# BLIND SIGNAL SPATIAL SIGNATURE ESTIMATION USING PARAFAC MODEL

By

Yue Rong

Thesis submitted to the faculty of Electrical Engineering
Gerhard Mercator University, Duisburg, Germany
in partial fulfillment of the requirements of the degree

Master of Science
Computer Science and Communications Engineering

Supervisors:

Prof. Dr. -Ing. Alex B. Gershman
Priv. -Doz. Dr. -Ing. Thomas Kaiser
Prof. Dr. -Ing. Andreas Czylwik

Department of Communication Systems
Gerhard Mercator University
Bismarckstrasse 81, 47057, Duisburg, Germany

October 2002
Duisburg, Germany

# Table of Contents

# List of Tables

# List of Figures

# Glossary of Symbols and Abbreviations

## Symbols

| | |
|---|---|
| $(\cdot)^T$ | transpose |
| $(\cdot)^H$ | Hermitian (conjugate transpose) |
| $(\cdot)^*$ | conjugate |
| $(\cdot)^\dagger$ | pseudo inverse |
| $\otimes$ | Kronecker product |
| $\odot$ | Khatri-Rao product |
| $\mathrm{diag}(\mathbf{v})$ | make from vector $\mathbf{v}$ a diagonal matrix |
| $\mathbf{D}_i(\mathbf{X})$ | make a diagonal matrix from the $i$-th row of matrix $\mathbf{X}$ |
| $\mathbf{\Pi}$ | permutation matrix |
| $\sigma^2_{\mathbf{s}}$ | signal power |
| $\sigma^2_{\mathbf{v}}$ | noise power |
| $\widehat{\mathbf{A}}$ | estimation of $\mathbf{A}$ |
| E | expected value |
| $f_{\mathrm{PCF}}$ | power change factor |
| $\mathbf{M}$ | signal amplitude matrix |
| $\mathbf{P}$ | power matrix |
| $\mathbf{R}_{\mathrm{s}}$ | covariance matrix of source signal |
| $\mathbf{R}_{\mathrm{s}(p)}$ | $p$-th covariance matrix of source signal |
| $\mathbf{R}_{\mathrm{y}}$ | covariance matrix of the received signal at antenna array |
| $\mathbf{R}_{\mathrm{y}(p)}$ | $p$-th covariance matrix slab of the received signal at antenna array |

| | |
|---|---|
| $\mathbf{R}_{\mathrm{y}}(a), \mathbf{R}_{\mathrm{y}}(b), \mathbf{R}_{\mathrm{y}}(c)$ | unfolded matrices of $\underline{\mathbf{R}}_{\mathrm{y}}$ |
| $\underline{\mathbf{R}}_{\mathrm{y}}$ | three way received signal covariance array |
| $\mathbf{V}$ | noise matrix |
| $\underline{\mathbf{X}}$ | three way array |
| $\mathbf{X}_{:,:,k}$ | $k$-th slab of three way array |
| $\mathbf{Y}$ | received signal matrix |

## Abbreviations

| | |
|---|---|
| AWG | additive white Gaussian |
| AMSE | average mean square error |
| BS | base station |
| CDMA | code division multiple access |
| CGRAM | compressed generalized rank annihilation method |
| D-CRB | deterministic Cramér-Rao bound |
| DOA | direction of arrival |
| DTLD | direct trilinear decomposition |
| ESPRIT | estimation of signal parameters via rotational invariance techniques |
| EVD | eigen value decomposition |
| ICI | inter-chip interference |
| ISI | inter-symbol interference |
| JADE | joint approximate diagonalization |
| LCMP | linear constrained minimum power |
| MAI | multiuser access interference |
| MF | matched filter |
| ML | maximum likelihood |
| MS | mobile station |
| MUSIC | multiple signal classification |
| MVDR | minimum variance distortionless response |
| PCF | power change factor |
| PARAFAC | parallel factor |
| S-CRB | stochastic Cramér-Rao bound |

| | |
|---|---|
| SDMA | spatial division multiple access |
| SNR | signal to noise ratio |
| SS | spatial signature |
| TALS | trilinear alternative least square |
| UCA | uniform circular array |
| ULA | uniform linear array |
| URA | uniform rectangular array |

# Acknowledgements

First I would like to thank Prof. Alex B. Gershman, my supervisor, who introduced me into the wonderful field of array processing. Thanks for many constructive suggestions and discussions during my thesis work. His enthusiasm and devotion to research encourage me greatly.

I would express my gratitude to Dr. Thomas Kaiser, my second supervisor. Thanks for his constant supports. The lectures given by him, TM1, TM2 and IE1 are my favorite lectures. Because of his sincere and genial character, working with him is alway enjoyable.

I would also thank Prof. Andreas Czylwik. I benefit a lot from his knowledge in cellular communications and the Wednesday seminars.

Prof. Nikos Sidiropoulos's papers first introduced me to the world of three way analysis. His expertise in PARAFAC gave me great help.

Sergiy Vorobyov introduced me to the field of JADE and blind signal separation. His knowledge in mathematics makes discussions with him always beneficial. Shahram Shahbazpanahi and Yisheng Xue offer me many different perspectives to signal processing and communication. I express my gratitude here.

I should also mention that my master's studies in Germany were supported by Deutscher Akademischer Austausch-Dienst (DAAD) and ABB company, through program "DAAD/ABB graduate sponsoring initiative Asia 2000 and 2001".

Among all, I'm grateful to my parents. Their patience and their tremendous unconditional *love* are the strongest support to me. Without them this work would never have come to existence.

Finally I wish to thank the following, Petra Hoetger for the administrative work, Feng Xie for the PC software maintenance work, and Zisan Zhang, Jianliang Shi, Yi Zhao, Dachuan Wang, Lei Huang, Guojun Yang for their friendships.

Duisburg, Germany — Yue Rong
Sept. 17, 2002

# Abstract

Demands for mobile communication increase tremendously in recent years. Multi-access by Spatial Division Multiple Access (SDMA), which exploits the spatial diversity of the wireless system, has inspired great research interests. A very important parameter in SDMA is the Spatial Signature (SS) of signals. By using antenna array at the basestation(BS), SS can be estimated. Many algorithms have been developed using training sequences. However, a serious disadvantage of using training sequence is that it reduces the overall transmission rate. Therefore, blind approaches, which don't need training sequences, are desired.

In this thesis, we propose a blind SS estimation algorithm, which exploits the structure of the second order statistics (covariance matrices) of signals. A PARAllel FACtor (PARAFAC) analysis model is used to develop such an algorithm. Identifiability potential of the PARAFAC model is studied. We show that our model can also be solved through another class of method — Joint Approximate Diagonalization (JADE). Numerical simulations are carried out for both PARAFAC and JADE. The results show that PARAFAC clearly outperforms JADE. The comparison with deterministic Cramér-Rao Bound (D-CRB) shows that our algorithm stays close to what is the theoretically best performance-wise.

# Introduction

In recent years, mobile communication becomes more and more popular. On one hand, the number of mobile communication users increases tremendously. On the other hand, services provided through mobile communications have exceeded pure speech and included multimedia data, which leads to the request of higher data rate transmission. Both trends exacerbate the main problem faced in mobile communication: interference. For example, in Code Division Multiple Access (CDMA) systems, the increasing density of users causes more serious Multiuser Access Interference (MAI). The increasing of data rate causes more serious Inter-symbol Interference (ISI) and Inter-chip Interference (ICI) [25]. A well known method to combat interference is diversity technique [19]. One promising way to achieve diversity is to apply antenna arrays, which can achieve spatial diversity. The Spatial Signature (SS) of signals, which is defined as the vector of received electric fields at antenna array, becomes very important in this case. It acts as an important role in Spatial Division Multiple Access (SDMA) systems as the temporal signature in CDMA systems. However, in mobile communication the SS is usually unknown to the receiver and therefore has to be estimated.

This work will focus on the estimation of SS. We consider the scenario of uplink (mobile—base station) SS estimation. First we would like to point out the differences of SS estimation and traditional Direction Of Arrival (DOA) estimation. Generally speaking, SS contains the mixture information of DOA, multi-path effect, wavefront distortion, etc. Therefore, the estimation of SS includes the estimation of DOA as its subset. In [28], the DOAs are determined from the SS. Further on, the number of DOAs which is identifiable

with given SS is discussed in [23].

Many algorithms have been developed to estimate the SS [23],[1], or jointly estimate SS with other parameters [11]. In [23], maximum likelihood (ML) algorithm was developed. A well known disadvantage for ML is that it requires parameter space search and is therefore computationally intensive. Some algorithms are computationally simple than ML algorithm. For example, provided large number of snapshots, matched filter (MF) method can be used [23]. However, it still requires training sequences, which reduces the overall transmission rate. In this thesis we propose a blind method which don't require training sequences. Some methods use parametric methods, where they assume that the geometry of the antenna array is known (usually Uniform Linear Array (ULA) is used). For example, in [1], a closed form ESPRIT-like estimator is proposed. However, one disadvantage of parametric method is that if the geometry of the array is unknown or some calibration errors exist, the performance of estimation will degrade seriously. See [26], [29] for detailed discussions. Our method proposed in this theis, which uses PARAllel FACtor (PARAFAC) model [10],[21],[22], is nonparametric and doesn't require any knowledge of the array geometry. In fact, the array geometry can be arbitrary.

This thesis is organized as follows. Chapter 1 introduces some background knowledge, which are necessary for the development of our algorithm. Chapter 2 presents our main theorem. In this chapter, we first build the discrete-time baseband equivalent data model that we work with. Then we solve our problem by fitting the PARAFAC model. The identifiability potential of the PARAFAC model in our scenario will be discussed. This is followed by showing the algorithm used for fitting PARAFAC model. Besides PARAFAC we also find that our model can be solved by Joint Approximate Diagonalization Estimation (JADE) [2]. We also deal with this in Chapter 2. Numerical simulation results are presented in Chapter 3, where we carry out simulations to verify our algorithm and present some discussions about PARAFAC model. The derivation of deterministic Cramér-Rao Bound

can be found in the appendix.

# Chapter 1

# Background

Since the application of antenna array in mobile communications, great impact has taken place. Antenna array has many advantages over conventional single antenna. It enhances signal power at the output of the antenna array and reduces interference. Therefore, system capacity is increased and communication reliability is improved. We will introduce some basic knowledge about array processing in this chapter.

On the other hand, PARAFAC or more general, three way analysis, is a very general mathematic model and analysis method. Due to its concise and general mathematic formulation, it has attracted many research interests. We will give a short introduction to it in the second half of this chapter.

## 1.1 Array Processing

Antenna array was first invented in military applications [12]. Now it plays an important role in a wide range of areas, such as radar, sonar, radio astronomy, communications, seismology and medical diagnosis. The possibility and benefit of applying antenna array in mobile communication have been studied intensively in the last decade. Due to the physical size and power consumption constraints, antenna arrays are normally installed in

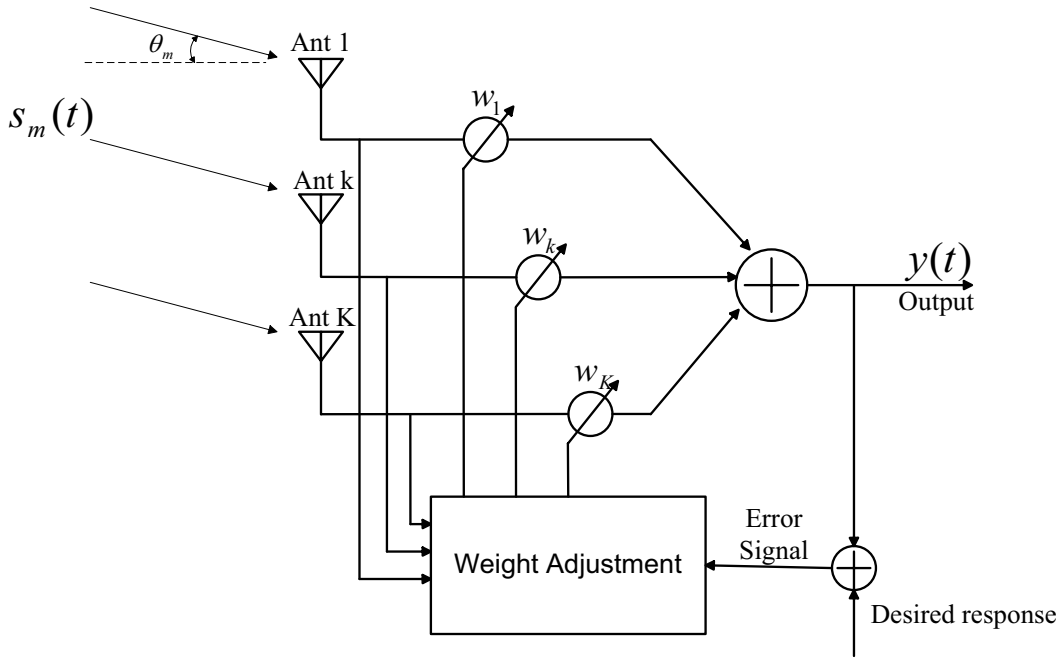


Figure 1.1: Basic diagram of a narrowband adaptive antenna array

base station (BS) [1]. Therefore, we constrain the introduction here only to the uplink (mobile station (MS) to BS) in mobile communication.

In rural and suburb environments, BSs are mounted on the top of high buildings or towers, where there are no obstacles near the antenna array. Normally the distances between MS and BS are large enough to satisfy the far region conditions for transmitting and receiving antennas. If we consider the uplink in mobile communication, the incoming electromagnetic wave at the BS can be treated as plane wave.

In Figure 1.1, we plot the basic diagram of a narrowband adaptive antenna array with $K$ antennas. The weight vector $\mathbf{w} = [w_1, w_2, \ldots, w_K]^T$ can be set based on a certain criteria. The DOA of incoming signals are calculated with respect to the broad side of the array

[1] recently mobile phones installed with 2 or 3 antennas are also available.

(Take the $m$-th signal as an example, whose DOA $\theta_m$ is marked on Figure 1.1). Without loss of generality, we first consider an antenna array with arbitrary manifold and the first antenna as the reference antenna. The signal $x_1(t)$ at the output of the first antenna can be written as

$$x_1(t) = s_m(t) + n_1(t), \tag{1.1}$$

where $s_m(t)$ is the received $m$-th signal, $n_1(t)$ is the noise at the first antenna. Under the plane wave assumption the signal received by the $k$-th antenna is a delayed (phase shift) version of $s_m(t)$.

$$x_k(t) = s_m(t) \exp(j\phi_{k-1}) + n_k(t), \tag{1.2}$$

where $\phi_{k-1}$ denotes the phase shift between the received signal at $k$-th antenna and the first antenna. If we recast the received signals into vector form, we will obtain:

$$\mathbf{x}(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_K(t) \end{pmatrix} = \begin{pmatrix} 1 \\ \exp(j\phi_1) \\ \vdots \\ \exp(j\phi_{K-1}) \end{pmatrix} s_m(t) + \begin{pmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_K(t) \end{pmatrix} = \mathbf{a}_m s_m(t) + \mathbf{n}(t), \tag{1.3}$$

where $\mathbf{a}_m = [1, exp(j\phi_1), \ldots, exp(j\phi_{K-1})]^T$ is called the steering vector or the spatial signature of signal $s_m(t)$.

Normally the sensors in the antenna array are arranged in a regular geometry, such as uniform linear array (ULA), uniform rectangular array (URA) or uniform circular array (UCA). The reason for this is that these geometries are symmetric and offer some redundancy which can be exploited. For example, let's consider here an ULA with $K$ antennas of equal displacement $d$. In this case, the phase shift term in equation (1.2) becomes

$$\phi_{k-1} = \frac{2\pi}{\lambda}(k-1)d\sin\theta_m. \tag{1.4}$$

And the spatial signature of $s_m(t)$ now becomes

$$\mathbf{a}_m = [1, \exp(j\frac{2\pi}{\lambda}d\sin\theta_m), \ldots, \exp(j\frac{2\pi}{\lambda}(K-1)d\sin\theta_m)]^T. \tag{1.5}$$

If the array is required to steer $[-\frac{\pi}{2}, \frac{\pi}{2}]$, we require $d \leqslant \frac{\lambda}{2}$, which is well known as a standard linear array.

Now suppose there are $M$ signals. The received signals at the antenna array is the superposition of $M$ signals

$$\mathbf{x}(t) = \sum_{m=1}^{M} \mathbf{a}_m s_m(t) + \mathbf{n}(t). \tag{1.6}$$

Recast it in matrix form, we have

$$\mathbf{x}(t) = \begin{pmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \dots & \mathbf{a}_M \end{pmatrix} \begin{pmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_M(t) \end{pmatrix} + \mathbf{n}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t), \tag{1.7}$$

where $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_M]$ is the SS matrix, $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_M(t)]^T$ is the signal vector at time $t$. We can easily find that $\mathbf{A}$ in an ULA has Vandermonde structure. This redundancy can be exploited, e.g. in spatial smoothing and sub array processing [24].

If we take a further look into Figure 1.1, we can find that the adaptive array has similarities with a transversal FIR filter. The weighting vector $\mathbf{w}$ is used to carry out spatial filtering. After the filtering operation, we will obtain the output signal

$$y(t) = \mathbf{w}^H \mathbf{x}(t). \tag{1.8}$$

There are two main topics contained in array processing. One is beamforming, where the key problem boils down to find an optimum $\mathbf{w}$, which satisfies certain performance criteria, such as maximum signal to interference plus noise ratio (SINR). Huge amount of researches have contributed to this topic. Further discussions about beamforming is out of the range of this thesis. Refer to [24] and the references therein for detailed discussion.

The problem to be discussed in this thesis belongs to another main topic in array processing—parameter estimation. The parameters we are interested in array processing are DOA, signal numbers, signal spatial signatures, etc.. Again tremendous research have

been done in the last two decades, see [24] and the references therein. This thesis deals with SS estimation.

## 1.2 Parallel Factor Analysis

Consider a three way array, whose element $x_{i,j,k}$ can be formed as

$$x_{i,j,k} = \sum_{f=1}^{F} a(i,f)b(k,f)c(f,j), \tag{1.9}$$

where $a(i,f)$ is the $(i,f)$-th element of a $I \times F$ matrix $\mathbf{A}$, $b(k,f)$ is the $(k,f)$-th element of a $K \times F$ matrix $\mathbf{B}$, $c(f,j)$ is the $(f,j)$-th element of a $F \times J$ matrix $\mathbf{C}^T$. Equation (1.9) expresses that $x_{i,j,k}$ is a sum of triple products. This model is known as trilinear analysis, trilinear decomposition or PARAllel FACtor analysis. This model was first proposed by Harshman in 1970 [10] for data analysis in psychometics. Later on it was also applied in chemometrics for spectrophotometric, chromatographic and flow injection analysis. Sidiropoulos first introduced this method to array signal processing [21]. PARAFAC analysis belongs to a more general analysis method—three way analysis in that there are some constraints between the factors in PARAFAC model. For a tutorial overview to PARAFAC analysis, we can read [4].

To make equation (1.9) more clear, let's represent it in a graphic way in Figure 1.2.

The matrix $\mathbf{X}_{:,:,k}$ can be written as $\mathbf{X}_{:,:,k} = \mathbf{A}\mathbf{D}_k(\mathbf{B})\mathbf{C}^T$. It can be imagined that by cutting the three way "cake" $\underline{\mathbf{X}}$ and take the $k$-th slide we will get $\mathbf{X}_{:,:,k}$. Further on, if we stack all of the $K$ matrix on top of one another, we will obtain a very tall matrix. We denote it as $\mathbf{X}(a)$. See the left part of Figure 1.3. If we repeat the same operation in other two directions, we will obtain another two tall matrices $\mathbf{X}(b)$ and $\mathbf{X}(c)$. See the middle and right part of Figure 1.3. $\mathbf{X}(a), \mathbf{X}(b)$ and $\mathbf{X}(c)$ are called the unfolded matrices of $\underline{\mathbf{X}}$.

Let's define some concepts about "three way" which will be used later in this thesis.

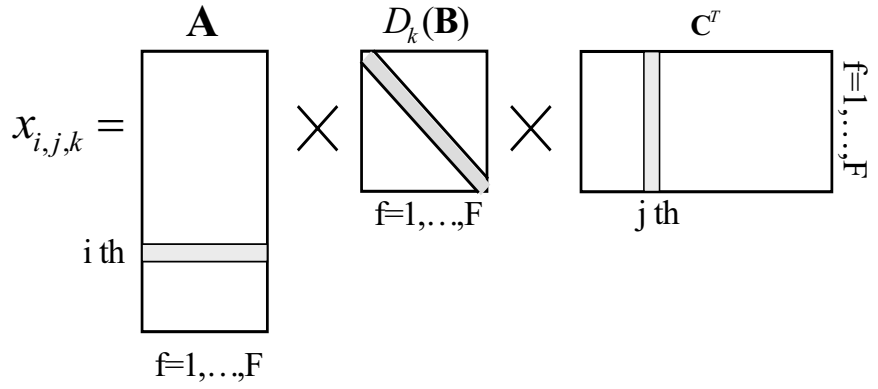


Figure 1.2: $(i, j, k)$-th element of a three way array

- Three way rank: Smallest number of rank-one terms for additive decomposition of a three way array. Basically it shares the same definition as a matrix rank.
- Three way rank-one factors: Rank-one three way array outer product of 3 vectors.

With these definitions, we can find the rank of $\underline{\mathbf{X}}$ is $F$ from equation (1.9).

Let's now derive the closed form of the unfolded matrices

$$\mathbf{X}(a) = \begin{pmatrix} \mathbf{A}\mathbf{D}_1(\mathbf{B})\mathbf{C}^T \\ \mathbf{A}\mathbf{D}_2(\mathbf{B})\mathbf{C}^T \\ \vdots \\ \mathbf{A}\mathbf{D}_K(\mathbf{B})\mathbf{C}^T \end{pmatrix} = \begin{pmatrix} \mathbf{A}\mathbf{D}_1(\mathbf{B}) \\ \mathbf{A}\mathbf{D}_2(\mathbf{B}) \\ \vdots \\ \mathbf{A}\mathbf{D}_K(\mathbf{B}) \end{pmatrix} \mathbf{C}^T = \begin{pmatrix} b_{11}\mathbf{a}_1 & b_{12}\mathbf{a}_2 & \dots & b_{1F}\mathbf{a}_F \\ b_{21}\mathbf{a}_1 & b_{22}\mathbf{a}_2 & \dots & b_{2F}\mathbf{a}_F \\ \vdots & \vdots & \dots & \vdots \\ b_{K1}\mathbf{a}_1 & b_{K2}\mathbf{a}_2 & \dots & b_{KF}\mathbf{a}_F \end{pmatrix} \mathbf{C}^T. \tag{1.10}$$

Each column in the most right matrix is now

$$\begin{pmatrix} b_{1i}\mathbf{a}_i \\ b_{2i}\mathbf{a}_i \\ \vdots \\ b_{Ki}\mathbf{a}_i \end{pmatrix} = \mathbf{b}_i \otimes \mathbf{a}_i. \tag{1.11}$$

Insert equation (1.11) into equation (1.10), we obtain

$$\mathbf{X}(a) = \begin{pmatrix} \mathbf{b}_1 \otimes \mathbf{a}_1 & \mathbf{b}_2 \otimes \mathbf{a}_2 & \dots & \mathbf{b}_F \otimes \mathbf{a}_F \end{pmatrix} \mathbf{C}^T = (\mathbf{B} \odot \mathbf{A})\mathbf{C}^T, \tag{1.12}$$

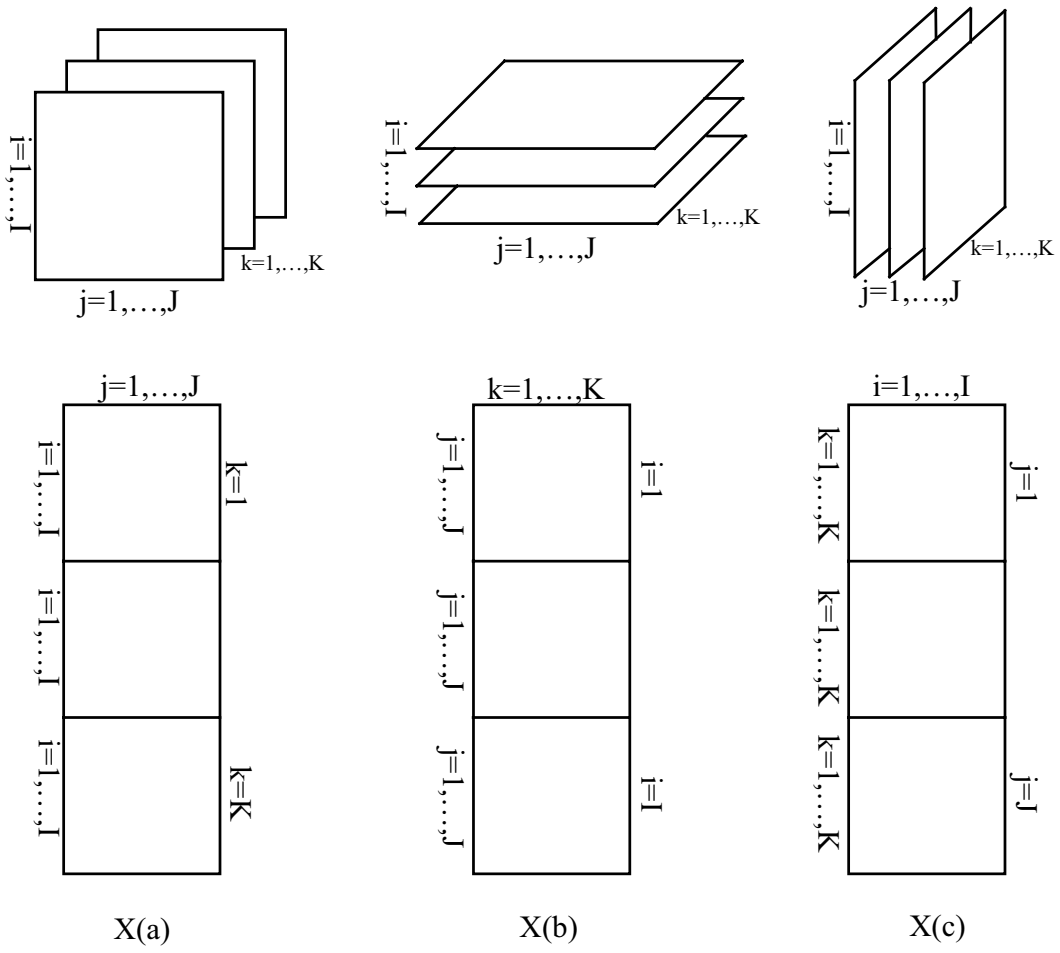


Figure 1.3: Cut three way array and form tall matrices

where $\odot$ denotes Khatri-Rao product, which is column-wise Kronecker product. Some properties about Khatri-Rao product can be found in [3]. In the same way we can obtain the closed form solution for $\mathbf{X}(b)$ and $\mathbf{X}(c)$

$$\mathbf{X}(b) = (\mathbf{A} \odot \mathbf{C})\mathbf{B}^T \tag{1.13}$$

$$\mathbf{X}(c) = (\mathbf{C} \odot \mathbf{B})\mathbf{A}^T. \tag{1.14}$$

# Chapter 2

# Main Theorem

In this chapter we will develop our algorithm for blind SS estimation. First we build the equivalent discrete base-band signal model that we work on. In the second section we apply PARAFAC analysis to solve our problem. In section 3 we introduce briefly another method JADE, which can also be applied to solve our problem.

## 2.1 Data Model

In this thesis we consider the scenario of uplink in CDMA mobile communication. Consider $M$ users (mobile stations) communicate with base station at the same time. Each user symbol bit is spread by Walsh orthogonal sequences [7]. The base station is equipped with an antenna array composed of $K$ antennas. We assume flat fading channel and Additive White Gaussian (AWG) noise in the receiving antenna array. Besides, we also assume that all the users of the same channel are synchronized in time to the accuracy of a small fraction of one chip duration. Therefore, the discrete-time baseband-equivalent data model can be written as

$$\mathbf{Y}(n) = \sum_{m=1}^{M} \mathbf{a}_m \mathrm{s}_m(n) \mathbf{c}_m + \mathbf{V}(n). \tag{2.1}$$

$\mathbf{Y}(n) \in \mathbb{C}^{K\times L}$ is the received mixed signal at the base station. $L$ is the number of chips, i.e. the length of Walsh code in one symbol duration. $n = 1, \ldots, N$ is the symbol duration. $N$ is the total number of snapshots. $\mathbf{a}_m \in \mathbb{C}^{K\times 1}$ is the SS of $m$-th user. $\mathbf{a}_m = [1, e^{-j\phi_{1m}}, \ldots, e^{-j\phi_{(K-1)m}}]^T$, where $\phi_{km}$ is the phase offset of $m$-th signal arriving at the $(k+1)$-th antenna with respect to the first antenna (without loss of generality, we take the first antenna to be the reference antenna). $m = 1, \ldots, M$, $M$ is the number of active users. $\mathrm{s}_m(n)$ is the $n$-th signal snapshot from the $m$-th user. $\mathbf{c}_m \in \{+1, -1\}^{1\times L}$ is the Walsh spreading sequence of the $m$-th user. $\mathbf{V} \in \mathbb{C}^{K\times L}$ is the AWG noise at base station antenna array. Note this model is very general without any assumption of array geometry. It's useful to recast the data model in matrix form by

$$\begin{aligned}
\mathbf{Y}(n) &= \begin{pmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \ldots & \mathbf{a}_M \end{pmatrix} \begin{pmatrix} \mathrm{s}_1(n)\mathbf{c}_1 \\ \mathrm{s}_2(n)\mathbf{c}_2 \\ \vdots \\ \mathrm{s}_M(n)\mathbf{c}_M \end{pmatrix} + \mathbf{V}(n) \qquad (2.2)\\
&= \begin{pmatrix} \mathbf{a}_1 & \ldots & \mathbf{a}_M \end{pmatrix} \begin{pmatrix} \mathrm{s}_1(n) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathrm{s}_M(n) \end{pmatrix} \begin{pmatrix} \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_M \end{pmatrix} + \mathbf{V}(n) \\
&= \mathbf{A}\mathbf{S}(n)\mathbf{C} + \mathbf{V}(n), \qquad (2.3)
\end{aligned}$$

where $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_M]$ is the SS matrix. We assume that the elements of $\mathbf{A}$ are constant over the N-sample observation window. $\mathbf{S}(n) = \mathrm{diag}(\mathrm{s}_1, \mathrm{s}_2, \ldots, \mathrm{s}_M)$ and $\mathbf{C} = (\mathbf{c}_1^T, \mathbf{c}_2^T, \ldots, \mathbf{c}_M^T)^T$. Our task now is to estimate $\mathbf{A}$ with a blind method.

Assume signals have zero mean and signals from different users are independent. Noises are i.i.d. Gaussian noise and are uncorrelated with user signals. Covariance matrix of the

received signal $\mathbf{y}$ can be calculated as

$$\begin{aligned}\mathbf{R}_{\mathrm{y}} &= \mathrm{E}\{\mathbf{Y}(n)\mathbf{Y}(n)^H\} \\ &= \mathrm{E}\{(\mathbf{AS}(n)\mathbf{C}+\mathbf{V}(n))(\mathbf{AS}(n)\mathbf{C}+\mathbf{V}(n))^H\} \\ &= \mathbf{A}\mathrm{E}\{\mathbf{S}(n)\mathbf{C}\mathbf{C}^H\mathbf{S}(n)^H\}\mathbf{A}^H+\mathrm{E}\{\mathbf{V}\mathbf{V}^H\}. \end{aligned} \tag{2.4}$$

Walsh spreading sequences are orthogonal sequences. We further normalize $\mathbf{C}\mathbf{C}^H=\mathbf{I}$. So now equation(2.4) can be further simplified to

$$\mathbf{R}_{\mathrm{y}} = \mathbf{A}\mathbf{R}_{\mathrm{s}}\mathbf{A}^H+\sigma_{\mathbf{v}}^2\mathbf{I}, \tag{2.5}$$

where $\mathbf{R}_{\mathrm{s}} = E\{\mathbf{S}(n)\mathbf{S}(n)^H\}$. $\sigma_{\mathbf{v}}^2$ is the noise power.

## 2.2 Main Idea

Let's first ignore the contribution of noise, then equation (2.5) can be written as

$$\mathbf{R}_{\mathrm{y}} = \mathbf{A}\mathbf{R}_{\mathrm{s}}\mathbf{A}^H. \tag{2.6}$$

Note that both $\mathbf{A}$ and $\mathbf{R}_{\mathrm{s}}$ are unknown at the base station. With the only known $\mathbf{R}_{\mathrm{y}}$, it is impossible to estimate $\mathbf{A}$. Even though we assume that each signal has the same power and the signal power is already known at the base station, then

$$\mathbf{R}_{\mathrm{y}} = \sigma_{\mathbf{s}}^2\mathbf{A}\mathbf{A}^H. \tag{2.7}$$

It's well known that $\mathbf{A}$ can only be estimated up to some unitary ambiguity [20]. Suppose that there's a matrix $\widetilde{\mathbf{A}} = \mathbf{AU}$, with $\mathbf{U}$ an unitary matrix. We can find $\mathbf{R}_{\mathrm{y}} = \sigma_{\mathbf{s}}^2\widetilde{\mathbf{A}}\widetilde{\mathbf{A}}^H$. Without further structure or additional information, this unitary ambiguity can not be removed.

But the situation will change if we add one "additional" dimension to the covariance matrix. Fist we assume that users move not very fast and the channel state varies slowly,

so that during $N$ snapshots the SS of signals keeps unchanged. Next, we uniformly divide the whole $N$ snapshots into $P$ sub-slabs (sub-blocks), so that each slab contains $N_s = \lfloor \frac{N}{P} \rfloor$ snapshots ($\lfloor \mathrm{x} \rfloor$ denotes the largest integer less than x). We fix the transmit power of signals inside each slab and change the transmit power between slabs. This can be implemented easily in practise. One interesting things here is that the flat fading effects of the channel will "help" us to estimate signal spatial signatures, because it also changes the amplitude of signals. With the changing power of signals, we now form $P$ different signal covariance matrices. Let's denote $\mathbf{R}_{\mathrm{y}(p)}, p = 1, \ldots, P$ as the covariance matrix for the received mixture signals of the $p$-th slab, $\mathbf{R}_{\mathrm{s}(p)}, p = 1, \ldots, P$ as the covariance matrix for the source signals of the $p$-th slab [1]. According to (2.5), we get

$$\begin{aligned} \mathbf{R}_{\mathrm{y}(1)} &= \mathbf{A}\mathbf{R}_{\mathrm{s}(1)}\mathbf{A}^H + \sigma_{\mathbf{v}}^2\mathbf{I} \\ \mathbf{R}_{\mathrm{y}(2)} &= \mathbf{A}\mathbf{R}_{\mathrm{s}(2)}\mathbf{A}^H + \sigma_{\mathbf{v}}^2\mathbf{I} \\ &\vdots \\ \mathbf{R}_{\mathrm{y}(\mathrm{P})} &= \mathbf{A}\mathbf{R}_{\mathrm{s}(\mathrm{P})}\mathbf{A}^H + \sigma_{\mathbf{v}}^2\mathbf{I}. \end{aligned} \tag{2.8}$$

Our idea is enlightened by the promising low rank decomposition of multi dimensional array model—PARAFAC, as introduced in section 1.2. If we stack the $P$ covariance matrices from the front to the back, we will get a three way array $\underline{\mathbf{R}_{\mathrm{y}}}$ (Refer to Figure 1.3 for this operation). In PARAFAC model it states that under a certain sufficient condition, this three way array can be uniquely decomposed to $M$ rank one three way arrays. Therefore the SS matrix $\mathbf{A}$ can be blindly estimated.

[1] In multi-way analysis choices of notations and symbols play an important role to avoid confusing [14].

## 2.3 PARAFAC Model

### 2.3.1 Fit Our Model to PARAFAC

Let's denote $r_{\mathrm{y}}(i,j,k)$ as the $(i,j,k)$-th element from the three way array $\underline{\mathbf{R}_{\mathrm{y}}}$. It holds that

$$r_{\mathrm{y}}(i,j,k) = \sum_{m=1}^{M} a(i,m) r_{\mathrm{s}}(k,m) a^*(m,j), \tag{2.9}$$

where $a(i,m)$ is the $(i,m)$-th element of $\mathbf{A}$, $r_{\mathrm{s}}(k,m)$ is the $(k,m)$-th element of $\mathbf{R}_{\mathrm{s}}$ and $a^*(m,j)$ is the $(m,j)$-th element of $\mathbf{A}^H$. (Note that equation (2.9) is a special case of equation (1.9) in that $c = a^*$ and $b$ is always positive. We will discuss this kind of "special case" later in section 3.5.3). We are now interested in under which conditions, this trilinear decomposition is essentially unique. Before the introduction of the unique decomposition theorem of PARAFAC, we first introduce the definition of Kruskal rank. The concept of Kruskal rank was implicit in the seminal work of Kruskal [16] and was later coined by Harshman.

DEFINITION: The Kruskal rank (K-rank) of a matrix $\mathbf{A} \in \mathbb{C}^{I \times J}$ is $k$, means that any equal or less than $k$ column vectors from $\mathbf{A}$ are linear independent.

To make this concept clear, let's look at two matrices:

$$\begin{pmatrix} 1 & 0 & a \\ 0 & 1 & 0 \end{pmatrix}; \begin{pmatrix} 1 & 0 & a \\ 0 & 1 & b \end{pmatrix}$$

For $a, b \neq 0$ the rank for both matrices are 2, they are both full rank. However their K-rank are different. According to the definition above, K-rank for the left matrix is 1 (K-rank deficient), while K-rank for the right matrix is 2 (full K-rank).

According to the definition, it's easy to see the following properties of K-rank (We will use $k_A$ to denote K-rank($\mathbf{A}$) and $r_A$ to denote rank($\mathbf{A}$) (normal rank) in the rest of this thesis for convenience):

1. $k_A \leq r_A$.

2. if **A** is full rank,then it's also full K-rank.

PROOF:

- K-rank stress that *any equal or less than k* column vectors from **A** are linear independent, while rank only requires that *at least one k* column vectors from A are linear independent. This proves $k_A \leq r_A$, which means K-rank is "strong rank".
- If **A** is full rank, then all column vectors of **A** are linear independent. From the definition above, **A** is also full K-rank.

Now let's assume that signal to noise ratio (SNR) is high enough, so that the contribution of noise powers in (2.8) to the covariance matrices $\mathbf{R}_{\mathrm{y}(p)}$ can be neglected (The role of noise will be discussed in chapter 3). Further on we find from (2.4) that the source signal covariance matrices $\mathbf{R}_{\mathrm{s}(p)}$ in (2.8) are diagonal matrices with each element corresponding to the source signal power (The diagonal property of $\mathbf{R}_{s(p)}$ is very important as to be discussed in chapter 3). Now we introduce a new matrix $\mathbf{P} \in \mathbb{R}^{P\times M}$, whose $p$-th row is taken from the diagonal elements of $\mathbf{R}_{\mathrm{s}(p)}$. We further define $\mathbf{D}_p(\mathbf{P})$ as take the $p$-th row of $\mathbf{P}$ and make it diagonal. We have now $\mathbf{D}_p(\mathbf{P}) = \mathbf{R}_{\mathrm{s}(p)}$. Equation (2.8) now can be rewritten as

$$\begin{aligned} \mathbf{R}_{\mathrm{y}(1)} &= \mathbf{A}\mathbf{D}_1(\mathbf{P})\mathbf{A}^H \\ \mathbf{R}_{\mathrm{y}(2)} &= \mathbf{A}\mathbf{D}_2(\mathbf{P})\mathbf{A}^H \\ &\vdots \\ \mathbf{R}_{\mathrm{y}(\mathrm{P})} &= \mathbf{A}\mathbf{D}_P(\mathbf{P})\mathbf{A}^H. \end{aligned} \tag{2.10}$$

Following the notation in [14], we denote $\mathbf{R}_{\mathrm{y}}(a)$ as cutting $\underline{\mathbf{R}_{\mathrm{y}}}$ from front to end and stack the resulting slabs to form a tall matrix. We can write (2.10) in closed form

$$\mathbf{R}_{\mathrm{y}}(a) = \begin{pmatrix} \mathbf{R}_{\mathrm{y}(1)} \\ \mathbf{R}_{\mathrm{y}(2)} \\ \vdots \\ \mathbf{R}_{\mathrm{y}(\mathrm{P})} \end{pmatrix} = \begin{pmatrix} \mathbf{A}\mathbf{D}_1(\mathbf{P})\mathbf{A}^H \\ \mathbf{A}\mathbf{D}_2(\mathbf{P})\mathbf{A}^H \\ \vdots \\ \mathbf{A}\mathbf{D}_P(\mathbf{P})\mathbf{A}^H \end{pmatrix} = \begin{pmatrix} \mathbf{A}\mathbf{D}_1(\mathbf{P}) \\ \mathbf{A}\mathbf{D}_2(\mathbf{P}) \\ \vdots \\ \mathbf{A}\mathbf{D}_P(\mathbf{P}) \end{pmatrix} \mathbf{A}^H = (\mathbf{P} \odot \mathbf{A})\mathbf{A}^H. \tag{2.11}$$

### 2.3.2 Identifiability

Our concern is that under which condition, $\underline{\mathbf{R}_{\mathrm{y}}}$ can be uniquely decomposed into equation (2.11). Kruskal's pioneering paper in 1977 [16] gave the answer for real three way array. Sidiropoulos *et al.* extended it to the complex array [22].

THEOREM: Consider a set of P matrices $\mathbf{R}_{\mathrm{y}(p)} = \mathbf{A}\mathbf{D}_p(\mathbf{P})\mathbf{A}^H, p = 1, \ldots, P$, where $\mathbf{A} \in \mathbb{C}^{K\times M}, \mathbf{P} \in \mathbb{R}^{P\times M}$. If

$$k_A + k_P + k_{A^*} \geq 2M + 2, \tag{2.12}$$

then $\mathbf{A}, \mathbf{P}, \mathbf{A}^*$ are unique up to permutation and scaling of column. That is to say, if there exist another $\overline{\mathbf{A}}, \overline{\mathbf{P}}, \overline{\mathbf{A}^*}$ which satisfies equation (2.12), they are related to $\mathbf{A}, \mathbf{P}, \mathbf{A}^*$ via

$$\overline{\mathbf{A}} = \mathbf{A}\mathbf{\Pi}\mathbf{\Delta}_1, \overline{\mathbf{P}} = \mathbf{P}\mathbf{\Pi}\mathbf{\Delta}_2, \overline{\mathbf{A}^*} = \mathbf{A}^*\mathbf{\Pi}\mathbf{\Delta}_3, \tag{2.13}$$

where $\mathbf{\Pi}$ is permutation matrix, $\mathbf{\Delta}_1, \mathbf{\Delta}_2, \mathbf{\Delta}_3$ are diagonal matrices and

$$\mathbf{\Delta}_1\mathbf{\Delta}_2\mathbf{\Delta}_3 = \mathbf{I}. \tag{2.14}$$

PROOF

Prove of this theorem is complicated, readers can refer to [16],[22]. ■

Note that scaling ambiguity is irrelevant in many applications. For example, in MVDR or LCMP beamformer algorithm, the change of scaling in $\mathbf{A}$ doesn't change the SNR at the output of antenna array. However if we do want to remove these ambiguities, we can turn to e.g. some semi-blind method, which capitalizes on periodically inserted pilots [17].

It's helpful to mention that Sidiropoulos and Bro has extended this theorem to N-way analysis [20], with the conclusion that moving to higher dimension will improve identifiability. However, the calculation expense will increase with increasing dimensions.

In a straight forward way we can get the following corollaries:

*Corollary 1*: In order to satisfy identification, the number of slabs must satisfy $P \geq 2$

PROOF

$$\begin{aligned} k_A = k_A^* \leq r_A \leq min(K, M) \leq M \\ 2M + k_P \geq k_A + k_A^* + k_P \geq 2M + 2 \\ k_P \geq 2 \end{aligned} \tag{2.15}$$

■

Which means we must collect at least 2 covariance matrices to form a "covariance array". In chapter 3 we will show that 2 matrices are enough to carry out the estimation. With increasing number of matrices, the performance of estimation doesn't improve.

*Corollary 2*: In order to satisfy identification, the number of antennas at the base station must satisfy $K \geq \frac{M+2}{2}$.

PROOF

$$\begin{aligned} k_A = k_A^* \leq min(K, M) \leq K \\ k_P \leq min(P, M) \leq M \\ 2K + M \geq k_A + k_P + k_A^* \geq 2M + 2 \\ K \geq \frac{M+2}{2} \end{aligned} \tag{2.16}$$

■

Compared to some conventional subspace methods like MUSIC, given a certain number of antennas, PARAFAC model nearly doubles the number of identifiable signals. However, $\frac{M+2}{2}$ is only the theoretical lower bound for identification. With increasing number of antennas, estimation results will improve a lot. We will discuss it in detail in chapter 3. (For the knowledge of number of signals, we can use some algorithms like [27] and the references therein to give a estimation).

### 2.3.3 Advantage of Using Covariance Matrices

In [22], data matrices were used to fit PARAFAC model. In our algorithm, however covariance matrices are applied to fit PARAFAC model. Now we analyze the advantage of using covariance matrices. If data matrices are used, the data model will be the same as we used to derive the D-CRB in appendix (A.1). In equation (A.1), suppose we collect $P$ data matrices, there will be altogether $PKN_s$ elements in matrices $\mathbf{Y}_{\mathrm{a}(p)}, p = 1, \ldots, P$. Therefore the number of equations is $PKN_s$. Now let's look at the unknowns. The total number of unknowns at the right side of (A.1) equals $(K-1)M+PM+PMN_s$. According the relation of equations and unknowns, we have the following inequality for identifiability

$$PKN_s > (K-1)M + PM + PMN_s. \tag{2.17}$$

It's easy to prove that $K > M$, which means more antennas than users, is necessary for identifiability.

Let's move to the covariance matrices. The number of equations in (2.10) is $PK^2$, while the number of unknowns is $(K-1)M+PM$. Therefore, we have the following identifiability condition:

$$PK^2 > (K-1)M + PM. \tag{2.18}$$

Now it's possible to identify $\mathbf{A}$ with $K < M$ (less antennas than users). The interpretation for this is simple: If we use data matrices, we must estimate the *signal data matrix* together with $\mathbf{A}$. Because signal data are different from snapshots to snapshots, we will have more unknowns. However if we use covariance matrices, the situation will change. We only need to estimate the *signal power*, which belongs to the statistical property of the signal and will stay unchanged in each covariance matrix slab. Therefore we will have less unknowns.

Before finishing this section, we would like to point out another advantage of PARAFAC besides identifiability potentials. From (2.13) we can see that the signal powers, which is a very useful information in many scenarios, can also be obtained.

## 2.4 Trilinear Alternative Least Square

With the model (2.10) and unique decomposition theorem at hand, now we try to find some methods to finally carry out the decomposition. Trilinear Alternative Least Square (TALS) can be applied. TALS is an iterative method. In each iteration, three updates are carried out. In each update we fix two matrices and get the least square solution of the third matrix. To put it in detail, in each iteration, we first fix $\mathbf{A}$ and $\mathbf{P}$ from last iteration and only update $\mathbf{A}^*$ by solving

$$\mathbf{A}^* = \arg\min_{\mathbf{A}^*} \|\mathbf{R}_{\mathrm{y}}(a) - (\mathbf{P} \odot \mathbf{A})\mathbf{A}^H\|_F^2. \tag{2.19}$$

We get the least square solution

$$\mathbf{A}^H = (\mathbf{P} \odot \mathbf{A})^\dagger \mathbf{R}_{\mathrm{y}}(a). \tag{2.20}$$

Then we use $\mathbf{P}$ from last iteration and $\mathbf{A}^*$ from this iteration to update $\mathbf{A}$ by solving

$$\mathbf{A} = \arg\min_{\mathbf{A}} \|\mathbf{R}_{\mathrm{y}}(b) - (\mathbf{A}^* \odot \mathbf{P})\mathbf{A}^T\|_F^2. \tag{2.21}$$

We obtain the solution

$$\mathbf{A}^T = (\mathbf{A}^* \odot \mathbf{P})^\dagger \mathbf{R}_{\mathrm{y}}(b). \tag{2.22}$$

Finally we use $\mathbf{A}$ and $\mathbf{A}^*$ from this iteration and update $\mathbf{P}$ by solving

$$\mathbf{P} = \arg\min_{\mathbf{P}} \|\mathbf{R}_{\mathrm{y}}(c) - (\mathbf{A} \odot \mathbf{A}^*)\mathbf{P}^T\|_F^2. \tag{2.23}$$

We obtain

$$\mathbf{P}^T = (\mathbf{A} \odot \mathbf{A}^*)^\dagger \mathbf{R}_{\mathrm{y}}(c). \tag{2.24}$$

For each iterative algorithm, there is the initialization problem. In TALS we can simply initialize $\mathbf{A}, \mathbf{A}^*$ and $\mathbf{P}$ with random numbers. Because in each iteration the update of three matrices may either improve or maintain but can not worsen the current fit, global convergence of TALS fitting is almost guaranteed.

The computation work of TALS are mainly consumed by least square fitting. Therefore the computational complexity of TALS in one iteration is $\mathcal{O}(K^2MP)$. The number of iterations required till convergence depends on different data and initial values. There are also some fast algorithms available to carry out three way decomposition. We will discuss it in chapter 3.

## 2.5 JADE

Take a further look into equation (2.10). We find that $\mathbf{R}_{\mathrm{y}(p)}$can be diagonalized to $\mathbf{D}_p(\mathbf{P})$ by $\mathbf{A}$. In other words, $\mathbf{A}$ is the joint diagonalizer of $P$ matrices $\mathbf{R}_{\mathrm{y}(1)}, \ldots, \mathbf{R}_{\mathrm{y}(P)}$. We can turn to some methods which can jointly diagonalize a set of matrices to find $\mathbf{A}$. A similar idea can be found in [2] and the references therein. In their case, they use a joint approximate diagonalization (JADE) method to separate a mixture of sources from a set of covariance matrices at different time delay. Take the idea of JADE in the time domain to our scenario in the space domain, we develop the following algorithm (for detailed derivations, refer to [2]).

1. Calculate the EigenValue Decomposition (EVD) of $\mathbf{R}_{\mathrm{y}(1)}$. Denote $\lambda_1, \ldots, \lambda_M$ as the $M$ largest eigenvalues, $\mathbf{h}_1, \ldots, \mathbf{h}_M$ the corresponding eigen vector.

2. Estimate the noise power $\widehat{\sigma}_{\mathrm{v}}^2$ by averaging the $(K-M)$ smallest eigenvalues.

3. Form a whitening matrix in the following way

$$\widehat{\mathbf{W}} = [(\lambda_1 - \widehat{\sigma}_{\mathrm{v}}^2)^{-\frac{1}{2}}\mathbf{h}_1, \ldots, (\lambda_M - \widehat{\sigma}_{\mathrm{v}}^2)^{-\frac{1}{2}}\mathbf{h}_M]^H. \tag{2.25}$$

4. Apply $\widehat{\mathbf{W}}$ to prewhite the received mixture signal $\mathbf{Y}(n)$. Denote $\mathbf{Z}(n) = \widehat{\mathbf{W}}\mathbf{Y}(n)$. And calculate the covariance matrix of the whitened signals to obtain: $\mathbf{R}_{\mathrm{z}(1)}, \ldots, \mathbf{R}_{\mathrm{z}(P)}$.

5. Obtain an unitary matrix $\widehat{\mathbf{U}}$ as a joint diagonlizer of $\mathbf{R}_{\mathrm{z}(1)}, \ldots, \mathbf{R}_{\mathrm{z}(P)}$.

6. The SS matrix $\mathbf{A}$ can be estimated as $\widehat{\mathbf{A}} = \widehat{\mathbf{W}}^{\dagger}\widehat{\mathbf{U}}$.

# Chapter 3

# Simulation

## 3.1 Simulation Setup

In our blind estimation algorithm, we don't exploit the manifold of antenna array. The manifold of the antenna array can be arbitrary. For the reason of convenience and closer to practice, we assume a uniform linear array. In all numerical simulations we assume the antennas are omnidirectional and displaced by half wavelength. User signals are spread by Walsh code and BPSK modulation scheme is applied.

The so called Sylvester construction can be used to generate Walsh code. That is:

$$\mathbf{C}_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}; \mathbf{C}_{2n} = \mathbf{C}_2 \otimes \mathbf{C}_n.$$

Based on this construction, $\mathbf{C}_4$ for example can be obtained as

$$\mathbf{C}_4 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}.$$

As analyzed in section 2.3, the signal powers should be changed between covariance matrix slabs. In simulations the powers are changed according to uniform distribution

$$\mathbf{P} = \mathrm{SNR}(\mathrm{ones}(P, M) + f_{\mathrm{PCF}}(\mathrm{rand}(P, M) - 0.5))\sigma_{\mathrm{v}}^2, \tag{3.1}$$

where SNR (signal to noise ratio) is defined as the ratio of single signal power at one receiving antenna to the power of noise at one receiving antenna, ones$(P, M)$ means a $P \times M$ matrix with all the elements equal one, rand$(P, M)$ is a $P \times M$ matrix with each elements $(0, 1)$ uniformly distributed, $f_{\text{PCF}}$ is the so called power change factor (PCF), which measures the extent the power changes, can be adjusted. The larger the $f_{\text{PCF}}$, the larger the power changes.

As mentioned in section 2.4. TALS can be applied to fit the PARAFAC model. However it can be shown that with random initial values, TALS needs many iterations to converge. A fast algorithm called COMFAC is applied in the simulation to fit the PARAFAC model [6]. COMFAC works by first compress the original three way array of dimension $K \times K \times P$ to a $K' \times K' \times P'$ compressed three way array using a Tucker3 model [5]. Normally, after compression $K' \times K' \times P'$ is much smaller than $K \times K \times P$. Subsequently the compressed array is fitted to the PARAFAC model. This process can be initialized with PARAFAC-TALS or by some better methods such as Complex Generalized Rank Annihilation Method(CGRAM) method or Direct TriLinear Decomposition (DTLD). Then the solution is decompressed to the original space. Finally a few safeguard PARAFAC-TALS steps are performed. Detailed information about other three way analysis algorithms mentioned above such as CGRAM and DTLD can be found in the webpage of Rasmus Bro at http://newton.foodsci.kvl.dk/users/rasmus/. N-way analysis toolbox for MATLAB can be downloaded from http://www.models.kvl.dk/source/.

As pointed out in equation (2.13), the results from PARAFAC analysis carry ambiguities. In simulations, the scaling ambiguity is removed in the following way. Recall that without loss of generality, we assumed that the first antenna in the antenna array is the reference antenna. So the elements in first row of $\mathbf{A}$ equal 1. By dividing in each column vector the

corresponding first element, i.e.

$$\widehat{\mathbf{A}}(:,m) = \frac{\widehat{\mathbf{A}}(:,m)}{\widehat{\mathbf{A}}(1,m)}, \tag{3.2}$$

for $m = 1, \ldots, M$. $\widehat{\mathbf{A}}$ denotes the estimation of $\mathbf{A}$. The scaling ambiguity is removed. Regarding to the permutation ambiguity, we applied the so called greedy matching method. The idea of greedy matching is to calculate the vector distance (Frobenius norm) between each column vectors in $\mathbf{A}$ and $\widehat{\mathbf{A}}$. Then permute the columns in $\widehat{\mathbf{A}}$ according to the comparison result. This algorithm is suboptimum, which is nevertheless robust and computationally efficient to carry out for any number of users.

We adopt average mean square error (AMSE) for performance evaluation and comparison, which is

$$\text{AMSE} = \frac{\|\widehat{\mathbf{A}} - \mathbf{A}\|_F^2}{KM}. \tag{3.3}$$

To show the performance of our algorithm, the AMSE is compared with deterministic CRB (D-CRB). As shown in [8], D-CRB is lower than stochastic CRB (S-CRB) and is not achievable in the general case. Therefore, the performance of our algorithm is compared to a more strict criteria. Corresponding to (3.3), the D-CRB is also normalized by diving the number of elements in $\mathbf{A}$

$$\text{CRB} = \frac{\text{Trace(CRBaa)}}{KM}. \tag{3.4}$$

In the following sections we will discuss different numerical simulations.

## 3.2 Estimation Error against Number of Snapshots

In these simulations, there are two signals arriving from 7° and 15° from the broad side of the antenna array. SNR = 10dB. The antenna array contains 10 antennas. We collect 5 covariance matrix slabs for each simulation. Each point is the average result from 100 runs. The abscissa denotes the total number of snapshots $N$ for $P$ matrix slabs ($P = 5$ in

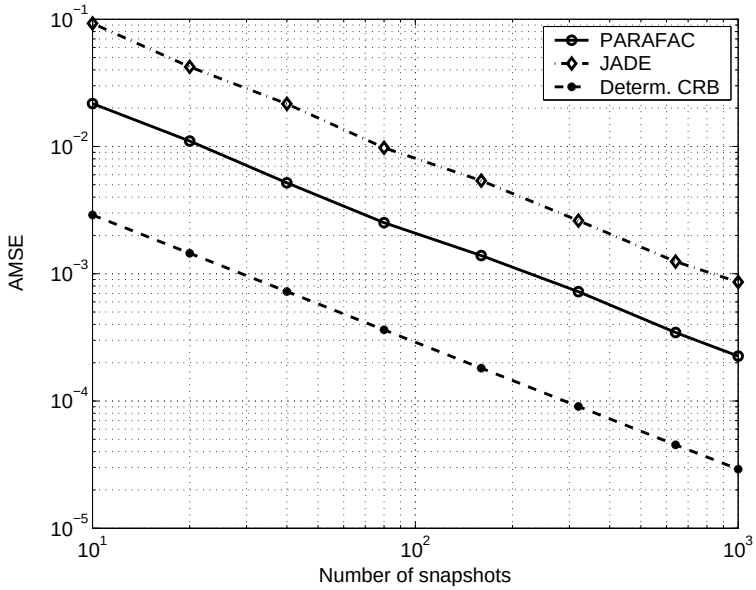


Figure 3.1: AMSE versus number of snapshots (10 antennas)

this case). The $N$ snapshots are uniformly divided into $P$ matrix slabs. Figure 3.1 shows results of the first simulation. We can see that PARAFAC algorithm stays close to D-CRB (note here that the plot is log-log scale) and performs better than JADE.

In the second simulation the simulation setups are the same as the first simulation except that the number of antennas are reduced from 10 to 5. We can see from Figure 3.2 that the estimation by PARAFAC is still better.

In the third simulation we add another two signals arriving from $-20°$ and $-40°$ respectively. There are 10 antennas in this simulation. The results are shown in Figure 3.3.

## 3.3 Estimation Error against SNR

In these simulations, we show the performance of estimation against SNR. The total number of snapshots is fixed to 500. The two impinging signals are from 7° and 15° from the broad side of the antenna array. There're 10 antenna sensors in the array. We collect 10 matrix

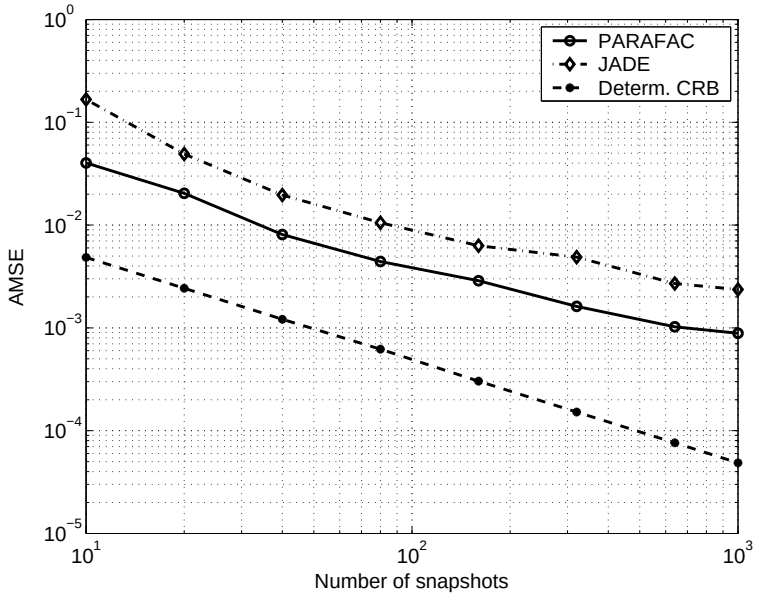


Figure 3.2: AMSE versus number of snapshots (5 antennas)

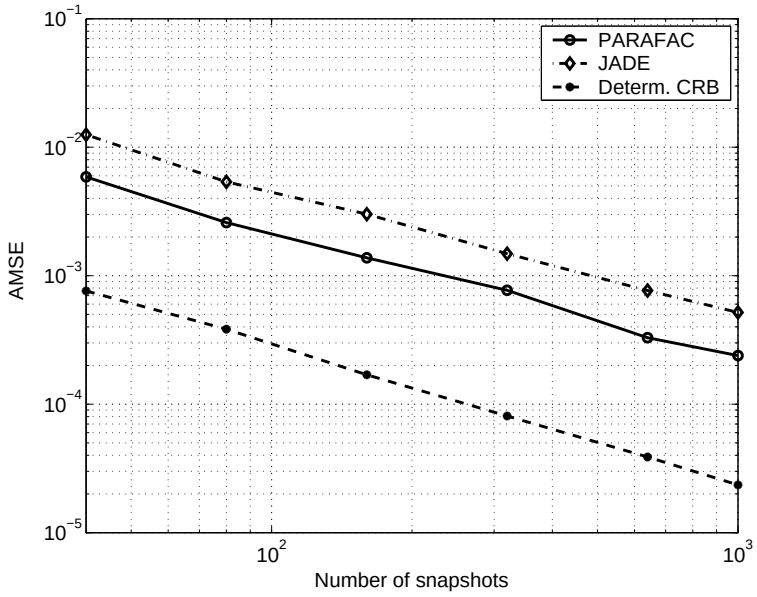


Figure 3.3: AMSE versus number of snapshots (4 users)

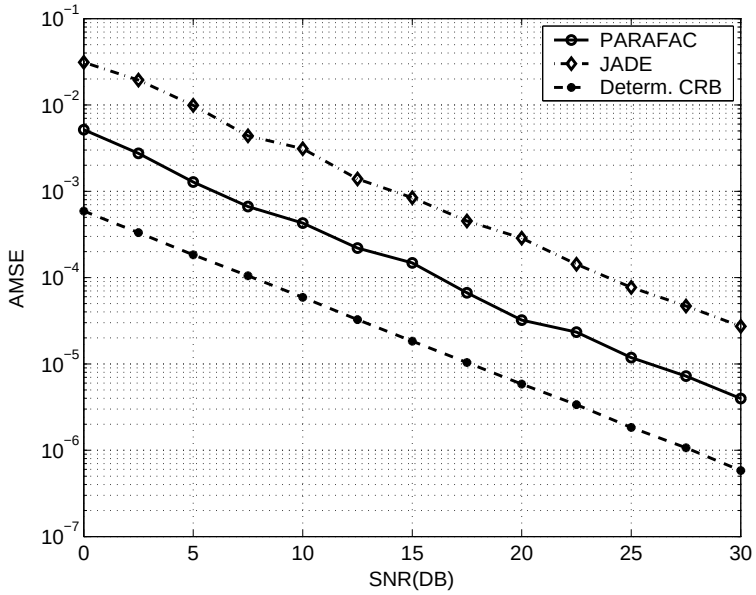


Figure 3.4: AMSE versus SNR (10 antennas)

slabs. In Figure 3.4 we can see that PARAFAC algorithm has about 8dB gain over JADE.

Figure 3.5 shows the results of the 5th simulation. The other parameters are the same as the 4th simulation except that the number of antenna is reduced from 10 to 5. We can find that PARAFAC still has 5dB gain over JADE.

## 3.4 Effects of Parameters

As analyzed in section 2.3, parameters that affects the identifiability and performance of PARAFAC are: the number of antenna elements, power change factor ($f_{\mathrm{PCF}}$), and the number of covariance matrix slabs. In simulation 6 and 7, we try to find the role of the number of antennas.

In simulation 6, we fix the number of covariance matrix slabs = 10, $f_{\mathrm{PCF}} = 12$ and two impinging signals. We vary the number of antennas from 2 to 24. From Figure 3.6 we can see that PARAFAC performs better with increasing number of antennas. But after a

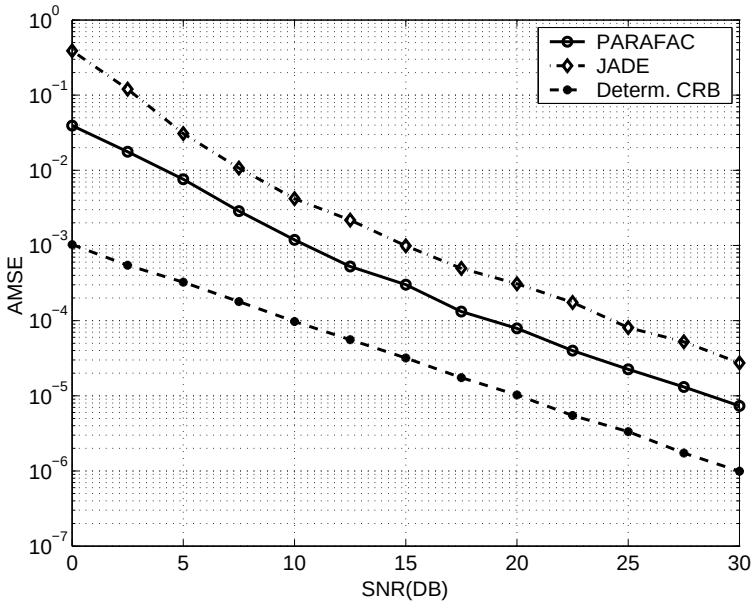


Figure 3.5: AMSE versus SNR (5 antennas)

certain threshold (10 antennas in this case) the performance improvement is not so obvious.

In the 7th simulation, there are 8 signals. The number of antennas varies from 3 to 24. According to *corollary 2* in section 2.3, 5 antennas are enough for identification. However in order to achieve better estimation, the number of antennas should be greater than 10. See Figure 3.7.

In simulation 8, we illustrate the performance against $f_{\mathrm{PCF}}$. The number of matrix slabs is 10. The number of antennas is 10. We can see from Figure 3.8 that the larger the power changes, the better the performance, which proves our analysis in section 2.3.

In the last simulation, we study the estimation performance against number of covariance matrix slabs. We fix the total number of snapshots and divide them into different number of matrix slabs. The results in Figure 3.9 show that this parameter doesn't play an important role in PARAFAC, yet it does affect JADE algorithm. Because the essence of JADE algorithm is try to diagonalize several matrices jointly to improve performance. Detailed discussion of JADE see [2].

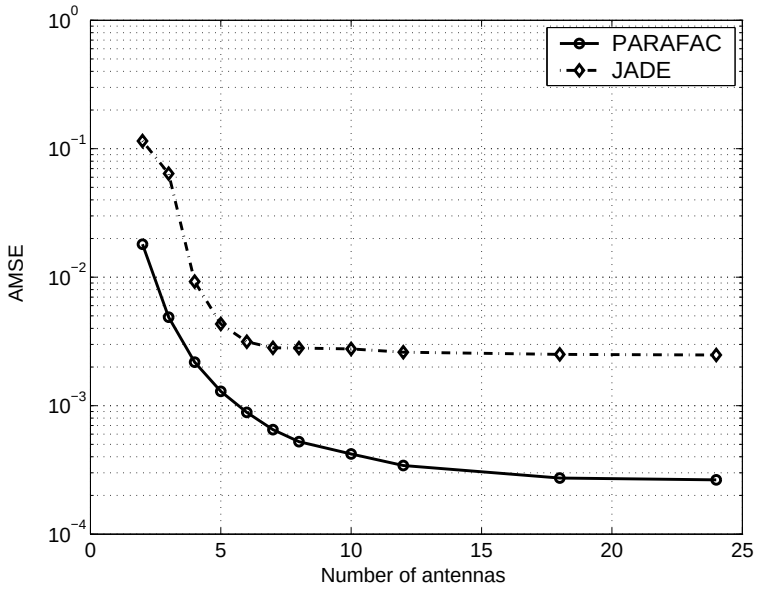


Figure 3.6: AMSE versus number of antennas (2 users)

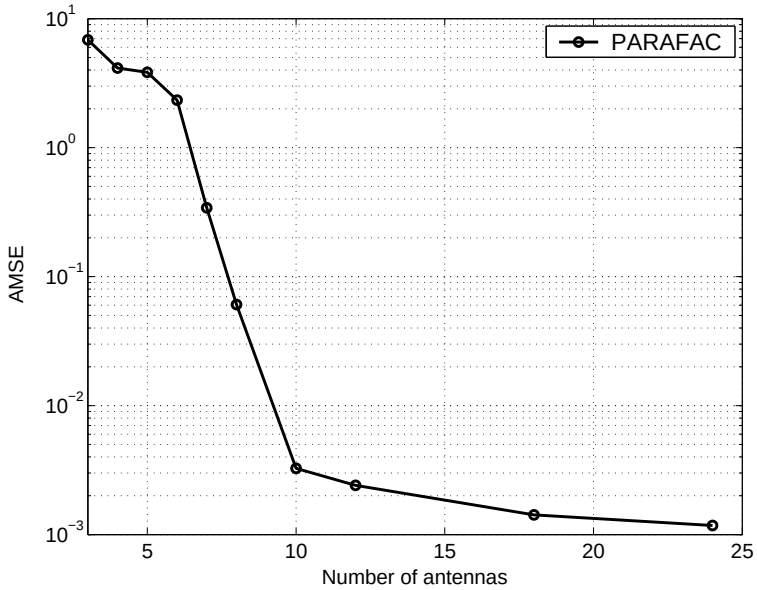


Figure 3.7: AMSE versus number of antennas (8 users)

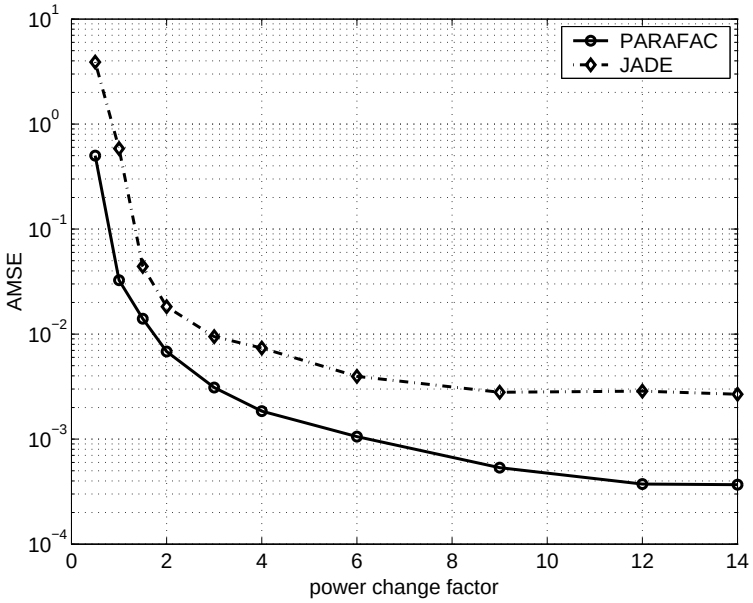


Figure 3.8: AMSE versus power change factor

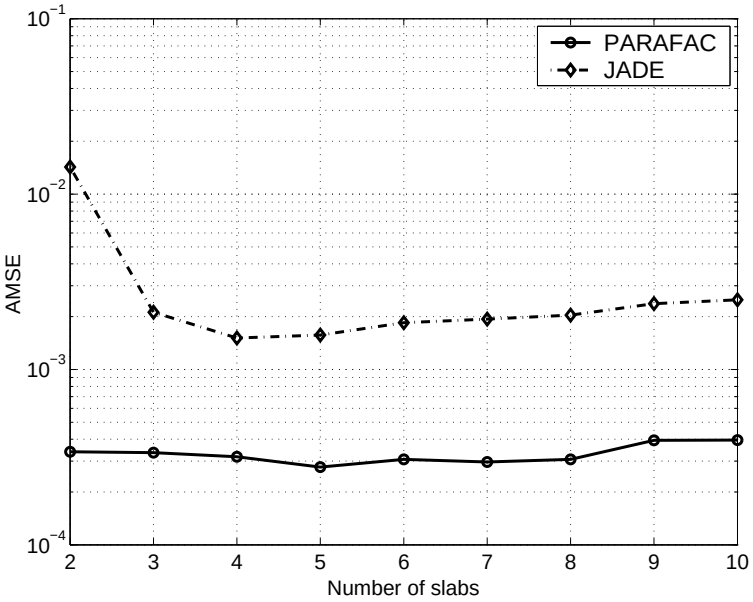


Figure 3.9: AMSE versus number of covariance matrices

## 3.5 Discussion

### 3.5.1 Diagonal Property of Covariance Matrices

As we mentioned in equation (2.8), $\mathbf{R}_{\mathrm{s}(p)}$ are diagonal matrices. This observation is very important. If $\mathbf{R}_{\mathrm{s}(p)}$ has nonzero off-diagonal elements, the performance of PARAFAC analysis will degrade dramatically. Let's rewrite equation (2.6) here for the $p$-th covariance matrix for the convenience of analysis

$$\mathbf{R}_{\mathrm{y}(p)} = \mathbf{A}\mathbf{R}_{\mathrm{s}(p)}\mathbf{A}^H. \tag{3.5}$$

Let's carry out the eigenvalue decomposition [9] of $\mathbf{R}_{\mathrm{s}(p)}$

$$\mathbf{R}_{\mathrm{s}(p)} = \mathbf{U}_{\mathrm{s}(p)}\mathbf{\Lambda}_{\mathrm{s}(p)}\mathbf{U}^H_{\mathrm{s}(p)}, \tag{3.6}$$

where $\mathbf{U}_{\mathrm{s}(p)}$ is the eigenvalue matrix of $\mathbf{R}_{\mathrm{s}(p)}$. $\mathbf{\Lambda}_{\mathrm{s}(p)}$ is the eigenvalue matrix of $\mathbf{R}_{\mathrm{s}(p)}$ and is diagonal. Insert (3.6) into (3.5), we obtain

$$\begin{aligned} \mathbf{R}_{\mathrm{y}(p)} &= \mathbf{A}\mathbf{U}_{\mathrm{s}(p)}\mathbf{\Lambda}_{\mathrm{s}(p)}\mathbf{U}^H_{\mathrm{s}(p)}\mathbf{A}^H \\ &= \widetilde{\mathbf{A}}_{(p)}\mathbf{\Lambda}_{\mathrm{s}(p)}\widetilde{\mathbf{A}}^H_{(p)}, \end{aligned} \tag{3.7}$$

where

$$\widetilde{\mathbf{A}}_{(p)} = \mathbf{A}\mathbf{U}_{\mathrm{s}(p)}.$$

Similarly, the $q$-th covariance matrix can be written as

$$\mathbf{R}_{\mathrm{y}(q)} = \widetilde{\mathbf{A}}_{(q)}\mathbf{\Lambda}_{\mathrm{s}(q)}\widetilde{\mathbf{A}}^H_{(q)}. \tag{3.8}$$

where $\mathbf{\Lambda}_{\mathrm{s}(q)}$ is diagonal. From (3.7) and (3.8) we find that the equivalent SS matrix $\widetilde{\mathbf{A}}_{(p)}$ is different for each covariance matrix slab. We can not fit them into PARAFAC model as we did in section 2.3. Therefore we can not estimate $\mathbf{A}$ properly.

As a matter of fact, we found in the simulation that PARAFAC is very sensitive to the nonzero off-diagonal elements. Even if the variances of off-diagonal elements towards zero

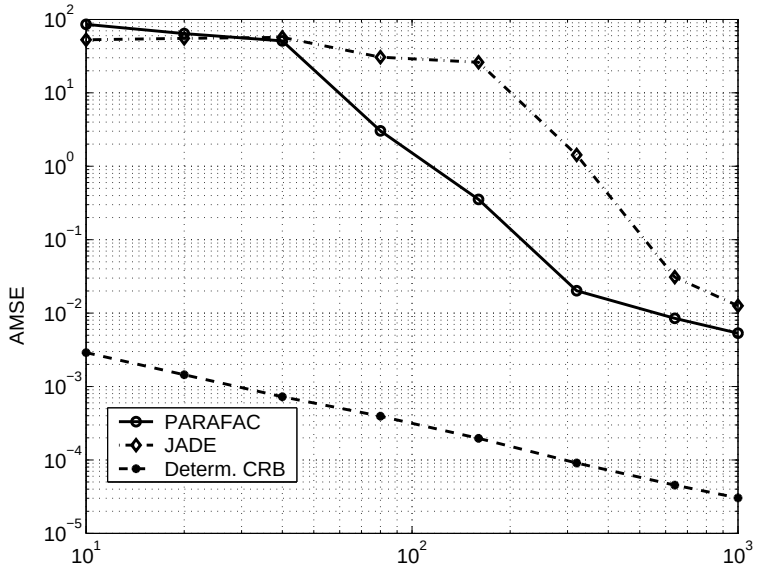


Figure 3.10: $\mathbf{R}_{\mathrm{s}(p)}$ not exactly diagonal (AMSE versus number of snapshots)

are as small as $10^{-3}$, there is still a big distance between estimation AMSE and D-CRB. Figure 3.10 illustrates AMSE versus number of snapshots when $\mathbf{R}_{\mathrm{s}(p)}$ is not exactly diagonal. Figure 3.11 shows AMSE versus SNR. Simulation setups for these two simulations are: two signals from 7° and 15° of the broad side of antenna array, 5 covariance matrices, 10 antennas in antenna array. For AMSE versus number of snapshots, SNR is fixed to 10dB. For AMSE versus SNR, total number of snapshots is fixed to 500. We can compare Figure 3.1 with Figure 3.10 and Figure 3.4 with Figure 3.11 to see the importance of diagonal property of covariance matrices in PARAFAC analysis. Notice in Figure 3.11, AMSE almost doesn't improve with increasing SNR. Our interpretation is: non-diagonal of $\mathbf{R}_{\mathrm{s}(p)}$ has more serious effect than the additive noise in PARAFAC analysis.

Thanks for the orthogonal property of Walsh code, $\mathbf{R}_{\mathrm{s}(p)}$ is exactly diagonal. However, there are still some problems left for study. If the synchronization is only roughly achieved, i.e. there is shift between the Walsh codes of different users. This shift will make $\mathbf{R}_{\mathrm{s}(p)}$ no more diagonal. Or in scenarios where the channel is frequency selective, the superposition

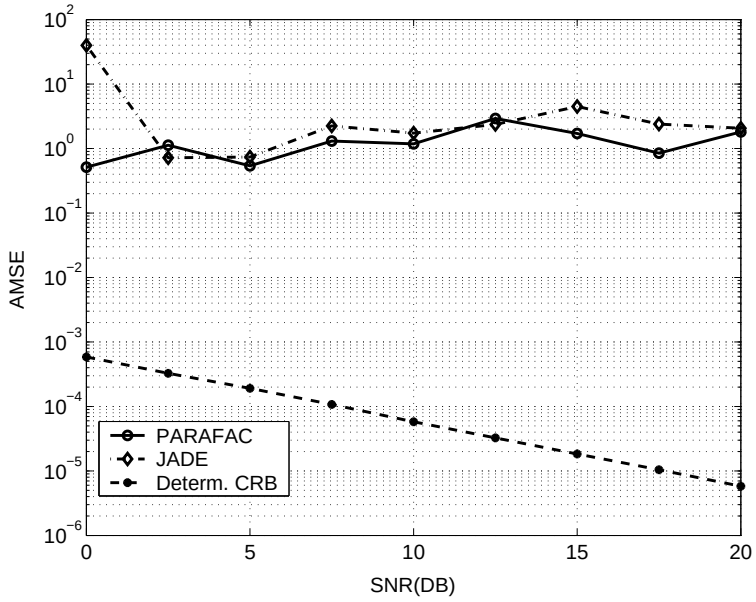


Figure 3.11: $\mathbf{R}_{\mathrm{s}(p)}$ not exactly diagonal (AMSE versus SNR)

of Walsh codes of different delay versions will again lead to the non-diagonal property of $\mathbf{R}_{\mathrm{s}(p)}$. Fortunately, in these cases, anther three way analysis model called PARAFAC2 will be of help. PARAFAC2 model can be written as

$$\mathbf{R}_{\mathrm{y}(p)} = \mathbf{A}\mathbf{D}_p\mathbf{H}\mathbf{D}_p^H\mathbf{A}^H, \tag{3.9}$$

where $\mathbf{D}_p$ is diagonal matrix and varies from different covariance matrices. Matrix $\mathbf{H}$ here need not to be diagonal. It is said that as long as $\mathbf{H}$ and $\mathbf{A}$ remain the same for each covariance matrix, $\mathbf{A}, \mathbf{H}$ can be uniquely decomposed given a set of matrices. Further discussions of PARAFAC2 are beyond the range of this thesis. We can turn to [15] and the reference therein for reference.

### 3.5.2 Elimination of Noise

We mentioned in section 2.3 that PARAFAC analysis doesn't model the noise terms. However in real applications, we always have noise at the antenna array. Some methods can be

used to estimate the noise power. Once noise power is estimated, we can subtract it from the covariance matrices before carrying out the PARAFAC analysis.

If the number of antennas is greater than the number of incoming signals, subspace method can be used to estimate the noise power. Let's carry out the eigenvalue decomposition of covariance matrices.

$$\begin{aligned} \mathbf{R}_{\mathrm{y}(p)} &= \mathbf{A}\mathbf{R}_{\mathrm{s}(p)}\mathbf{A}^H + \sigma_{\mathrm{v}}^2\mathbf{I} \\ &= \mathbf{U}_{\mathrm{y}(p)}\mathbf{\Lambda}_{\mathrm{y}(p)}\mathbf{U}_{\mathrm{y}(p)}^H, \end{aligned} \tag{3.10}$$

where

$$\mathbf{\Lambda}_{\mathrm{y}(p)} = \begin{pmatrix} \sigma_{s1}^2 + \sigma_{\mathrm{v}1}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{sM}^2 + \sigma_{\mathrm{v}M}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\mathrm{v}M+1}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{\mathrm{v}K}^2 \end{pmatrix} \tag{3.11}$$

is the eigenvalue matrix. $\sigma_{sm}^2, m = 1, \ldots, M$ are signal powers, $\sigma_{\mathrm{v}k}^2, k = 1, \ldots, K$ are noise powers. Now for each covariance matrix slab, we carry out the following calculation to eliminate the noise:

1. Calculate the eigenvalue decomposition. Denote $\lambda_1, \ldots, \lambda_K$ as the eigenvalues.

2. Average the $K - M$ smallest eigenvalues to get the estimation of noise power

$$\widehat{\sigma}_{\mathrm{v}}^2 = \Sigma_{i=M+1}^{K} \lambda_i.$$

3. Subtract $\widehat{\sigma}_{\mathrm{v}}^2$ from covariance matrices

$$\widetilde{\mathbf{R}}_{\mathrm{y}(p)} = \mathbf{R}_{\mathrm{y}(p)} - \widehat{\sigma}_{\mathrm{v}}^2\mathbf{I}.$$

Subsequently we apply PARAFAC to $\widetilde{\mathbf{R}}_{\mathrm{y}(p)}, p = 1, \ldots, P$. From Figure 3.12 we can see that there is some small improvement where SNR is low. In higher SNR there's almost no improvement. The simulation setups are the same as simulation 4.

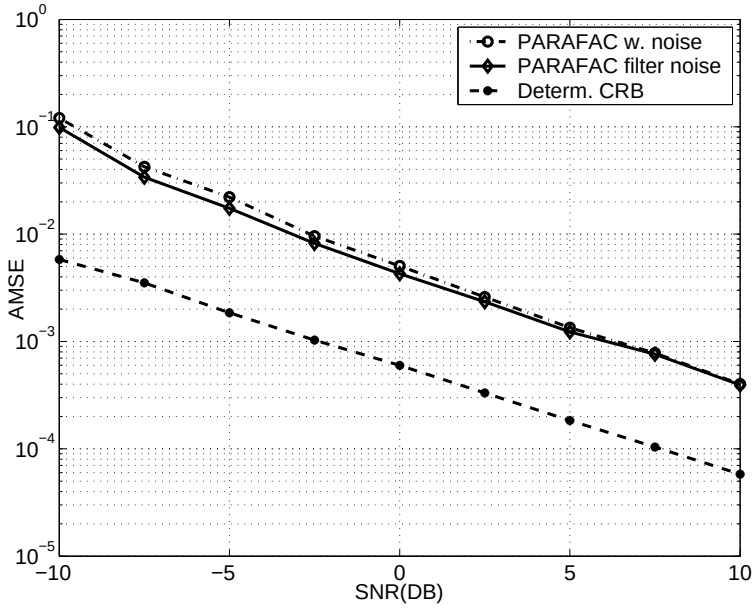


Figure 3.12: The effects of noise in PARAFAC analysis

### 3.5.3 Symmetry of PARAFAC Model

Before we finish the discussion section, we would like to point out that our model is actually a constrained version of normal PARAFAC model, in that $\mathbf{C} = \mathbf{A}^*$ in the model presented in section 1.2. That is to say, our model is "symmetric". We can exploit this additional constraint. For example, we can average the result of $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{A}}^*$ to get better estimation results. We can also reduce the computational complexity in PARAFAC model fitting by simply letting $\mathbf{A}^T = (\mathbf{A}^H)^*$ in equation (2.22). The following table shows the CPU time(in seconds) consumed by common PARAFAC fitting versus symmetric constrained PARAFAC fitting. The computer we used for simulation is a PC with Pentium4 2GHz CPU. The simulation setup is the same as simulation 4. We calculate the CPU time for different SNR. The data in table 3.1 shows the CPU time that COMFAC needs till convergence. The data is the average results of 50 runs. The data in table 3.2 shows the CPU time that TALS needs till convergence. The data is the average results of 50 runs. From these two figures,

| SNR(DB) | 0 | 5 | 10 | 15 | 20 | 25 | 30 |
|---|---|---|---|---|---|---|---|
| Common | 0.174 | 0.167 | 0.168 | 0.166 | 0.167 | 0.167 | 0.172 |
| Symmetric | 0.141 | 0.150 | 0.144 | 0.143 | 0.142 | 0.142 | 0.142 |

Table 3.1: Comparison of CPU time between common COMFAC and symmetric constrained COMFAC

| SNR(DB) | 0 | 5 | 10 | 15 | 20 | 25 | 30 |
|---|---|---|---|---|---|---|---|
| Common | 0.562 | 0.574 | 0.637 | 0.615 | 0.638 | 0.691 | 1.533 |
| Symmetric | 0.508 | 0.552 | 0.500 | 0.512 | 0.568 | 0.574 | 0.781 |

Table 3.2: Comparison of CPU time between common TALS and symmetric constrained TALS

we can also see that COMFAC needs shorter time to converge than TALS.

# Conclusion

This thesis has addressed the problem of signal spatial signature (SS) estimation, which is very important in mobile communications. Based on second order statistics (covariance matrix) of signals, we proposed a blind algorithm for SS estimation. Because of the unitary ambiguity, we can not estimate SS correctly using only *one* covariance matrix. However if we fix the signal power in a certain number of snapshots (windows) and change them between different windows, we can obtain *several* different covariance matrices. Therefore, we can form a three way covariance "array". Based on the PARAFAC analysis, we can now estimate SS correctly from this three way array only up to some scaling and permutation ambiguities. The identifiability potential of our algorithm is derived. In this thesis, our algorithm is also compared to JADE—another class of algorithm which can be used to solve our problem. Numerical simulations demonstrate that our algorithm clearly outperforms JADE and stays close to what is theoretically best performance-wise.

# Appendix A

# Derivation of Deterministic CRB

In this appendix, we derive the deterministic CRB for the estimation of SS matrix $\mathbf{A}$ by following [18]. Let's first make some simplifications:

1. assume that Walsh spreading sequences are known at the receiver, which means $\mathbf{c}_i (i = 1, \ldots, M)$ in (2.2) is not a parameter to be estimated. To derive CRB, we just take the unspreaded user symbols.

2. Although user symbols are different between matrix slabs, they do share the same statistical properties. Therefore we denote them as the same in the following derivation.

Now we can write the data model with simplification

$$\mathbf{y}_{(p)}(n) = \mathbf{A}\mathbf{D}_p(\mathbf{M})\mathbf{s}(n) + \mathbf{v}(n), \tag{A.1}$$

where $\mathbf{y}_{(p)}$ is the received signal vector in $p$th matrix slab. $n = 1, \ldots, N_s$, $\mathbf{M}$ is the amplitude matrix of signals (i.e. $\mathbf{M} = \sqrt{\mathbf{P}}$), $\mathbf{s}(n) = [s_1(n), s_2(n), \ldots, s_M(n)]^T$. We stack $N_s$ signal vectors and cast them into matrix form to obtain

$$\mathbf{Y}_{\mathrm{a}(p)} = \mathbf{A}\mathbf{D}_p(\mathbf{M})\mathbf{S}^T + \mathbf{N}_1, \tag{A.2}$$

where $\mathbf{Y}_{\mathrm{a}(p)} = [\mathbf{y}_{(p)}(1), \ldots, \mathbf{y}_{(p)}(N_s)]$, $\mathbf{S} = [\mathbf{s}^T(1), \ldots, \mathbf{s}^T(N_s)]^T$, $\mathbf{N}_1 = [\mathbf{v}^T(1), \ldots, \mathbf{v}^T(N_s)]^T$. By stacking matrices $\mathbf{Y}_{\mathrm{a}(1)}, \ldots, \mathbf{Y}_{\mathrm{a}(P)}$ from front to end together, we get a three-way array $\underline{\mathbf{Y}}$. We now unfold $\underline{\mathbf{Y}}$ into a tall matrix $\mathbf{Y}_{\mathrm{a}}$

$$\mathbf{Y}_{\mathrm{a}} = \begin{pmatrix} \mathbf{Y}_{\mathrm{a}(1)} \\ \mathbf{Y}_{\mathrm{a}(2)} \\ \vdots \\ \mathbf{Y}_{\mathrm{a}(\mathrm{P})} \end{pmatrix} = \begin{pmatrix} \mathbf{A}\mathbf{D}_1(\mathbf{M})\mathbf{S}^T + \mathbf{N}_1 \\ \mathbf{A}\mathbf{D}_2(\mathbf{M})\mathbf{S}^T + \mathbf{N}_1 \\ \vdots \\ \mathbf{A}\mathbf{D}_P(\mathbf{M})\mathbf{S}^T + \mathbf{N}_1 \end{pmatrix} = \begin{pmatrix} \mathbf{A}\mathbf{D}_1(\mathbf{M}) \\ \mathbf{A}\mathbf{D}_2(\mathbf{M}) \\ \vdots \\ \mathbf{A}\mathbf{D}_P(\mathbf{M}) \end{pmatrix} \mathbf{S}^T + \mathbf{N}_1 = (\mathbf{M} \odot \mathbf{A})\mathbf{S}^T + \mathbf{N}_1. \tag{A.3}$$

By unfolding $\underline{\mathbf{Y}}$ in other two directions and after the same manipulation, we obtain

$$\begin{aligned} \mathbf{Y}_{\mathrm{b}} &= (\mathbf{A} \odot \mathbf{S})\mathbf{M}^T + \mathbf{N}_2 \\ \mathbf{Y}_{\mathrm{c}} &= (\mathbf{S} \odot \mathbf{M})\mathbf{A}^T + \mathbf{N}_3. \end{aligned} \tag{A.4}$$

$\mathbf{N}_2, \mathbf{N}_3$ here denote noise matrix contained in $\mathbf{Y}_{\mathrm{b}}$ and $\mathbf{Y}_{\mathrm{c}}$ respectively. Under AWG noise assumption, the likelihood function of $\underline{\mathbf{Y}}$ can be written as

$$\begin{aligned} Ł(\underline{\mathbf{Y}}) &= \frac{1}{(\pi\sigma_{\mathrm{v}}^2)^{MNsP}} \exp\{-\frac{1}{\sigma_{\mathrm{v}}^2} \sum_{n=1}^{N_s} \|\mathbf{Y}_{\mathrm{a}}(n) - (\mathbf{M} \odot \mathbf{A})\mathbf{s}(n)\|^2\} \\ Ł(\underline{\mathbf{Y}}) &= \frac{1}{(\pi\sigma_{\mathrm{v}}^2)^{MNsP}} \exp\{-\frac{1}{\sigma_{\mathrm{v}}^2} \sum_{p=1}^{P} \|\mathbf{Y}_{\mathrm{b}}(p) - (\mathbf{A} \odot \mathbf{S})\mathbf{m}(p)\|^2\} \\ Ł(\underline{\mathbf{Y}}) &= \frac{1}{(\pi\sigma_{\mathrm{v}}^2)^{MNsP}} \exp\{-\frac{1}{\sigma_{\mathrm{v}}^2} \sum_{k=1}^{K} \|\mathbf{Y}_{\mathrm{c}}(k) - (\mathbf{S} \odot \mathbf{M})\mathbf{a}(k)\|^2\}. \end{aligned} \tag{A.5}$$

The unknown parameters which are going to be estimated are

$$\alpha = [\mathbf{a}_1^T, \ldots, \mathbf{a}_K^T, \mathbf{m}_1^T, \ldots, \mathbf{m}_P^T, \mathbf{s}_1^T, \ldots, \mathbf{s}_{N_s}^T, \mathbf{a}_1^H, \ldots, \mathbf{a}_K^H, \mathbf{s}_1^H, \ldots, \mathbf{s}_N^H, \sigma_{\mathrm{v}}^2]. \tag{A.6}$$

The log-likelihood function $LL(\alpha) = \ln(L(\underline{\mathbf{Y}}))$ can be written

$$\begin{aligned} LL(\alpha) &= -MNsP\ln(\pi\sigma_{\mathrm{v}}^2) - \frac{1}{\sigma_{\mathrm{v}}^2} \sum_{n=1}^{N_s} \|\mathbf{Y}_{\mathrm{a}}(n) - (\mathbf{M} \odot \mathbf{A})\mathbf{s}(n)\|^2 \\ LL(\alpha) &= -MNsP\ln(\pi\sigma_{\mathrm{v}}^2) - \frac{1}{\sigma_{\mathrm{v}}^2} \sum_{p=1}^{P} \|\mathbf{Y}_{\mathrm{b}}(p) - (\mathbf{A} \odot \mathbf{S})\mathbf{m}(p)\|^2 \\ LL(\alpha) &= -MNsP\ln(\pi\sigma_{\mathrm{v}}^2) - \frac{1}{\sigma_{\mathrm{v}}^2} \sum_{k=1}^{K} \|\mathbf{Y}_{\mathrm{c}}(k) - (\mathbf{S} \odot \mathbf{M})\mathbf{a}(k)\|^2. \end{aligned} \tag{A.7}$$

Taking partial derivation of $LL(\alpha)$ with respect to the unknown parameters. We get

$$\frac{\partial LL(\alpha)}{\partial s_{n,m}} = \frac{1}{\sigma_{\mathrm{v}}^2}(\mathbf{Y}_{\mathrm{a}}(n) - (\mathbf{M} \odot \mathbf{A})\mathbf{s}(n))^H(\mathbf{M} \odot \mathbf{A})\mathbf{e}_m \quad \text{(A.8)}$$

$$\frac{\partial LL(\alpha)}{\partial m_{p,m}} = \frac{1}{\sigma_{\mathrm{v}}^2}(\mathbf{Y}_{\mathrm{b}}(p) - (\mathbf{A} \odot \mathbf{S})\mathbf{m}(p))^H(\mathbf{A} \odot \mathbf{S})\mathbf{e}_m \quad \text{(A.9)}$$

$$\frac{\partial LL(\alpha)}{\partial a_{k,m}} = \frac{1}{\sigma_{\mathrm{v}}^2}(\mathbf{Y}_{\mathrm{c}}(k) - (\mathbf{S} \odot \mathbf{M})\mathbf{a}(k))^H(\mathbf{S} \odot \mathbf{M})\mathbf{e}_m. \quad \text{(A.10)}$$

Here $\mathbf{e}_m$ is a column vector with the $m$-th element equals one and other elements zero. Following the development in [13],[18], we derive the Fisher Information Matrix (FIM) to calculate CRB. FIM can be formed as

$$\mathbf{F} = \begin{pmatrix} \mathbf{\Psi} & 0 & 0 \\ 0 & \mathbf{\Psi}^* & 0 \\ 0 & 0 & f_{\sigma\sigma} \end{pmatrix}, \quad \text{(A.11)}$$

where

$$\mathbf{\Psi} = \begin{pmatrix} \mathbf{\Psi}_{aa} & \mathbf{\Psi}_{am} & \mathbf{\Psi}_{as} \\ \mathbf{\Psi}_{am}^H & \mathbf{\Psi}_{mm} & \mathbf{\Psi}_{ms} \\ \mathbf{\Psi}_{as}^H & \mathbf{\Psi}_{ms}^H & \mathbf{\Psi}_{ss} \end{pmatrix} \quad \text{(A.12)}$$

and $f_{\sigma\sigma}$ corresponds to the noise term in FIM. Under AWG noise, the elements in (A.12) can be derived now

$$E\{\frac{\partial LL(\alpha)}{\partial a_{k1,m1}^*}\frac{\partial LL(\alpha)}{\partial a_{k2,m2}}\} = \frac{1}{\sigma_{\mathrm{v}}^2}\mathbf{e}_{m1}^H(\mathbf{S} \odot \mathbf{M})^H(\mathbf{S} \odot \mathbf{M})\mathbf{e}_{m2}\delta_{k1k2}$$

$$E\{\frac{\partial LL(\alpha)}{\partial d_{p1,m1}^*}\frac{\partial LL(\alpha)}{\partial d_{p2,m2}}\} = \frac{1}{\sigma_{\mathrm{v}}^2}\mathbf{e}_{m1}^H(\mathbf{A} \odot \mathbf{S})^H(\mathbf{A} \odot \mathbf{S})\mathbf{e}_{m2}\delta_{p1p2}$$

$$E\{\frac{\partial LL(\alpha)}{\partial s_{n1,m1}^*}\frac{\partial LL(\alpha)}{\partial s_{n2,m2}}\} = \frac{1}{\sigma_{\mathrm{v}}^2}\mathbf{e}_{m1}^H(\mathbf{M} \odot \mathbf{A})^H(\mathbf{M} \odot \mathbf{A})\mathbf{e}_{m2}\delta_{n1n2}$$

$$E\{\frac{\partial LL(\alpha)}{\partial a_{k,m1}^*}\frac{\partial LL(\alpha)}{\partial d_{p,m2}}\} = \frac{1}{\sigma_{\mathrm{v}}^4}\mathbf{e}_{m1}^H(\mathbf{S} \odot \mathbf{M})^H E\{\mathbf{N}_3(k)\mathbf{N}_2(p)^H\}(\mathbf{A} \odot \mathbf{S})\mathbf{e}_{m2}$$

$$E\{\frac{\partial LL(\alpha)}{\partial a_{k,m1}^*}\frac{\partial LL(\alpha)}{\partial s_{n,m2}}\} = \frac{1}{\sigma_{\mathrm{v}}^4}\mathbf{e}_{m1}^H(\mathbf{S} \odot \mathbf{M})^H E\{\mathbf{N}_3(k)\mathbf{N}_1(n)^H\}(\mathbf{M} \odot \mathbf{A})\mathbf{e}_{m2}$$

$$E\{\frac{\partial LL(\alpha)}{\partial d_{p,m1}^*}\frac{\partial LL(\alpha)}{\partial s_{n,m2}}\} = \frac{1}{\sigma_{\mathrm{v}}^4}\mathbf{e}_{m1}^H(\mathbf{A} \odot \mathbf{S})^H E\{\mathbf{N}_2(p)\mathbf{N}_1(n)^H\}(\mathbf{M} \odot \mathbf{A})\mathbf{e}_{m2},$$

where $\delta$ denotes the Kronecker delta. The values of $E\{\mathbf{N}_3(k)\mathbf{N}_2(p)^H\}, E\{\mathbf{N}_3(k)\mathbf{N}_1(n)^H\}$ and $E\{\mathbf{N}_2(p)\mathbf{N}_1(n)^H\}$ depend on the correlation of corresponding noise term. According to [13]

$$\mathbf{\Psi}^{-1} = \begin{pmatrix} \mathrm{CRB}_{aa} & \mathrm{CRB}_{am} & \mathrm{CRB}_{as} \\ \mathrm{CRB}_{am}^H & \mathrm{CRB}_{mm} & \mathrm{CRB}_{ms} \\ \mathrm{CRB}_{as}^H & \mathrm{CRB}_{ms}^H & \mathrm{CRB}_{ss} \end{pmatrix}. \tag{A.13}$$

Let's denote

$$\mathbf{\Psi}_1 = \begin{pmatrix} \mathbf{\Psi}_{am} & \mathbf{\Psi}_{as} \end{pmatrix} \tag{A.14}$$

and

$$\mathbf{\Psi}_2 = \begin{pmatrix} \mathbf{\Psi}_{mm} & \mathbf{\Psi}_{ms} \\ \mathbf{\Psi}_{ms}^H & \mathbf{\Psi}_{ss} \end{pmatrix}. \tag{A.15}$$

Apply block matrix inversion rule, we finally obtain

$$\mathrm{CRB}_{aa} = (\mathbf{\Psi}_{aa} - \mathbf{\Psi}_1 \mathbf{\Psi}_2^{-1} \mathbf{\Psi}_1^H)^{-1}. \tag{A.16}$$

# Bibliography


[1] D. Astèly, A. L. Swindlehurst, and B. Ottersten, *Spatial signature estimation for uniform linear arrays with unknown receiver gains and phases*, IEEE Trans. on Signal Processing **47** (1999), no. 8, 2128–2138.

[2] A. Belouchrani, K. Abed-Meraim, J-F. Cardoso, and E. Moulines, *A blind source separation technique using second-order statistics*, IEEE Trans. on Signal Processing **45** (1997), no. 2, 434–444.

[3] J. W. Brewer, *Kronecker products and matrix caculus in system theorey*, IEEE Trans. on Circuits and Systems **cas-25** (1978), no. 9, 772–781.

[4] R. Bro, *PARAFAC: tutorial and applications*, Chemometrics and intelligent laboratory systems **38** (1997), 149–171.

[5] ———, *Multi-way analysis in the food industry models, algorithms, and applications*, Ph.D. thesis, Royal veterinary and agricultural university Denmark, 1998.

[6] R. Bro, N. D. Sidiropoulos, and G. B. Giannakis, *A fast least square algorithm for separating trilinear mixtures*, Int. Workshop Independent Component Analysis and Blind Signal Separation (Aussois, France), Jan 1999.

[7] E. H. Dinan and B. Jabbari, *Spreading codes for direct sequence CDMA and wideband CDMA cellular networks*, IEEE Communication Magzine (1998), 48–54.

[8] A. B. Gershman, P. Stoica, M. Pesavento, and E. G. Larsson, *Stochastic Cramér-Rao bound for direction estimation in unknown noise fields*, IEE Proc. radar sonar Navigation **149** (2002), no. 1, 2–8.

[9] G. H. Golub and C. F. Van Loan, *Matrix computations*, 2 ed., The Johns Hopkins university press, 1993.

[10] R. A. Harshman, *Foundation of the PARAFAC procedure: model and conditions for an "exlanatory" multi-mode factor analysis*, UCLA working papers in phonetics **16** (1970), 1–84.

[11] A. Jakobsson and A. L. Swindlehurst, *A time domain method for joint estimation of time delays, doppler shifts and spatial signatures*, 9th IEEE workshop on statistical signal and array processing, 1998, pp. 388–391.

[12] D. H. Johnson and D. E. Dudgeon, *Array signal processing: concepts and techniques*, Prentice Hall, 1993.

[13] S. M. Kay, *Fundamentals of statistical signal processing–estimation theory*, vol. 2, Prentice Hall, Upper Saddle River, NJ, 1993.

[14] H. A. L. Kiers, *Towards a standardized notation and terminology in multiway analysis*, J.chemometrics **14** (2000), 105–122.

[15] H. A. L. Kiers, J. M. F. ten Berge, and R. Bro, *PARAFAC2 - Part I: A direct fitting algorithm for the parafac2 model*, J. Chemometrics **13** (1999), 275–294.

[16] J. B. Kruskal, *Three-way arrays: rank and uniqueness of trilliner decompositions, with application to arithmetic complexity and statistics*, Linear Algebra Application **16** (1977), 95–138.

[17] H. L. Li, X. Lu, and G. B. Giannakis, *Capon multiuser receiver for CDMA systems with space-time coding*, IEEE Trans. on Signal Processing **50** (2002), no. 5, 1193–1204.

[18] X. Liu and N. D. Sidiropoulos, *Cramér-rao lower bounds for low-rank decomposition of multidimensional arrays*, IEEE Trans. on Signal Processing **49** (2001), no. 9, 2074–2086.

[19] J. G. Proakis, *Digital communications*, 4 ed., McGraw-Hill Companies Inc, 2001.

[20] N. D. Sidiropoulos and R. Bro, *On the uniqueness of multilinear decomposition of N-way arrays*, J. chemometrics **14** (2000), 229–239.

[21] N. D. Sidiropoulos, R. Bro, and G. B. Giannakis, *Parallel factor analysis in sensor array processing*, IEEE Trans. on Signal Processing **48** (2000), no. 8, 2377–2388.

[22] N. D. Sidiropoulos, G. B. Giannakis, and R. Bro, *Blind PARAFAC receivers for DS-CDMA systems*, IEEE Trans. on Signal Processing **48** (2000), no. 3, 810–823.

[23] A. L. Swindlehurst, *Time delay and spatial signature estimation using known asynchronous signals*, IEEE Trans. on Signal Processing **46** (1998), no. 2, 449–462.

[24] H. Van Trees, *Optimum array processing Part IV of detection, estimation and modulation theory*, John Wiley & Sons., New York, 2002.

[25] A. J. Viterbi, *CDMA: principles of spread spectrum communication*, Addison-Wesley, 1997.

[26] M. Wax and Y. Anu, *Performance analysis of the minimum variance beamformer in the presence of steering vector errors*, IEEE Trans. on Signal Processing **44** (1996), no. 4, 938–947.

[27] M. Wax and T. Kailath, *Detection of signals by information theoretic criteria*, IEEE Trans. on Acoustics, Speech, and Signal Processing **ASSP-33** (1985), no. 2, 387–392.

[28] G. Xu and H. Liu, *An effective transmission beamforming scheme for frequency-division-duplex digital wireless communication systems*, Proc. ICASSP (Detroit, MI), 1995, pp. 1729–1732.

[29] J. Yang and A. L. Swindlehurst, *The effects of array calibration errors on DF-based signal copy performance*, IEEE Trans. on Signal Processing **43** (1995), no. 11, 2724–2730.