\Big]^2 d\epsilon_1\, d\epsilon_2, \qquad \text{(Using 1. and 2.)}$$

$$\therefore \Xi = \mathbb{E}_\lambda - \sum_{l=1}^{L} \big[\alpha_l\big]^2. \qquad \text{(Using 3. 4. and 5.)}$$

Afterwards, the error performance measure ξ is defined in terms of Ξ as

$$\xi \overset{\text{def}}{=} \frac{\mathbb{E}\{\Xi\}}{\mathbb{E}\{E_\lambda\}}. \tag{5.5}$$

The ξ value is preferred over Ξ as it is normalized to the energy of the likelihood process. Furthermore, it can be computed by evaluating (5.5):

$$\xi = 1 - \frac{\mathbb{E}\Big\{\sum_{l=1}^{L}\Big[\iint_\Omega \lambda(\epsilon_1,\epsilon_2)\Phi_l(\epsilon_1,\epsilon_2)\, d\epsilon_1\, d\epsilon_2\Big]^2\Big\}}{\mathbb{E}\{E_\lambda\}},$$

$$\xi = 1 - \frac{\sum_{l=1}^{L}\mathbb{E}\Big\{\iiiint_\Psi \lambda(\epsilon_1,\epsilon_2)\lambda(\tilde{\epsilon}_1,\tilde{\epsilon}_2)\Phi_l(\epsilon_1,\epsilon_2)\Phi_l(\tilde{\epsilon}_1,\tilde{\epsilon}_2)\, d\epsilon_1\, d\epsilon_2\, d\tilde{\epsilon}_1\, d\tilde{\epsilon}_2\Big\}}{\mathbb{E}\Big\{\iint_\Omega \lambda(\epsilon_1,\epsilon_2)^2\, d\epsilon_1\, d\epsilon_2\Big\}},$$

$$\therefore \xi = 1 - \frac{\sum_{l=1}^{L}\Big[\iiiint_\Psi R_\lambda(\epsilon_1,\epsilon_2,\tilde{\epsilon}_1,\tilde{\epsilon}_2)\Phi_l(\epsilon_1,\epsilon_2)\Phi_l(\tilde{\epsilon}_1,\tilde{\epsilon}_2)\, d\epsilon_1\, d\epsilon_2\, d\tilde{\epsilon}_1\, d\tilde{\epsilon}_2\Big]}{\iint_\Omega R_\lambda(\epsilon_1,\epsilon_2,\epsilon_1,\epsilon_2)\, d\epsilon_1\, d\epsilon_2},$$

where the regions of integration are $\Omega = [-0.5,\, 0.5] \times [-0.5,\, 0.5]$ and $\Psi = \Omega \times \Omega$. As it is going to be dealt with the estimated correlation matrix $R_{\boldsymbol{\lambda}}$ in (5.3), previous equation analogous discretized version becomes:

$$\therefore \xi \simeq 1 - \frac{\sum_{l=1}^{L} \boldsymbol{\Phi}_l^{\mathsf{T}} R_{\boldsymbol{\lambda}} \boldsymbol{\Phi}_l}{\operatorname{tr}(R_{\boldsymbol{\lambda}})} = 1 - \frac{\operatorname{tr}(\Phi^{\mathsf{T}} R_{\boldsymbol{\lambda}} \Phi)}{\operatorname{tr}(R_{\boldsymbol{\lambda}})}, \tag{5.6}$$

where $\boldsymbol{\Phi}_l$ is the l^{th} discretized basis function as a column vector, Φ is the matrix formed by $\boldsymbol{\Phi}_l$ on its columns $\forall l$ and $R_{\boldsymbol{\lambda}}$ is the correlation matrix of the likelihood process. It is important to point out that $\Phi\Phi^{\mathsf{T}} \neq \mathrm{I}$ even though its columns are ortho-normal vectors as in general Φ is a tall matrix. The ξ value in decibels (i.e. $\xi_{dB} = 10\log_{10}(\xi)$) is going to be considered from now on unless otherwise stated.

Now that we have built the error performance measure for any set of basis functions, the Karhunen-Loève basis can work as a benchmark in order to know what is the best it can be aimed for. Therefore, Figure 5.2 shows the cumulative normalized approximation residual power (referred to as the ξ value from now on) for different number of Karhunen-Loève components up to 19.

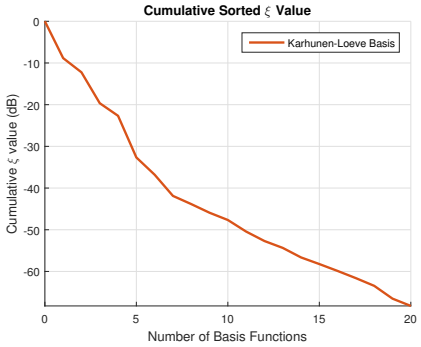

Figure 5.2: ξ Value for the 19 Most Contributing Karhunen-Loève Components.

Figure 5.2 can be seen as a benchmark for the approximation error when decomposing the likelihood.

It must be kept in mind that this is considered a first approach as **the ξ value only measures performance of approximation to the likelihood function**. In fact, performance of estimation should be evaluated in depth using an optimization algorithm over the approximated likelihood function. Afterwards, concluding results from this simulation will be drawn by comparing it to the optimization over the true likelihood function.

The next step is to find a base which analytically described could achieve a good approximation error based on this plot. Having an analytic description of the set of basis functions is important for robustness of the final estimator when using gradient dependent optimization methods over the likelihood approximation. Particularly, because an expression for the gradient is also analytically available (compared to the use of approximations of the gradient via finite differences).

There is a high amount of basis function families in the literature worthwhile to be considered when selecting one in order to approximate the likelihood.

In order to find a good basis, some groups of functions were considered based on some properties appealing to be exploited. Specifically, the families considered were mainly:

- Fourier based.
- Sinc based.

To begin with, the Fourier basis was attractive considering the work in [4]. It indeed showed to be good when selecting a big number of basis functions. This means the number of likelihood function evaluations required in order to find the representation in this basis would also be big (not to have an under-determined system of equations when solving for the decomposition coefficients, see Section 6.2).

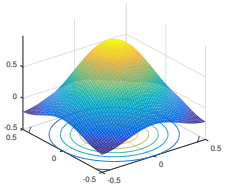

Figure 5.3: 2D Sinc for $\gamma = 2$

As the idea is to reduce the number of evaluations due to their computational complexity, the Fourier basis was discarded.

Afterwards, and based on experiences from studying many likelihood function realizations (quasi-convex at a large scale under general good circumstances), the idea of using a set of functions having several minima at uniformly distributed locations of the parameter's domain seemed appealing.

Hence, two dimensional Sincs were given use with good results for small sets, specifically the set of Sincs is given by

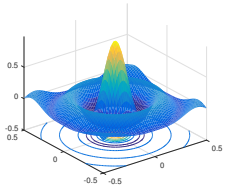

Figure 5.4: 2D Sinc for $\gamma = 7$

$$\Phi_l(\epsilon_1, \epsilon_2) = \text{sinc}\left(\gamma\sqrt{(\epsilon_1 - c_{l,1})^2 + (\epsilon_2 - c_{l,2})^2}\right), \quad \forall l = 1, 2, ..., L \qquad (5.7)$$

where γ is a value to control the width of the two dimensional sinc (measured to the first zero crossing, e.g. see Figure 5.3 and 5.4) and $\boldsymbol{c}_l$ refers to the center of the l^{th} basis function in the (ϵ_1, ϵ_2) space.

The γ and $\boldsymbol{c}_l$ parameters are optimized via a constrained exhaustive search giving the lowest ξ value among a range of γ and number of Sincs to be utilized.

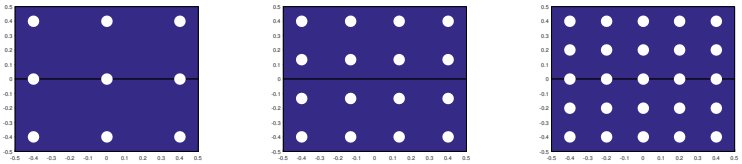

Figure 5.5: Center Distribution for 10 (9 Sincs + Constant), 17 (16 Sincs + Constant) and 26 (25 Sincs + Constant) Sized Bases.

Figure 5.5 shows the center distribution for the Sincs in the case of the 10-sized, 17-sized and 26-sized bases. Notice that the base size is a squared number plus a constant term to account for the mean. Also, the distance among any two pair of centers is constant for each case (this was the optimized parameter).

Technically, the optimized parameter could give $\boldsymbol{c}_l$ located even outside the region of interest in the (ϵ_1, ϵ_2) space but by doing this only a small influence on the approximation error was observed (a fraction of decibel difference in the ξ value compared to the distributions shown in Figure 5.5).

For the set of Sinc functions of size 10 (nine Sincs plus a constant term) the obtained cumulative ξ value per ordered basis function was:

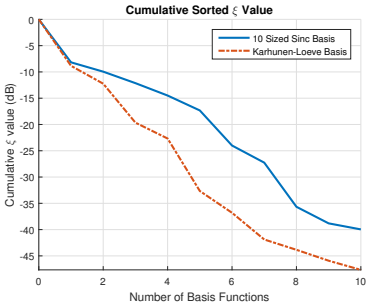

Figure 5.6: Cumulative Sorted ξ Value for 10 Sized Sinc Basis.

In Figure 5.6, it is shown that a performance of half the best power approximation error (by comparing its value in decibels against Figure 5.2) was obtained by using approximately half the number of basis functions. This is not the Karhunen-Loève performance but could be sufficiently good for the purpose of approximation.

5.4 Selecting the Size of the Base

It is reasonable to think that according to the size of the basis functions set, the approximation error is improved. In effect, the following plots show the ξ value is improved for sets basis functions of size 17 (16 Sincs plus a constant term) and 26 (25 Sincs plus a constant term):

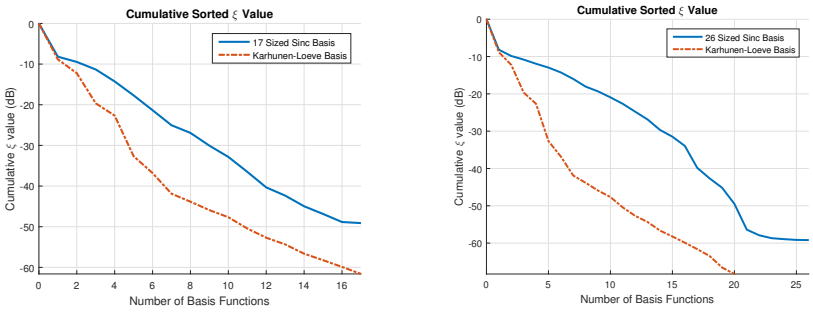

Figure 5.7: Cumulative Sorted ξ Value for 17 (left) and 26 (right) Sized Sinc Bases.

It is worth noting that due to the conditions on the derivation of the ξ measure (see Section 5.3) the set of basis functions were orthogonalized and normalized for the sake of approximation performance evaluation. This means that there is no direct correspondence between the Sinc functions and the functions in the x label of Figure 5.6 and Figure 5.7. Nevertheless, this is not an issue as what matters is the space spanned and such a space is representation-invariant.

Chapter 6
Simulator

Based on the system model of Section 2.3, a simulation program was built on Matlab® in order to show how through functional approximation the reduced complexity version of (4.4) can be built.

The following table shows the name assigned to every parameter in the simulator, its value for every of the two designed simulation setups and a short description of the parameter:

Simulation Parameters			
Name	**Value Setup 1**	**Value Setup 2**	**Description**
L	2		Number of users
N_{FFT}	32	128	FFT size (determines also the time and frequency resolution for an OFDM symbol)
PDP type	Constant along Channel		Power delay profile for simulated environment with $\sum \mathbf{p}_{\boldsymbol{h}} = N_{\mathrm{FFT}}$
$\dim\left(\mathbf{p}_{\boldsymbol{h}}\right)$	7 samples	15 samples	Channel length
T_{m}	4	8	Channel mean delay in samples [13].
S_{τ}	2	4.32	Channel RMS Delay Spread in samples [13].
Pilot symbol distribution	See Figure 2.3		Pilot symbol distribution in the frequency domain for users 1 and 2
Pilot symbol constellation	4-QAM		Constellation type and size for pilot symbols
Continues on next page			

Continuation of Simulation Parameters			
Name	**Value Setup 1**	**Value Setup 2**	**Description**
$N_{\text{realizations}}$	5000		Number of realizations used for likelihood correlation estimation
$\text{RES}_{\epsilon_1,\epsilon_2}$	31×31	51×51	Resolution in the (ϵ_1, ϵ_2) space for simulation purposes
Family of functions	Sinc based		Family of functions for the basis
Optimization interval for γ	$(0, 6]$		Optimization interval for γ parameter in Sinc functions. See (5.7)
Optimization interval for minimum distance among $\boldsymbol{c}_l$ pairs	$\left[\frac{1.5}{\sqrt{L-1}}, \frac{2.5}{\sqrt{L-1}}\right]$		Optimization interval for inter-center distance parameter. L stands for size of the base. See (5.7)
Orthogonalization algorithm	Gram-Schmidt		Orthogonalization algorithm used when ξ value computation required. Modified (stabilized) version is generally used for improved performance in finite precision arithmetic.

Table 6.1: Simulation Parameters for Setup 1 and 2.

Recalling from Section 4, the estimation problem can be expressed as

$$\left(\hat{\epsilon}_1^{\text{ML}}, \hat{\epsilon}_2^{\text{ML}}\right) = \underset{\epsilon_1,\epsilon_2}{\arg\min} \left\{ \log\left(|\Sigma(\epsilon_1, \epsilon_2)|\right) + \boldsymbol{s}^H \Sigma(\epsilon_1, \epsilon_2)^{-1} \boldsymbol{s} \right\}, \qquad (4.4)$$

where the covariance matrix $\Sigma(\epsilon_1, \epsilon_2)$ determinant and inverse calculations require a high amount of resources per (ϵ_1, ϵ_2) evaluation. Specifically, the covariance matrix is given by

$$\Sigma(\epsilon_1, \epsilon_2) = \sum_{k=1}^{2} D(\epsilon_k) Q P_k Q^H \text{diag}(\mathbf{p}_{\boldsymbol{h}}) Q P_k^H Q^H D(\epsilon_k)^H + I_{N_{\text{FFT}}} N_0. \quad (2.5)$$

It can be noticed in (2.5) the dependence on $\text{diag}(\mathbf{p}_{\boldsymbol{h}})$. As the delay spread of the channel is in general much shorter than the symbol time (i.e. $\dim(\mathbf{p}_{\boldsymbol{h}}) \ll N_{\text{FFT}}$), there is a way to reduce the complexity of the covariance determinant and inversion.

Covariance Determinant/Inversion Computational Complexity Reduction

Using the Woodbury matrix identity the inverse of the $N_{\text{FFT}} \times N_{\text{FFT}}$ sized covariance matrix can be replaced by the inverse of two $N \times N$ matrices for $N = dim(\mathbf{p_h})$ according to:

$$\big(A + BCD\big)^{-1} = A^{-1} - A^{-1}B\big(C^{-1} + DA^{-1}B\big)^{-1}DA^{-1}$$

Via an iterative algorithm over (2.5), assuming:

- $A = I_{N_{\text{FFT}}} N_0$
- $B = D\big(\epsilon_1\big)QP_1Q^HX$
- X being any matrix such that $XX^H = \text{diag}\big(\mathbf{p_h}\big)$
- $C = I_{\text{dim}\,\left(\mathbf{p_h}\right)}$
- $D = B^H$

Afterwards, updating according to:

- A as the result of the previous iteration.
- B, C, and D as previously but this time for ϵ_2 and P_2

The inverse of the covariance matrix corresponds to the result from the last iteration.

In exactly the same way the determinant of the big covariance matrix can be replaced by the determinant of two much smaller matrices given that the inverses from the last algorithm are also available. Specifically, the determinant can be computed iteratively through:

$$\big|A + BCD\big| = \big|C^{-1} + DA^{-1}B\big|\,\big|C\big|\,\big|A\big|$$

Where $\big|A\big|$ stands for the determinant of matrix A and the two step process is repeated with the same matrix definitions as for the Woodbury matrix identity.

6.1 Simulation Setup

The problem of evaluating the set of basis functions estimation performance was subdivided into three steps (according mainly to what was explained in Section 5) with some additional implementational considerations.

It is important to differentiate what belongs to the algorithmic design stage and what would actually be implemented at the receiver side. Most of the following steps correspond to the algorithm design and only parts of the third step (Section 6.1.3) and Section 6.2 correspond to the receiver's implementation.

6.1.1 Database Generation

In order to obtain a good estimate of the likelihood correlation matrix, thousands of realizations are obtained and, along with (5.4), the correlation and covariance are estimated.

It is worth noting here that as we are always departing from the continuous frequency offset likelihood function, some considerations regarding sampling interval and its limitations should be kept in mind:

- There is a estimation performance dependency on the condition number of the likelihood correlation matrix.
- The size of the correlation (and hence covariance) matrix is fixed by the resolution in the (ϵ_1, ϵ_2) space.
- The resolution in the (ϵ_1, ϵ_2) space for simulation implies a trade-off in speed and accuracy for the implemented algorithms. However, this applies only for the algorithm design stage and does not imply a major increase in complexity for the receiver end.

6.1.2 Parameter Optimization

Based on the selection of the basis function family, its parameters are optimized via an exhaustive search on the ξ value (see Sections 5.3 and 5.4).

Basically, the idea is to find the set of parameters that minimize the value Figure 6.1.

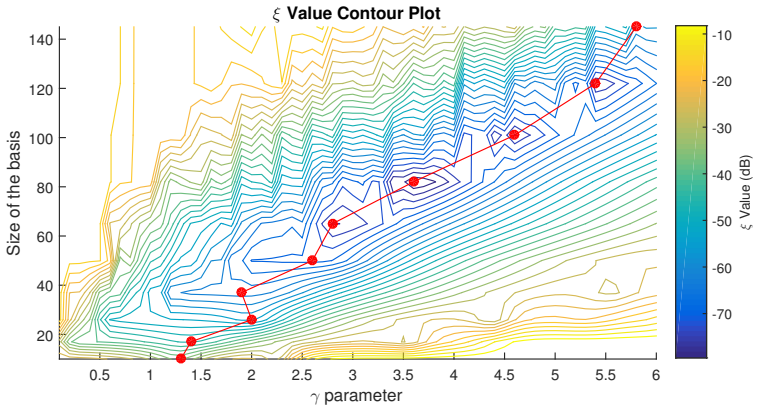

Figure 6.1: Contour Plot of the ξ Value for Basis Parameter Optimization

Figure 6.1 shows a contour plot of the ξ value in terms of γ and the size of the basis (here the $\boldsymbol{c}_k$ parameters were already set to their optimum value for each case, see equation 5.7). Also, the optima are shown for 10-sized, 17-sized, 26-sized, etc. basis in red color. As it would be expected, the optimum γ (reciprocal of the Sinc 'bandwidth') increases with the number of functions in the basis in order to cover the available (limited) grid.

6.1.3 Performance Validation via Monte-Carlo Simulation

Lastly, in order to measure the estimation performance of the final algorithm, several thousands of channel+system simulations are generated giving a high amount of likelihood realizations.

By knowing the frequency offset used to construct the received signal, minimization according to (4.4) is carried out over the true likelihood function and different approximations of it. Afterwards, RMS estimation performance for each method is measured by its empirical mean (this is known as the Monte-Carlo method). Then, comparisons of the net estimation performance for the maximum likelihood estimator versus its low dimension approach are made.

6.2 Considerations for the Receiver's Algorithm

The ξ value over which the parameters are optimized assumes sampling at exactly the positions determined by the grid in the (ϵ_1, ϵ_2) space (see parameter optimization in Section 6.1.2 and derivation of ξ value in Section 5.3). The idea of the likelihood approximation, on the other hand, is to require way less likelihood evaluations at the receiver end.

In order to be able to make use of the optimized parameters accurately, a transformation matrix T from the reduced space of true likelihood evaluations to the space of likelihood decomposition coefficients (namely, α space in equation 5.1) is defined, particularly

$$\widehat{\boldsymbol{\alpha}} = T\boldsymbol{\lambda}_{\text{ss}}, \tag{6.1}$$

where $\boldsymbol{\lambda}_{\text{ss}}$ is a subset of $\boldsymbol{\lambda}$. As a result of (6.1), in order not to have an underdetermined system of linear equations, **the required amount of likelihood evaluations at the receiver side is bounded to be at least equal to the number of base functions**.

Particularly, the idea of (6.1) is to represent such a linear transformation that minimizes the mean square approximation error (MMSE), i.e.:

$$T_{\text{opt}} = \arg\min_{T} \mathbb{E}_{\boldsymbol{\lambda}}\{\|\boldsymbol{\lambda} - \Phi\widehat{\boldsymbol{\alpha}}\|^2\}. \tag{6.2}$$

The transformation represented by the matrix T aims to achieve this. It is worth noting that we are specially interested in the case where $\dim(\boldsymbol{\alpha}) = \dim(\boldsymbol{\lambda}_{\text{ss}})$ (i.e. T being a square matrix).

T Transformation Matrix Derivation

In order to derive an expression for the optimal T transformation in (6.2), the following is assumed to be known:

1. Let's define $\boldsymbol{\lambda}_{\text{ss}}$ as the subset of $\boldsymbol{\lambda}$ obtained by selecting some components from the latter according to an extraction matrix S. Specifically

$$\boldsymbol{\lambda}_{\text{ss}} \overset{\text{def}}{=} S\boldsymbol{\lambda},$$

where S is a sparse 'fat' full row rank matrix with only one non-zero

component per row which extracts the corresponding value from λ.

2. It can be easily verified that $\|\boldsymbol{a}-\boldsymbol{b}\|^2 = \boldsymbol{a}^\mathsf{T}\boldsymbol{a} - 2\mathrm{tr}(\boldsymbol{b}\boldsymbol{a}^\mathsf{T}) + \boldsymbol{b}^\mathsf{T}\boldsymbol{b}$.
3. The trace operator can be applied to any scalar not affecting the result. Furthermore, products inside the trace can be circularly shifted.
4. Recalling the likelihood correlation definition $R_{\boldsymbol{\lambda}} = \mathbb{E}\{\boldsymbol{\lambda}\boldsymbol{\lambda}^\mathsf{T}\}$, by the previous properties of the trace we get: $\mathbb{E}\{\boldsymbol{\lambda}^\mathsf{T}\boldsymbol{\lambda}\} = \mathrm{tr}(R_{\boldsymbol{\lambda}})$.
5. Using $\boldsymbol{\nabla}$, the matrix gradient operator over a scalar function $f(\chi)$ can be denoted according to $\boldsymbol{\nabla}_\chi f(\chi) = \frac{\partial f(\chi)}{\partial \chi}$ where χ is a matrix of any size.
6. Recall that for any convex smooth (continuous gradient) matrix-valued function $f(\chi)$, its minimum can be obtained by solving $\boldsymbol{\nabla}_\chi f(\chi) = 0$ for χ.

   Proving that the error criterion function defined here is globally convex is left to the reader.

First, let's define and expand the error criterion according to (6.2) as

$$\begin{aligned}\varepsilon(T) &\stackrel{\text{def}}{=} \mathbb{E}_{\boldsymbol{\lambda}}\{\|\boldsymbol{\lambda} - \Phi\widehat{\boldsymbol{\alpha}}\|^2\},\\ &= \mathbb{E}_{\boldsymbol{\lambda}}\{\|\boldsymbol{\lambda} - \Phi T\boldsymbol{\lambda}_{\text{ss}}\|^2\},\\ &= \mathbb{E}_{\boldsymbol{\lambda}}\{\|\boldsymbol{\lambda} - \Phi TS\boldsymbol{\lambda}\|^2\}, \qquad \text{(Using 1.)}\end{aligned}$$

$$\begin{aligned}\varepsilon(T) = \mathbb{E}_{\boldsymbol{\lambda}}\{\boldsymbol{\lambda}^\mathsf{T}\boldsymbol{\lambda}\} &- 2\,\mathrm{tr}\Big(\mathbb{E}_{\boldsymbol{\lambda}}\{\boldsymbol{\lambda}\boldsymbol{\lambda}^\mathsf{T}\}\Phi TS\Big)\\ &+ \mathrm{tr}\Big(\mathbb{E}_{\boldsymbol{\lambda}}\{\boldsymbol{\lambda}\boldsymbol{\lambda}^\mathsf{T}\}S^\mathsf{T}T^\mathsf{T}\Phi^\mathsf{T}\Phi TS\Big), \qquad \text{(Using 2. and 3.)}\end{aligned}$$

$$\varepsilon(T) = \mathrm{tr}(R_{\boldsymbol{\lambda}}) - 2\,\mathrm{tr}\Big(SR_{\boldsymbol{\lambda}}\Phi T\Big) + \mathrm{tr}\Big(SR_{\boldsymbol{\lambda}}S^\mathsf{T}T^\mathsf{T}\Phi^\mathsf{T}\Phi T\Big). \quad \text{(Using 4.)}$$

Then, in order to find the optimum transformation, it should be found a T such that

$$\boldsymbol{\nabla}\,\varepsilon(T)\big|_{T_{\text{opt}}} = 0_{\dim(T)}.$$

Hence, the minimum error according to the just defined measure is found using 5. and 6., by

$$\begin{aligned}\boldsymbol{\nabla}\,\varepsilon(T) &= -2\Phi^\mathsf{T}R_{\boldsymbol{\lambda}}S^\mathsf{T} + 2\Phi^\mathsf{T}\Phi TSR_{\boldsymbol{\lambda}}S^\mathsf{T},\\ \therefore T_{\text{opt}} &= \big(\Phi^\mathsf{T}\Phi\big)^{-1}\Phi^\mathsf{T}R_{\boldsymbol{\lambda}}S^\mathsf{T}\big(SR_{\boldsymbol{\lambda}}S^\mathsf{T}\big)^{-1},\end{aligned}$$

where $\big(\Phi^\mathsf{T}\Phi\big)^{-1}\Phi^\mathsf{T}$ could also be written as Φ^+, i.e. the pseudo-inverse of the basis function matrix Φ.

By the use of the linear transformation T, if only the components of $\boldsymbol{\lambda}$ extracted by the matrix S are actually computed at the receiver side, then the expression in (6.1) gives us the optimum $\widehat{\boldsymbol{\alpha}}$ estimator according to the criterion in (6.2). Therefore, enabling us to express the likelihood function approximation in (5.1) based on the few computed likelihood evaluations previously denoted by $\boldsymbol{\lambda}_{\text{ss}}$ as:

$$\widehat{\lambda}(\epsilon_1, \epsilon_2) = \boldsymbol{\lambda}_{\text{ss}}^{\mathsf{T}} T^{\mathsf{T}} \boldsymbol{\phi}(\epsilon_1, \epsilon_2). \tag{6.3}$$

Here again, bold letters denote column vectors. Specifically, $\boldsymbol{\phi}(\epsilon_1, \epsilon_2)$ refers to the vector formed by appending the basis functions evaluated at (ϵ_1, ϵ_2) in a column vector fashion. Also of great importance for the use of gradient dependent optimization algorithms in ML estimation, the gradient for the likelihood approximation is straightforward.

By (6.3) the gradient of such a function can be expressed analytically in terms of the Jacobian matrix of $\boldsymbol{\phi}(\epsilon_1, \epsilon_2)$, denoted by $J_{\boldsymbol{\phi}}(\epsilon_1, \epsilon_2)$, as

$$\boldsymbol{\nabla}\widehat{\lambda}(\epsilon_1, \epsilon_2) = J_{\boldsymbol{\phi}}(\epsilon_1, \epsilon_2)\, T \boldsymbol{\lambda}_{\text{ss}}, \tag{6.4}$$

where it can be seen that the complexity of evaluation, either for the likelihood approximation or its gradient, depends mostly on the selected set of basis functions.

6.3 Results

Based on the simulation setup (Section 6.1), a Monte-Carlo simulation is built in order to measure the actual estimation performance of the proposed method compared to joint and independent ML estimation (see Section 4).

Particularly, performance simulations are carried over 10-sized, 17-sized, 26-sized and 37-sized bases for setups 1 and 2 (see Table 6.1). Recall that the number of functions in the base is a square number plus one due to the constant term accounting for the mean of the likelihood function.

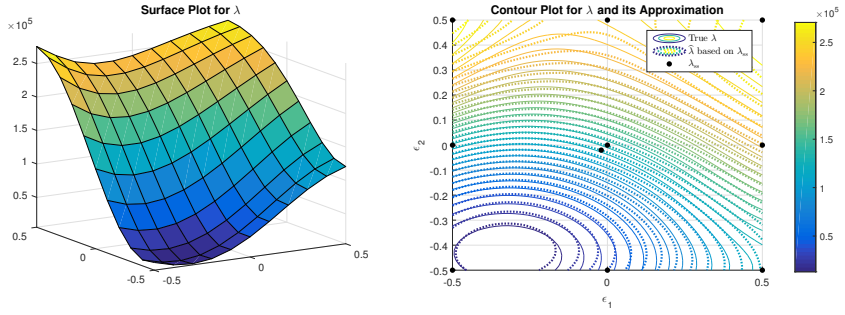

Figure 6.2: Surface and Contour Plot for λ and its Approximation. Basis: 10-Sized Sinc.

Figures 6.2 and 6.3 show on the left a surface plot for different realizations of the likelihood function at 10 dB of SNR (see Section 2.3 for SNR derivation). Furthermore, at the right of each figure, it is shown the contour plot for the

corresponding likelihood realization and its approximation based on the algorithm developed throughout this project.

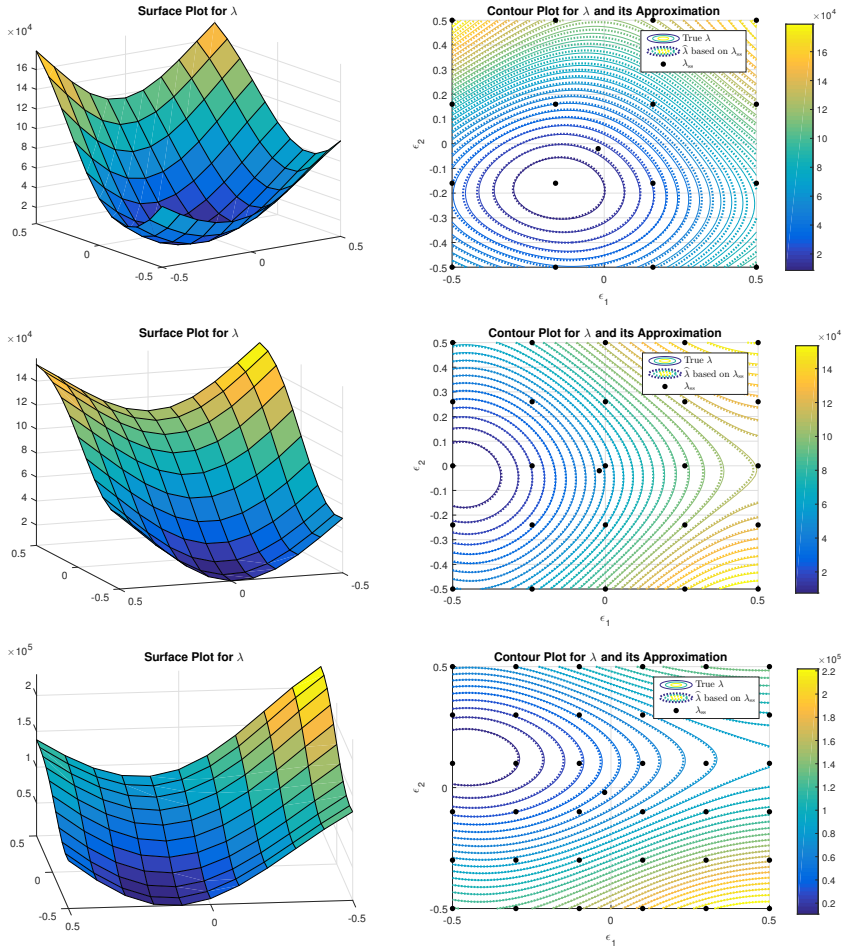

Figure 6.3: Surface and Contour Plot for λ and its Approximation. Basis: 17-Sized (top), 16-Sized (middle) and 37-Sized (bottom) Sinc.

The behavior observed in Figures 6.2 and 6.3 is just a pictorial confirmation of the expected results from the analysis using the ξ value on Section 5.4 (see Figures 5.6 and 5.7). It is easy to see how the bigger the basis, the better the approximation. Nonetheless, it must be noted that along with the size of the basis, the required number of evaluations to the likelihood function grows respectively.

However, as mentioned in Section 5, the ξ value was only a first approach when looking for a performance measure. The latter, due to the fact that an analytic expression can be derived for it (see ξ value derivation in Section 5.3).

On the other hand, because it would be rather complicated to find an expres-

sion for the estimation performance of the proposed method, it must be evaluated through Monte-Carlo simulation. In order to do so, many realizations of the likelihood process are to be generated. Then, an optimization method is to be used over the likelihood and its approximations and, finally, the root mean square estimation error is to be approximated through its empirical mean for different SNR.

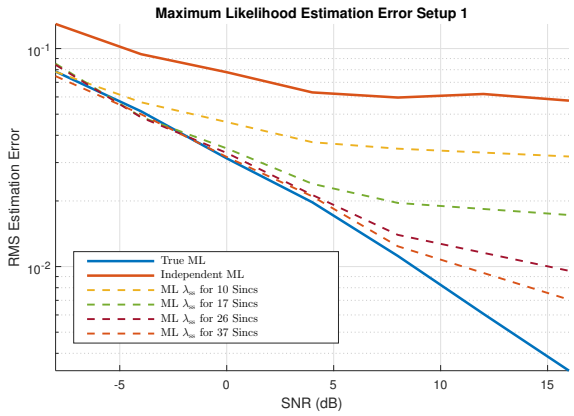

Figure 6.4: RMS Estimation Error for Independent ML, ML and Several Likelihood Approximations. Setup 1

Figure 6.4 shows the joint ML performance in solid blue color for setup 1. Also, independent estimation performance is shown (in solid orange color). As it would be expected, the methods giving use of approximations to the likelihood lie in-between (performance wise) the joint and independent ML.

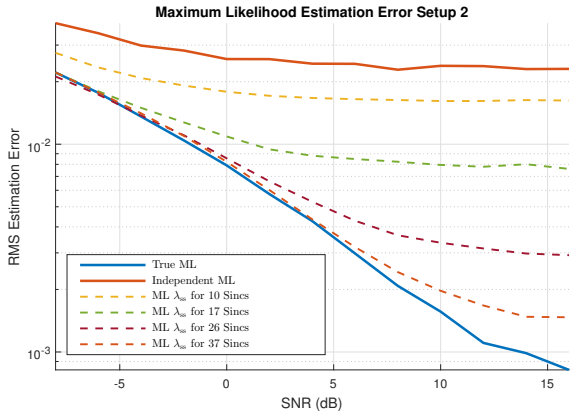

Figure 6.5: RMS Estimation Error for Independent ML, ML and Several Likelihood Approximations. Setup 2

Meanwhile, Figure 6.5 shows the same simulation for setup 2. Both, simulation for setup 1 and setup 2, exhibit a saturation on the approximation-oriented

estimations (see dashed lines) depending on the size of the base.

Saturation in the logarithmic scale is a behavior exhibited due to the residual non-captured energy of truncated functions and explains why this methods is powerful when aiming reduce computational complexity on constrained systems. As an example, if there is no need in reducing frequency offset below 10^{-2} using setup 2 according to some criteria, it would be a waste of resources (not to mention probably unfeasible) to implement joint ML estimation; still, independent ML estimation would not be good enough. Hence, using the proposed method for 17-Sized basis (see Figure 6.5) would be a more appropriate solution.

In general, considering the reduced computational complexity and increased performance of the proposed method compared to joint ML and independent ML estimation (respectively), Figures 6.4 and 6.5 clearly exhibit the performance-complexity trade-off among these two methods via the proposed one.

Finally, in order to have a better insight on what happens when the SNR is modified (from a functional decomposition point of view), Figure 6.6 shows the average power contribution per basis function for setup 1 (up to the first 5 functions in the orthogonalized Sinc basis).

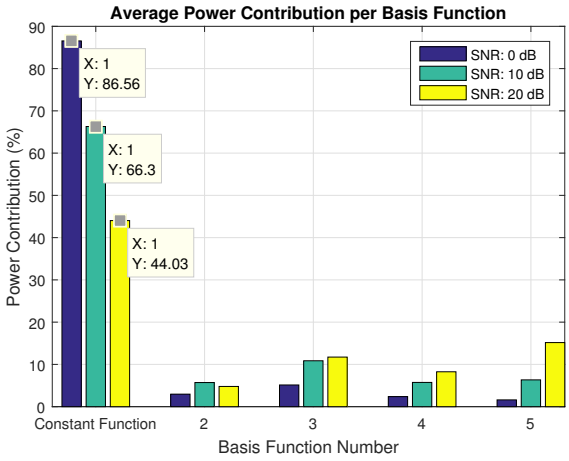

Figure 6.6: Likelihood Process Average Power Contribution per Basis Function for Different SNR

Figure 6.6 can be seen as the power spectral density (PSD) when decomposing the likelihood function using the Sinc basis for the first 5 orthonormal components. This way, it is possible to identify some of the most contributing components. Particularly, from the plot it can be seen an inverse relation between the constant function contribution and the SNR.

The fact that most of the likelihood's power is concentrated on the constant function (which is the common component among all-sized bases) at low SNR explains why as the SNR increases, the proposed method error deviates from the true ML curve (which is our benchmark). Also, it explains that by increasing the size of the basis, on the high SNR, the saturation level is controlled due to the better ability of the approximation to capture the residual non-constant power.

Chapter 7

Conclusions and Future Work

To conclude it is important to note that the goal on the development of the algorithm was accomplished. In particular, the achieved likelihood function's low dimensional approximation allowed us to control the complexity-performance trade-off between the independent and joint ML estimators.

In order to draw a comparison to the developed method throughout this thesis, Table 7.1 summarizes the main differences among the studied methods and the proposed one. K stands for the number of users at the transmitter side, for our case $K = 2$.

ML Estimation Method Comparison				
Method	**Likelihood Evaluation Complexity**	**Optimization Complexity**	**Performance**	**Notes**
Independent ML	$O\left(N_{\mathrm{FFT}}^3\right)$	K searches on one dimension each	Worst among studied	It is possible to estimate only one frequency offset.
Proposed Method	Determined by basis (Overhead due to α computation)	Search on K dimensions	Determined by basis	Analytic gradient and Hessian available at no cost.
Joint ML	$O\left(N_{\mathrm{FFT}}^3\right)$	Search on K dimensions	Best among studied	Gradient hard to pre-compute, complex to estimate on-line.

Table 7.1: Estimation Methods Comparison. K stands for the number of users at the transmitter side.

Even though the number of users assumed along this project was restricted to two, the derivations upon which the proposed method relies on generalizes to the general case of K users (having in mind the problem grows in complexity according to what was summarized in Table 7.1).

As observed in Table 7.1, the main benefit of the proposed low dimension approximation method lies on the possibility of controlling performance and complexity of estimation through the selected decomposition basis (see Section 5.3).

It must be noted that the approximated function used by the proposed method is not globally convex and, particularly, not convex around its optimum. The latter, because it approximates the joint likelihood which by nature exhibits a multi-modal behavior.

Most importantly, the non-convex region around the optima determines the estimation error limit (and hence the performance for the proposed method in Table 7.1) controlled via the size of the basis function on the implemented simulator.

Likewise, complexity was controlled by the size of the basis function as it determines the minimum number of likelihood evaluations required, as seen in Section 6.2.

Finally, **two things were thought of as future work**. First, keep in mind that the constant function has such an strong influence on the approximation (specially on the low SNR regime). Thus, ideally there should be a way to precompute the constant term based on the model used. In particular, by obtaining the mean of the optimized function in (4.4), a better fit of the proposed method's estimation performance in Figures 6.4 and 6.5 would be expected. Unfortunately, this would not affect the saturation level as the constant term influence reduces when the SNR is increased (see Figure 6.6).

Second and last, what influences the approximation the most is the richness of the base and the density of evaluations in the (ϵ_1, ϵ_2) space. Hence, an extension to the originally proposed method can be envisaged. The idea would be to increase the density of evaluations around the optimum by conceiving a successive approximation algorithm. Therefore, the modification would consist of approximating the likelihood, optimizing, then approximating around the estimate and repeating the process. A representation of the first successive approximation is shown in Figure 7.1.

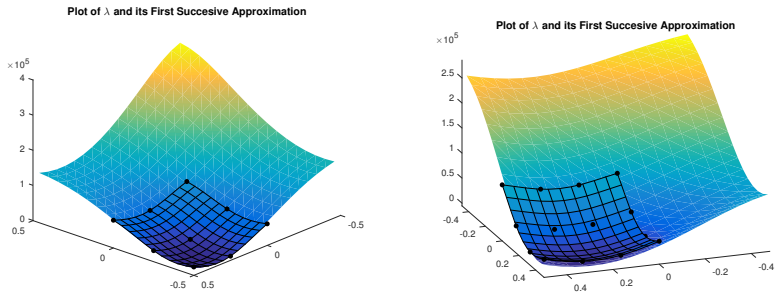

Figure 7.1: λ and its First Succesive Approximation for 10-Sized (left) and 17-Sized (right) Sinc Bases.

References

[1] P. H. Moose, "A technique for orthogonal frequency division multiplexing frequency offset correction," *IEEE Transactions on Communications*, vol. 42, pp. 2908–2914, Oct 1994.

[2] J. Van de Beek, M. Sandell, P. O. Borjesson, *et al.*, "ML estimation of time and frequency offset in OFDM systems," *IEEE transactions on signal processing*, vol. 45, no. 7, pp. 1800–1805, 1997.

[3] R. Fantacci, D. Marabissi, and S. Papini, "Multiuser interference cancellation receivers for OFDMA uplink communications with carrier frequency offset," in *Global Telecommunications Conference, 2004. GLOBECOM '04. IEEE*, vol. 5, pp. 2808–2812 Vol.5, Nov 2004.

[4] F. Rusek and B. Priyanto, "Method and receiver in a wireless communication system," July 14 2015. US Patent 9,083,597.

[5] R. G. Gallager, "Circularly-symmetric Gaussian random vectors," *preprint*, pp. 1–9, 2008.

[6] A. Hyvarinen, J. Karhunen, and E. Oja, *Independent component analysis*. Adaptive and learning systems for signal processing, communications, and control, New York, Chichester, Weinheim: John Wiley, 2001.

[7] N. Pham, A. Malinowski, and T. Bartczak, "Comparative study of derivative free optimization algorithms," *IEEE Transactions on Industrial Informatics*, vol. 7, pp. 592–600, Nov 2011.

[8] P. Belitz and T. Bewley, "Efficient derivative-free optimization," in *Decision and Control, 2007 46th IEEE Conference on*, pp. 5358–5363, Dec 2007.

[9] A. Levey and M. Lindenbaum, "Sequential Karhunen-Loève basis extraction and its application to images," *IEEE Transactions on Image Processing*, vol. 9, pp. 1371–1374, Aug 2000.

[10] H. M. Abbas and M. M. Fahmy, "Neural model for Karhunen-Loève transform with application to adaptive image compression," *IEE Proceedings I - Communications, Speech and Vision*, vol. 140, pp. 135–143, Apr 1993.

[11] P. G. Howard, "Lossless and lossy compression of text images by soft pattern matching," in *Data Compression Conference, 1996. DCC '96. Proceedings*, pp. 210–219, Mar 1996.

[12] A. Alexanderian, "A brief note on the Karhunen-Loève expansion," *arXiv preprint arXiv:1509.07526*, 2015.

[13] A. F. Molisch, *Wireless communications*. John Wiley & Sons, 2007.

LUND
UNIVERSITY

Series of Master's theses
Department of Electrical and Information Technology
LU/LTH-EIT 2016-498

http://www.eit.lth.se