



**HAL**  
open science

# Low rank decomposition, completion problems and applications: low rank decomposition of Hankel matrices and tensors

Jouhayna Harmouch

► **To cite this version:**

Jouhayna Harmouch. Low rank decomposition, completion problems and applications: low rank decomposition of Hankel matrices and tensors. Algebraic Geometry [math.AG]. COMUE Université Côte d'Azur (2015 - 2019); Université Libanaise, 2018. English. NNT : 2018AZUR4236 . hal-02073968v2

**HAL Id: hal-02073968**

**<https://theses.hal.science/hal-02073968v2>**

Submitted on 16 Apr 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



# THÈSE DE DOCTORAT

## Décomposition de petit rang, problèmes de complétion et Applications

Décomposition des matrices de Hankel et des tenseurs de rang  
faible

**Jouhayna Harmouch**  
INRIA-Sophia Antipolis Méditerranée

Présentée en vue de l'obtention  
du grade de docteur en Mathématiques  
d'Université Côte d'Azur  
et de l'Université Libanaise  
Dirigée par : Bernard Mourrain  
Co-encadrée par : Houssam Khalil  
Soutenue le:19/12/2018

Devant le jury, composé de :  
Annie Cuyt, Université d'Antwerp  
M. Thomas Sauer, Université de Passau  
Pierre Comon, CNRS  
Bernhard Beckermann, Université de Lille  
André Galligo, Université de Nice  
Bernard Mourrain, INRIA  
Mustapha Jazar, Université Libanaise  
Houssam Khalil, Université Libanaise

# Décomposition de petit rang, problèmes de complétion et Applications

## Décomposition des matrices de Hankel et des tenseurs de rang faible

### Jury :

**Président du jury** M. Bernhard Beckermann, Professeur des universités, Laboratoire de Mathématiques Paul Painlevé CNRS, Université de Lille

### Rapporteurs

1. Mme Annie Cuyt, Full Professor, Department of Mathematics and Computer Science, University of Antwerp.
2. M. Thomas Sauer, Full Professor, Department of Mathematics and Computer Science, University of Passau.

### Examineurs

1. M. Pierre Common, Directeur de Recherche, CNRS, gipsa-lab
2. M. Bernhard Beckermann, Professeur des universités

### Invités

1. M. Bernard Mourrain, Directeur de Recherche, INRIA Sophia Antipolis
2. M. Mustapha Jazar, Professeur, LaMA, Université Libanaise
3. M. Houssam Khalil, Professeur associé, LaMA, Université Libanaise

## Résumé

On étudie la décomposition de matrice de Hankel  $H_\sigma$  comme une somme des matrices de Hankel de rang faible en corrélation avec la décomposition de son symbole  $\sigma$  comme une somme des séries exponentielles polynomiales. On présente un nouvel algorithme qui calcule la décomposition d'un opérateur de Hankel de petit rang et sa décomposition de son symbole en exploitant les propriétés de l'algèbre quotient de Gorenstein  $\mathcal{A}_\sigma$ . La base de  $\mathcal{A}_\sigma$  est calculée à partir la décomposition en valeurs singuliers d'une sous-matrice de matrice de Hankel  $H_\sigma$ . Les fréquences et les poids se déduisent des vecteurs propres généralisés des sous matrices de Hankel déplacés de  $H_\sigma$ . On présente une formule pour calculer les poids en fonction des vecteurs propres généralisés au lieu de résoudre un système de Vandermonde. Cette nouvelle méthode est une généralisation de Pencil méthode déjà utilisée pour résoudre un problème de décomposition de type de Prony. On analyse son comportement numérique en présence des moments contaminés et on décrit une technique de redimensionnement qui améliore la qualité numérique des fréquences d'une grande amplitude. On présente une nouvelle technique de Newton qui converge localement vers la matrice de Hankel de rang faible la plus proche au matrice initiale et on montre son effet à corriger les erreurs sur les moments.

On étudie la décomposition d'un tenseur multi-symétrique  $T$  comme une somme des puissances de produit des formes linéaires en corrélation avec la décomposition de son dual  $T^*$  comme une somme pondérée des évaluations. On utilise les propriétés de l'algèbre de Gorenstein associée  $\mathcal{A}_\tau$  pour calculer la décomposition de son dual  $T^*$  qui est définie à partir d'une série formelle  $\tau$ . On utilise la décomposition d'un opérateur de Hankel de rang faible  $H_\tau$  associé au symbole  $\tau$  comme une somme des opérateurs indécomposables de rang faible. La base d' $\mathcal{A}_\tau$  est choisie de façon que la multiplication par certains variables soit possible. On calcule les coordonnées des points et leurs poids correspondants à partir la structure propre des matrices de multiplication. Ce nouvel algorithme qu'on propose marche bien pour les matrices de Hankel de rang faible. On propose une approche théorique de la méthode dans un espace de dimension  $n$ . On donne un exemple numérique de la décomposition d'un tenseur multilinéaire de rang 3 en dimension 3 et un autre exemple de la décomposition d'un tenseur multi-symétrique de rang 3

en dimension 3. On étudie le problème de complétion de matrice de Hankel comme un problème de minimisation. On utilise la relaxation du problème basé sur la minimisation de la norme nucléaire de la matrice de Hankel.

On adapte le SVT algorithm pour le cas d'une matrice de Hankel et on calcule l'opérateur linéaire qui décrit les contraintes du problème de minimisation de norme nucléaire.

On montre l'utilité du problème de décomposition à dissocier un modèle statistique ou biologique.

### **Mots-clés**

Hankel, polynôme, série exponentielle, décomposition de rang faible, valeurs propres, vecteurs propres, décomposition en valeurs singuliers, tenseur symétrique, tenseur multi-symétrique..

## Abstract

We study the decomposition of a multivariate Hankel matrix  $H_\sigma$  as a sum of Hankel matrices of small rank in correlation with the decomposition of its symbol  $\sigma$  as a sum of polynomial-exponential series. We present a new algorithm to compute the low rank decomposition of the Hankel operator and the decomposition of its symbol exploiting the properties of the associated Artinian Gorenstein quotient algebra  $\mathcal{A}_\sigma$ . A basis of  $\mathcal{A}_\sigma$  is computed from the Singular Value Decomposition of a sub-matrix of the Hankel matrix  $H_\sigma$ . The frequencies and the weights are deduced from the generalized eigenvectors of pencils of shifted sub-matrices of  $H_\sigma$ . Explicit formula for the weights in terms of the eigenvectors avoid us to solve a Vandermonde system. This new method is a multivariate generalization of the so-called Pencil method for solving Prony-type decomposition problems. We analyse its numerical behaviour in the presence of noisy input moments, and describe a rescaling technique which improves the numerical quality of the reconstruction for frequencies of high amplitudes. We also present a new Newton iteration, which converges locally to the closest multivariate Hankel matrix of low rank and show its impact for correcting errors on input moments.

We study the decomposition of a multi-symmetric tensor  $T$  as a sum of powers of product of linear forms in correlation with the decomposition of its dual  $T^*$  as a weighted sum of evaluations. We use the properties of the associated Artinian Gorenstein Algebra  $\mathcal{A}_\tau$  to compute the decomposition of its dual  $T^*$  which is defined via a formal power series  $\tau$ . We use the low rank decomposition of the Hankel operator  $H_\tau$  associated to the symbol  $\tau$  into a sum of indecomposable operators of low rank. A basis of  $\mathcal{A}_\tau$  is chosen such that the multiplication by some variables is possible. We compute the sub-coordinates of the evaluation points and their weights using the eigen-structure of multiplication matrices. The new algorithm that we propose works for small rank. We give a theoretical generalized approach of the method in  $n$  dimensional space. We show a numerical example of the decomposition of a multi-linear tensor of rank 3 in 3 dimensional space.

We show a numerical example of the decomposition of a multi-symmetric tensor of rank 3 in 3 dimensional space.

We study the completion problem of the low rank Hankel matrix as a

minimization problem. We use the relaxation of it as a minimization problem of the nuclear norm of Hankel matrix. We adapt the SVT algorithm to the case of Hankel matrix and we compute the linear operator which describes the constraints of the problem and its adjoint.

We try to show the utility of the decomposition algorithm in some applications such that the LDA model and the ODF model.

**Keywords**

Hankel, polynomial, exponential series, low rank decomposition, eigenvector, Singular Value Decomposition, symmetric tensor, multi-symmetric tensor.

I would like to start my thesis by thanking everyone who has contributed to it and to my personal development through this scientific project.

To name a few,

I would like to thank M. Bernard Mourrain for his transparency, honesty, patience and commitment. M. Bernard is an exceptional mentor; he continues to guide me by encouraging me through my daily life and by pushing me to find answers through my thesis. He has aided me with the development of both my thesis and myself, and through him, I have become a more confident and holistic person, and for that I am truly grateful.

Furthermore, I extend my thank you:

To AROMATH—for accepting me as a PHD candidate and for providing me with the financial and technical materials needed for my thesis work.

To M. Ammar Assoum and M. Bilal Barakeh from Lebanese University—for helping me with the complicated administrative procedure.

To M. André Gallico for being by my side through my thesis.

And To Mme. Sophie Honnorat—for being the wonderful person that she is, and for standing by me as both a mentor and determined assistant. Mme. Sophie is incredibly humane, helpful and she inspires me to be my best self everyday.

I would also like to thank the jury, for taking the time to read and comment on my thesis and for moving to follow the defense, and I would like to thank my thesis, for being my hardest and most enjoyable project and for allowing me to be the person that I am today, and last but certainly not least, I would like to thank my parents—for believing in me, encouraging me, and making it possible for me to be here today.

Everyone in the last three years has made a permanent impact on my life, and for that, i would thank them.



# Contents

|          |                                                         |           |
|----------|---------------------------------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>                                     | <b>9</b>  |
| 1.1      | Context and Literature Review . . . . .                 | 9         |
| 1.2      | Approach and Methodology . . . . .                      | 12        |
| 1.3      | Contributions . . . . .                                 | 16        |
| 1.4      | Thesis Structure . . . . .                              | 17        |
| <b>2</b> | <b>Hankel Decomposition Problem</b>                     | <b>18</b> |
| 2.1      | Duality and Hankel Operators . . . . .                  | 19        |
| 2.1.1    | Duality . . . . .                                       | 19        |
| 2.1.2    | Polynomial-exponential series . . . . .                 | 21        |
| 2.1.3    | Differential Operators . . . . .                        | 21        |
| 2.1.4    | Z-transform . . . . .                                   | 23        |
| 2.1.5    | Hankel Operators . . . . .                              | 23        |
| 2.2      | Structured decomposition of Hankel matrices . . . . .   | 26        |
| 2.2.1    | Solving polynomial equations by eigenvector computation | 26        |
| 2.3      | Generalized Kronecker Theorem . . . . .                 | 30        |
| 2.3.1    | Undecomposable Series . . . . .                         | 32        |
| 2.3.2    | Decomposition of Formal Power Series . . . . .          | 33        |
| 2.3.3    | Decomposition Algorithm . . . . .                       | 35        |
| 2.4      | Experimentation . . . . .                               | 39        |
| 2.4.1    | Reconstruction using Fourier Coefficients . . . . .     | 40        |
| 2.4.2    | Numerical example . . . . .                             | 40        |
| 2.4.3    | Numerical behavior against perturbation . . . . .       | 42        |
| 2.4.4    | Rescaling . . . . .                                     | 44        |
| 2.4.5    | Numerical Rank . . . . .                                | 45        |

|          |                                                                                      |            |
|----------|--------------------------------------------------------------------------------------|------------|
| <b>3</b> | <b>Tensor Decomposition Problem</b>                                                  | <b>48</b>  |
| 3.1      | Symmetric and Multi-Symmetric Tensors . . . . .                                      | 49         |
| 3.1.1    | Symmetric and Multi-Symmetric tensors . . . . .                                      | 49         |
| 3.1.2    | Symmetric Tensor . . . . .                                                           | 49         |
| 3.1.3    | Non symmetric Tensor . . . . .                                                       | 50         |
| 3.1.4    | Multi-Symmetric Tensor . . . . .                                                     | 56         |
| 3.2      | Rank One Tensors from a Geometric Point of View . . . . .                            | 57         |
| 3.2.1    | Segré variety . . . . .                                                              | 58         |
| 3.2.2    | Veronese variety . . . . .                                                           | 58         |
| 3.2.3    | Segré-Veronese variety . . . . .                                                     | 59         |
| 3.2.4    | Rank and Border Rank . . . . .                                                       | 60         |
| 3.3      | Symmetric and Non Symmetric Tensor Decomposition problem in the literature . . . . . | 61         |
| 3.3.1    | Power Method . . . . .                                                               | 62         |
| 3.3.2    | ALS algorithm . . . . .                                                              | 70         |
| 3.3.3    | Simultaneous Eigenvalue Decomposition Method . . . . .                               | 72         |
| 3.4      | Multi Symmetric Tensor Decomposition Problem . . . . .                               | 78         |
| 3.5      | Non Symmetric Tensor Decomposition Problem . . . . .                                 | 80         |
| 3.6      | Example . . . . .                                                                    | 84         |
| <b>4</b> | <b>Tensors in Learning Latent Variable Models</b>                                    | <b>89</b>  |
| 4.1      | Discret Distributions . . . . .                                                      | 89         |
| 4.2      | Continuous Distributions . . . . .                                                   | 91         |
| 4.2.1    | Multivariate Gaussian Distribution . . . . .                                         | 93         |
| 4.2.2    | Dirichlet Distribution . . . . .                                                     | 94         |
| 4.3      | Learning Latent Variable Models . . . . .                                            | 95         |
| 4.3.1    | Exchangeable Single Topic Model . . . . .                                            | 95         |
| 4.3.2    | Spherical Gaussian Mixtures . . . . .                                                | 96         |
| 4.4      | Latent Dirichlet Allocation . . . . .                                                | 98         |
| 4.5      | Numerical Results . . . . .                                                          | 100        |
| <b>5</b> | <b>Tensor in Fiber Crossing Detection Model</b>                                      | <b>103</b> |
| 5.1      | Extracting Coefficients of Fiber Crossing Tensor . . . . .                           | 104        |
| 5.2      | Algorithm . . . . .                                                                  | 107        |
| 5.2.1    | Numerical Results . . . . .                                                          | 109        |

|          |                                                           |            |
|----------|-----------------------------------------------------------|------------|
| <b>6</b> | <b>Completion of Hankel Matrix of Low Rank</b>            | <b>112</b> |
| 6.1      | Semidefinite Programming . . . . .                        | 113        |
| 6.2      | Minimization Rank Problem-RMP . . . . .                   | 114        |
| 6.2.1    | RMP Semidefinite Positive . . . . .                       | 114        |
| 6.3      | Relaxations: Trace, Nuclear Norm . . . . .                | 115        |
| 6.3.1    | Matrix Norms . . . . .                                    | 115        |
| 6.3.2    | Trace Minimization . . . . .                              | 117        |
| 6.3.3    | X is Symmetric Non Semidefinite Positive . . . . .        | 118        |
| 6.4      | General Case: Minimization of Nuclear Norm . . . . .      | 119        |
| 6.5      | Convex Hull of the Rank function . . . . .                | 122        |
| 6.6      | Numerical Example SDP . . . . .                           | 124        |
| 6.7      | Singular Value Thresholding Algorithm . . . . .           | 126        |
| 6.7.1    | Algorithm Outline . . . . .                               | 127        |
| 6.8      | Completion of Hankel matrix using SVT algorithm . . . . . | 132        |
| <b>7</b> | <b>Newton Iteration</b>                                   | <b>135</b> |
| <b>8</b> | <b>Conclusions and Perspectives</b>                       | <b>144</b> |

# Chapter 1

## Introduction

### 1.1 Context and Literature Review

The problem of decomposition of Hankel matrices into sum of low rank matrices of the same structure appears in some problems such as the reconstruction of signals from Fourier Coefficients, the recovery of a sparse polynomial by interpolation or the reconstruction from moments. An algebraic or statistical model provide the coefficients of Hankel matrix which helps to describe the phenomenon. An efficient way to analyze the structure of the underlying model is to decompose the Hankel matrix associated to the coefficients of the model into sum of Hankel matrices of low rank.

Natural questions arise. What are the indecomposable Hankel matrices? Are they necessarily of rank 1? How to compute a decomposition of a Hankel matrix as a sum of indecomposable Hankel matrices? Is the structured low rank decomposition of a Hankel matrix unique? These questions have simple answers for non-structured or dense matrices: The indecomposable dense matrices are the matrices of rank one, which are the tensor product of two vectors. The Singular Value Decomposition of a dense matrix yields a decomposition as a minimal sum of rank one matrices, but this decomposition is not unique. It turns out that for the Hankel structure, the answers to these questions are not so direct and involve the analysis of the so-called symbol associated to the Hankel matrix. The symbol is a formal power series defined from the coefficients of the Hankel matrix. As we will see, the structured decomposition of an Hankel matrix is closely related to the decomposition of the symbol as a sum of polynomial-exponential series.

Our research focus is to extend the Prony method [PT14] to more general Hankel matrices and tensors. An important goal is to overcome the numerical instability of the algebraic decomposition method [Mou16] which comes from the sensitivity of eigenvalues of Hankel matrix. We apply the Singular Value Thresholding process on the Hankel matrix in that purpose. We show the efficiency of the method by linking it to some real statistical and biological models. In decomposition problems, it is sometimes required to complete the associated Hankel matrix with the missing elements in order to obtain the desired reconstruction results. In that objective, we propose an optimization algorithms to minimize the nuclear norm of Hankel matrix.

Several works have been developed in one dimensional case such as Prony method which constructs sum of exponentials from equally spaced values by computing a polynomial in the kernel of a Hankel matrix and by deducing the decomposition from the roots of the polynomial [PT14]. Another type of methods which is called Pencil method computes the generalized eigenvalues of a pencil of Hankel matrices instead of computing a recurrence relation [PS10]. Other optimization techniques which implement a variable projection algorithm have been proposed such as MUSIC [SK92] and ESPRIT [RK90]. The decomposition problem has also been studied in the multivariate case [PT13], [KPRv16]. These methods are applicable when the dimension of Hankel matrices is high enough to recover the multivariate solutions from some projections in one dimension.

In [Sau17], Sauer extends Prony method for interpolation polynomial problem to the multivariate case by constructing an  $H$ -basis and a graded homogeneous basis for an inverse system  $N$  of normal forms based on multivariate Hankel matrices. He computes multiplication matrices with respect to the graded basis and he uses the eigenvectors of these matrices to compute the points of the decomposition which are the coefficient vectors of interpolation polynomials. He finally uses a Vandermonde system to compute the weights. In [Sau18], he uses the concept of universal interpolation as a weak generalization of univariate Chebychev systems to estimate the rank of Hankel operator associated to the interpolation polynomial needed to solve Prony's problem.

In [BCL17], Cuyt adapts the Prony method in one and more variables to

the reconstruction of sparse signals from Fourier Coefficients where she uses a smaller number of moments and which is of low computational cost. It has been recently shown in [CL18] that the minimal number of moments sufficient to recover the multivariate exponential or the sparse interpolation is equal to  $(n+1)*r$  where  $r$  is the rank of Hankel operator and  $n$  is the dimension of the problem. In [CTVL18], they compute the multivariate exponential decomposition of noisy moments by viewing it as a Padé Approximation problem when it is difficult to detect the origin of noise. They model separately the noise and the signal instead of removing the noise at an initial step and return a low order approximation.

In [Mou16], Mourrain uses a projection process, similar to Gram-Schmidt orthogonalization process in order to compute the orthogonal basis.

The problem of decomposition of symmetric and multi-symmetric tensors into sum of tensors of low enough rank appears in many domains such as statistics [SGB00], neuroscience [MKBD14], phylogenetic trees model, signal processing [DLC07] and so on... For example, the study of symmetric tensor gives us an idea about the geometric structure of intersecting fibers in human brain [MKBD14]. It helps to analyze the content of a corpus of web pages as a mixture of several topics.

A symmetric tensor is a tensor whose components stay invariant by any permutation of indices. A non symmetric tensor is in correspondence with a multi-linear map from a product of vector spaces to the coefficient field.

Important efforts have been developed over the last decades to better understand the theoretical and algorithmic aspects of the symmetric and non symmetric tensor decomposition problem. Some of them use local optimization techniques such as Alternative Least Squares [DLDMV04] which fixes all factor matrices but one at each time and its convergence relies on the initial choices. Other like Simultaneous Eigenvalue Decomposition method relies on the diagonalization of a collection of similar matrices obtained from the given tensor [DLDMV04] and Robust Tensor Power method computes a sequence of vectors which converges to the eigenvector which corresponds to the largest eigenvalue [AGH<sup>+</sup>15]. Gradient Descent and Quazi-Newton are either used to minimize the distance between the tensor and its decomposition [XY13][CGT90]. Homotopy techniques have been either developed recently

to compute this decomposition [BDHM17].

To decompose a tensor associated to a real mathematical model, we sometimes do not have enough number of moments to apply the direct method of decomposition. In some applications of tensor decomposition problem [KB09], we compute the missing data in order to satisfy the constraints of the direct decomposition. The problem of recovering of an unknown Hankel matrix from a small number of entries appears in many applications such that the LDA model where the rank of the matrix is bigger than the dimension [HA15]. The completion problem is a rank minimization problem which is NP hard and hard to solve [Faz02].

## 1.2 Approach and Methodology

The thesis extends some algebraic tools, mostly developed in [Mou16], to obtain multiplication matrices using some shifted submatrices of the Hankel matrix  $H_\sigma$  and to use their eigenvalues connected by joint eigenvectors to compute the decomposition of the symbol  $\sigma$  associated to  $H_\sigma$ . We analyze the algebraic, geometric and algorithmic aspects of the decomposition problems. We apply numerically stable linear algebra tools on the submatrices of Hankel matrix like the singular value decomposition process in order to obtain better decompositions.

We define the Hankel matrix  $H_\sigma$  associated to the formal power series  $\sigma$  and we recall Kronecker theorem [Kro81] which establishes a correspondence between the decomposition of  $H_\sigma$  as a sum of Hankel matrices of small rank and the decomposition of its symbol  $\sigma$  as a sum polynomial-exponential series. In [Ped99] [OP01] [HT04] and [Mou16], it has been shown that the dual of the ring of polynomials  $\mathbb{K}[\mathbf{x}] = \mathbb{K}[x_1, \dots, x_n]$  is isomorphic to the set of formal power series  $\mathbb{K}[\mathbf{y}] = \mathbb{K}[y_1, \dots, y_n]$ , then a formal power series  $\sigma$  can be seen as a linear form on polynomials  $p(\mathbf{x}) \in \mathbb{K}[\mathbf{x}]$ . Using the properties of the inner product  $\langle \cdot, \cdot \rangle_\sigma$  defined on  $\mathbb{K}[\mathbf{x}]$  associated to  $\sigma$ , we see that the polynomial  $p(\mathbf{x})$  can be used as a differential operator  $p(\partial)(\sigma)$  on series  $\sigma$  which allows us to solve a system of differential equations. Following [Mou16], we show that if  $I_\sigma$  is the kernel of  $H_\sigma$  then the elements in the dual of finite dimensional quotient Algebra  $\mathcal{A}_\sigma = \mathbb{K}[\mathbf{x}]/I_\sigma$  are computed in terms of polynomial-exponential

functions associated to the inverse system of a characteristic variety  $\mathcal{V}(I_\sigma)$  of roots of polynomials in  $I_\sigma$ . The decomposition of  $H_\sigma$  is also related to the decomposition of the quotient algebra  $A_\sigma$  in terms of sub-algebras  $A_{\xi_i}$  associated to the finite number of roots  $\xi_i$  in  $\mathcal{V}_\sigma$ . We recall techniques exploiting the eigenstructure properties of multiplication operators  $\mathcal{M}_{x_i}$  by the variables  $x_i$  for solving polynomial systems and we show how they can be used in particular for the resolution of decomposition problem. The multiplication matrices  $M_{x_i}$  by the variables  $x_i$  in an orthogonal basis of  $A_\sigma$  are computed using some shifted sub-matrices  $H_{x_i * \sigma}$  called pencils of the Hankel matrix. An orthogonal basis of  $A_\sigma$  can be extracted from any maximal non-zero minor of the matrix of  $H_\sigma$ . We show how to compute an orthogonal basis of  $A_\sigma$  using the Singular Value Decomposition of a sub-matrix of  $H_\sigma$ . We describe precisely how to compute the roots  $\xi_i$  of polynomial-exponential decomposition of  $\sigma$  from the generalized eigenvectors of multiplication matrices  $M_{x_i}$  in the fixed basis. We also show that the weights  $\omega_i$  of the decomposition can be recovered directly from the eigenvectors of  $M_{x_i}$  without the resolution of a Vandermonde system. This process is a multivariate generalization of the so-called Pencil method for solving Prony-type decomposition problems. The Pencil method has been used before to decompose series as a sum of polynomial-exponential functions from a fixed number of moments  $\sigma = (\sigma_\alpha)_\alpha$ . We also analyze the numerical behavior of the decomposition method in presence of noisy input moments for different dimensions, different degrees of exponential terms in the decomposition and different amplitudes of roots. We present a rescaling technique, which improves the numerical quality of the reconstruction of frequencies with high amplitudes. We present a new Newton iteration, which converges locally to the multivariate Hankel matrix of a given rank the closest to a given input Hankel matrix. Numerical experimentations show that the Newton iteration combined the decomposition method allows to compute accurately and efficiently the polynomial-exponential decomposition of the symbol, even for noisy input moments.

We recall definitions of symmetric, non-symmetric and multi symmetric tensors of low rank [BBCM13] and we give some basic operations on them like "how to convert a tensor to a matrix and how to convert a matrix to a vector". We describe the projective variety of rank one tensors for each



family of them. We recall approximation techniques which minimize the distance between the tensor  $T$  and its decomposition as a finite weighted sum of product of powers of linear forms such as Robust Tensor Power method and Alternative Least Square algorithm and Simultaneous Eigenvalue Decomposition technique. Because of the dual of the tensor  $T^*$  can be defined as a formal power series  $\tau$  using the so-called apolar product [BBCM13], we show that the decomposition of multi-symmetric tensor  $T$  as a finite sum of product of powers of linear forms coincides with the decomposition of its dual  $T^*$  as a finite weighted sum of evaluations  $\mathbf{e}_{\xi_i}$ . We recast the low rank decomposition method of Hankel operator into a sum of indecomposable operators of low rank to the case of non-symmetric tensor decomposition problem. Our direct method extends the techniques of [SK90] to more general tensors and to tensors of higher rank. It is closely connected to the multivariate Prony method investigated in [Mou16]. We recover the points and weights from eigenvectors of multiplication operators of the quotient algebra associated to the decomposition. The algorithm does not require the solution of polynomial equations. We split the coordinates of each variable  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  into three bunches of sub-variables  $\mathbf{x}$ ,  $\mathbf{y}$  and  $\mathbf{z}$  and we exploit the structure of quotient algebra  $A_\tau$  of the three dimensional space  $\mathbb{K}[\mathbf{x}, \mathbf{y}, \mathbf{z}]$  by  $I_\tau$  to solve the non-symmetric tensor decomposition problem. We choose two bases  $A_1$  and  $A_2$  of monomials such that all given moments of the tensor  $T$  appear in the matrix  $H_\tau$  associated to  $T$  in the bases  $A_1$  and  $A_2$  and we substitute  $x_0$  by one. We compute the Singular Value Decomposition of the Hankel matrix  $H_\tau$  in order to extract an orthogonal basis  $B_2$  in which the multiplication matrices  $M_{y_j}^{B_2}$  by one collection of variables  $y_j$  are well defined. We exploit the eigen-structure properties of multiplication matrices  $M_{y_j}^{B_2}$  to compute each bunch of coordinates  $(a_{i,p}), (b_{i,q}), (c_{i,l})$  of each point  $\xi_i = (a_{i,p}, b_{i,q}, c_{i,l})$  associated to  $\mathbf{x}, \mathbf{y}$  and  $\mathbf{z}$  separately and their corresponding weights  $\omega_i$ . We show the constraints which arise from the computation of all multiplication matrices in higher dimension spaces.

We recall the definition of Multivariate Gaussian Distribution and Dirichlet Distribution. We recall the definition of Exchangeable single topic model which describes the web pages content of a corpus as a mixture of topics and shows the usage of the symmetric tensor decomposition problem through the

model [AGH<sup>+</sup>15]. We recall the Latent Dirichlet Allocation model where the topic mixture in the corpus follows a Dirichlet Distribution and proved theoretically how to compute the symmetric tensor decomposition using the cross moments  $M_1 = P(x_1)$ ,  $M_2 = P(x_1 \otimes x_2)$  and  $M_3 = P(x_1 \otimes x_2 \otimes x_3)$  which describe the model. We give a numerical example applied on a big corpus of documents and passes through a whitening step using Julia and we analyze the result. Each point  $\xi_i$  of the decomposition represent the probability vector of each topic, and each weight  $\omega_i$  represents the mass of each topic in the corpus.

In order to describe the geometric structure of intersecting fibers in human brain, we recall a method which extracts the coefficients of the so called ODF tensor [MKBD14]. We use the symmetric tensor decomposition method to compute the weights  $\omega_i$  and the directions  $\xi_i$  of fibers in a voxel. All fibers separated with an angle bigger than  $30^\circ$  are reconstructed using the decomposition algorithm.

Using Semi Definite programming, a relaxation of the Rank minimization problem is presented to minimize the trace of the matrix when it is semi definite positive or the nuclear norm when it is not semi definite positive neither symmetric. We adapt these two heuristics to the case of Hankel matrix with some known entries. We propose another relaxation of the problem which minimizes the nuclear norm of the matrix. We adapt the singular value thresholding SVT algorithm which is a type of Lagrangian algorithm to minimize the nuclear norm of the matrix when it is Hankel with small number of entries. This algorithm converges if the threshold is big enough even when the matrix is of big dimensions [CCS10]. We adapt the singular value thresholding SVT algorithm to minimize the nuclear norm of the Hankel matrix with few number of known entries. This iterative algorithm produces a sequence of matrices  $(X^k, Y^k)$  and at each step performs a soft thresholding algorithm operation on the singular values of the matrix  $Y^k$  which consists only of keeping the singular values which are bigger than threshold and moving them towards zero. The choice of a big threshold reduces the storage space at each iteration and the computational cost. It is proved that the iterates of the algorithm converge under the condition of big threshold. We adapt the nuclear norm minimization problem to the Hankel case. We compute the

linear operator  $\mathcal{A}(X)$  which describes the constraints of the low nuclear norm minimization problem in the case of Hankel matrix with a fixed number of known entries and its transpose  $\mathcal{A}^\top(y)$ . In some cases the completion does not provide a Hankel matrix, we call the newton iteration to minimize the distance between the matrix and its representation as a Hankel matrix. We use Julia to do the implementations.

We propose a new Newton iteration which helps to remove noise on the moments of input series along with the multivariate Hankel decomposition method. We give numerical examples with MAPLE which shows the efficiency of Newton method to reduce the maximum absolute error between input frequencies and output frequencies and the maximum error between input weights and output weights. We give another detailed numerical example with Julia which minimizes the distance between a Hankel matrix after SVT step and its direct decomposition where the last one does not give the desired weights and frequencies.

### 1.3 Contributions

The main contributions of this thesis can be resumed as follows:

- Low rank multivariate Hankel matrix decomposition algorithm where the computation of multiplication matrices is done through a singular value decomposition of the Hankel matrix. The minimal number of terms in the decomposition is the same of the numerical rank of the Hankel matrix.
- The combination of new Newton iteration with the decomposition step to compute accurately and efficiently the polynomial-exponential decomposition of the symbol, even for noisy input moments.

Thesis work leads to the publication of two papers:

- **Linear Algebra and Applications:**

Publication of Special Issue entitled with: *Structured low rank decomposition of multivariate Hankel matrices*. J. Harmouch, B. Mourrain, H. Khalil, volume: 542, pages: 162-185, 1 April 2018.

- **MACIS 2017:** International Conference on Mathematical Aspects of Computer and Information Sciences.

Presentation and publication in the proceedings of the conference of a paper entitled with: *Decomposition of Low Rank Multi-Symmetric Tensor*. Jouhayna Harmouch, Bernard Mourrain, Houssam Khalil. Publisher: Springer, pages: 51-66, 15 November 2017.

## 1.4 Thesis Structure

The thesis is developed through the following chapters:

- Chapter 2. We introduce the notion of multivariate Hankel operator and we describe the decomposition algorithm of it in presence of noisy input moments and high amplitude frequencies.
- Chapter 3. We recall the definitions of symmetric and multi-symmetric tensors and we describe the existing decomposition methods of them. We propose a new algorithm to decompose a non-symmetric tensor.
- Chapter 4. We introduce the statistical model which describes the web pages content of a corpus of documents. We extract the symmetric tensor structure from the moments which model the statistical phenomenon.
- Chapter 5. We recall the algorithm which gives the coefficients of biological model which describes the geometric structure of fibers in human brain. We analyze the symmetric tensor structure behind the model.
- Chapter 6. We introduce the SDP's to solve RMP if the objective matrix is Hankel. We recast the Singular Value Thresholding algorithm to predict the missing elements of a Hankel matrix of low nuclear norm.
- Chapter 7. We propose a Newton iteration which helps to remove noise on moments of series. We adapt this method to minimize the distance between the Hankel matrix after SVT step and its decomposition.
- Chapter 8. We recast the main results of the thesis.

## Chapter 2

# Hankel Decomposition Problem

In this chapter we set the identification among the set of sequences, the ring of formal power series and the dual of the ring of polynomials. We first recall the definition of polynomial-exponential series. We show how series acts as differential operators on polynomials. In the opposite, a polynomial can be seen as a differential operator on series. We review definitions of *Z-transform* and *Borel transform* which relate a formal power series in variables  $\mathbf{y}$  to a power series in  $\mathbf{z}$ . We show that a variable  $x_i$  acts as a shift operators on a series in  $\mathbf{z}$ . We define a Hankel operator (resp. Hankel) matrix in univariate case and multivariate case. We give the definition of truncated Hankel operator (resp. truncated Hankel operator) associated to a formal power series in a basis of quotient algebra and its dual basis. We propose a method which decomposes a Hankel operator as a sum of indecomposable polynomial exponential series which reduces to the solution of polynomial equations. Using Kronecker Theorem, we establish a correspondence between Artinian Gorenstein Algebra, a Hankel operator of low rank and polynomial exponential series. We recall techniques for solving equations by eigenvalue and eigenvector computation of multiplication matrices and we apply them to the Hankel case. We use the first right and left singular vectors of a truncated Hankel matrix to define an orthogonal base of all multiplication operators which guarantee a better numerical computation of weights and points of the decomposition. We give a simple formula which computes weights in the orthogonal basis. We propose an efficient algorithm which shows the decomposition of a formal power series of low rank into weighted sum of evaluations in all steps. We analyse its numerical behaviour in the presence of noisy input moments, for different

numbers of variables, of exponential terms of the symbol and different amplitudes of the frequencies. We present a rescaling technique, which improves the numerical quality of the reconstruction for frequencies of high amplitudes.

## 2.1 Duality and Hankel Operators

We consider a field  $\mathbb{K}$  of characteristic 0. In applications  $\mathbb{K} = \mathbb{R}$  ou  $\mathbb{C}$ . We are going to use the following notations:  $R = \mathbb{K}[\mathbf{x}] = \mathbb{K}[x_1, \dots, x_n]$  is the ring of polynomials in the variables  $x_1, \dots, x_n$  with coefficients in the field  $\mathbb{K}$ ,  $\mathbb{K}[[\mathbf{y}]] = \mathbb{K}[[y_1, \dots, y_n]]$  is the ring of formal power series in the variables  $y_1, \dots, y_n$  with coefficients in the field  $\mathbb{K}$ . For  $\alpha, \beta \in \mathbb{N}^n$  multi-index exponents we say that  $\alpha \ll \beta$  if  $\alpha_i \leq \beta_i$  for  $i = 1, \dots, n$ .

### 2.1.1 Duality

The natural isomorphism between the set of linear forms on the ring of polynomials  $(\mathbb{K}[\mathbf{x}])^* = \text{Hom}_{\mathbb{K}}(\mathbb{K}[\mathbf{x}], \mathbb{K})$  and  $\mathbb{K}[[\mathbf{y}]]$  is defined via the pairing:

$$\mathbb{K}[[\mathbf{y}]] \times \mathbb{K}[\mathbf{x}] \rightarrow \mathbb{K} \quad (2.1)$$

$$(\mathbf{y}^\alpha, \mathbf{x}^\beta) \mapsto \begin{cases} \alpha! & \text{if } \alpha = \beta. \\ 0 & \text{otherwise.} \end{cases} \quad (2.2)$$

where  $\alpha! = \prod_{i=1}^n \alpha_i!$  for  $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{N}^n$ .

If  $\sigma \in (\mathbb{K}[\mathbf{x}])^*$  is a linear form, it can be represented by the series

$$\sigma(\mathbf{y}) = \sum_{\alpha \in \mathbb{N}^n} \sigma(\mathbf{x}^\alpha) \frac{\mathbf{y}^\alpha}{\alpha!} \in \mathbb{K}[[\mathbf{y}]].$$

In the opposite, any series  $\sigma(\mathbf{y}) = \sum_{\alpha \in \mathbb{N}^n} \sigma_\alpha \frac{\mathbf{y}^\alpha}{\alpha!} \in \mathbb{K}[[\mathbf{y}]]$  can be interpreted as a linear form such that  $\langle \sigma | p \rangle = \sum_{\alpha \in A \subset \mathbb{N}^n} p_\alpha \sigma_\alpha$  where  $p(\mathbf{x}) = \sum_{\alpha \in A \subset \mathbb{N}^n} p_\alpha \mathbf{x}^\alpha$ . The linear form  $\sigma$  is uniquely identified by its coefficients  $\sigma_\alpha = \langle \sigma | \mathbf{x}^\alpha \rangle$  for  $\alpha \in \mathbb{N}^n$ , which are called the *moments* of  $\sigma$ .

The set of multi-index sequences  $\mathbb{K}^{\mathbb{N}^n}$  can be identified with the ring of formal power series  $\mathbb{K}[[y_1, \dots, y_n]] = \mathbb{K}[[\mathbf{y}]]$ . A sequence  $\sigma = (\sigma_\alpha)_\alpha \in \mathbb{K}^{\mathbb{N}^n}$  is identified with the series

$$\sigma(\mathbf{y}) = \sum_{\alpha \in \mathbb{N}^n} \sigma_\alpha \frac{\mathbf{y}^\alpha}{\alpha!} \in \mathbb{K}[[\mathbf{y}]]$$

where  $\mathbf{y}^\alpha = y_1^{\alpha_1} \dots y_n^{\alpha_n}$ ,  $\alpha! = \prod_{i=1}^n \alpha_i!$  for  $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{N}^n$ .

For an ideal  $I \subset \mathbb{K}[\mathbf{x}]$ , we denote by  $I^\perp$  the space of linear forms orthogonal to all polynomials of  $I$  via the scalar product.

For a vector space  $D \subset \mathbb{K}[\mathbf{y}]$ , we denote by  $D^\perp$  the space of polynomials orthogonal to all power linear forms of  $D$  via the scalar product.

The truncation of an element  $\sigma(\mathbf{y})$  in degree  $d \in \mathbb{N}^*$  is obtained by taking all moments with associated monomials of degree lower than  $d$ . It is denoted by  $\sigma(\mathbf{y})((\mathbf{y}))^{d+1}$  which is the class of  $\sigma$  modulo the ideal  $(y_1, \dots, y_n)^{d+1} \in \mathbb{K}[[\mathbf{y}]]$ .

For an ideal  $I \subset \mathbb{K}[\mathbf{x}]$  of polynomials, we have  $I^{\perp\perp} = I$ .

For a vector space  $D \subset \mathbb{K}[[\mathbf{y}]]$  of formal power series, if  $D$  is closed for the  $\mathbf{y}$ -adic topology than  $D^{\perp\perp} = D$ .

The dual space  $\mathbb{K}[\mathbf{x}]^* \equiv \mathbb{K}[[\mathbf{y}]]$  has a natural structure of  $\mathbb{K}[\mathbf{x}]$ -module, defined as follows:

$$\forall \sigma(\mathbf{y}) \in \mathbb{K}[[\mathbf{y}]], \forall p(\mathbf{x}), q(\mathbf{x}) \in \mathbb{K}[\mathbf{x}], \langle p(\mathbf{x}) \star \sigma(\mathbf{y}) \mid q(\mathbf{x}) \rangle = \langle \sigma(\mathbf{y}) \mid p(\mathbf{x})q(\mathbf{x}) \rangle.$$

We check that  $\forall \sigma(\mathbf{y}) \in \mathbb{K}[[\mathbf{y}]], \forall p(\mathbf{x}), q(\mathbf{x}) \in \mathbb{K}[\mathbf{x}], (pq) \star (\sigma) = p \star (q \star \sigma)$ . For more details, see [Ems78] and [Mou97].

For a polynomial  $p = \sum_{\alpha \in \mathbb{N}^n} p_\alpha \mathbf{x}^\alpha$  with  $p_\alpha = 0$  for  $\alpha \in \mathbb{N}^n$ , we have the expansion series

$$p \star \sigma = \sum_{\beta \in \mathbb{N}^n} \left( \sum_{\alpha \in \mathbb{N}^n} p_\alpha \sigma_{\alpha+\beta} \right) \frac{\mathbf{y}^\alpha}{\alpha!}.$$

The sequence associated to this series  $(\sum_{\alpha \in \mathbb{N}^n} p_\alpha \sigma_{\alpha+\beta})_{\alpha \in \mathbb{N}^n}$  is called the cross-correlation sequence of  $p$  and  $\sigma$ .

We also identify  $\mathbb{K}[\mathbf{x}]$  with the set of sequences with a finite support  $L_0(\mathbb{K}^{\mathbb{N}^n})$  i.e. the set of sequences with a finite number of non-zero terms.

For  $\sigma \in \mathbb{K}[\mathbf{y}]$ , the inner product associated to  $\sigma$  on  $\mathbb{K}[\mathbf{x}]$  is defined as

$$\langle p, q \rangle_\sigma := \langle \sigma(\mathbf{y}) \mid p(\mathbf{x})q(\mathbf{x}) \rangle$$

for all  $p(\mathbf{x}), q(\mathbf{x}) \in \mathbb{K}[\mathbf{x}]$ .

### 2.1.2 Polynomial-exponential series

We define the evaluation at a point  $\xi = (\xi_1, \xi_2, \dots, \xi_n) \in \mathbb{K}^n$  which is an element of  $\mathbb{K}[\mathbf{x}]^*$  as follows:

$$\mathbf{e}_\xi : \mathbb{K}[x_1, x_2, \dots, x_n] \rightarrow \mathbb{K} \quad (2.3)$$

$$p(\mathbf{x}) = p(x_1, x_2, \dots, x_n) \mapsto p(\xi) = p(\xi_1, \xi_2, \dots, \xi_n) \quad (2.4)$$

It corresponds to the series

$$\mathbf{e}_\xi(\mathbf{y}) = \sum_{\alpha \in \mathbb{N}^n} \xi^\alpha \frac{\mathbf{y}^\alpha}{\alpha!} = e^{\xi_1 y_1 + \xi_2 y_2 + \dots + \xi_n y_n} = e^{\langle \xi | \mathbf{y} \rangle} \in \mathbb{K}[[\mathbf{y}]].$$

**Definition 2.1.1.** *Let*

$$\mathcal{POLYEXP}(\mathbf{y}) = \left\{ \sigma = \sum_{i=1}^r \omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}(\mathbf{y}) \in \mathbb{K}[[\mathbf{y}]] \mid \xi_i \in \mathbb{K}^n, \omega_i(\mathbf{y}) \in \mathbb{K}[\mathbf{y}] \right\}$$

*be the set of polynomial-exponential series. The polynomials  $\omega_i(\mathbf{y})$  are called the weights of  $\sigma$  and  $\xi_i$  the frequencies.*

Notice that the product of  $\mathbf{y}^\alpha \mathbf{e}_\xi(\mathbf{y})$  by the monomial  $\mathbf{x}^\beta$  can be seen as a differential operator of the monomial evaluated at the point  $\xi$  such that:

$$\begin{cases} \frac{\beta!}{(\beta-\alpha)!} \xi^{\beta-\alpha} = \partial_{x_1}^{\alpha_1} \partial_{x_2}^{\alpha_2} \dots \partial_{x_n}^{\alpha_n} (\mathbf{x}^\beta)(\xi) & \text{if } \alpha \ll \beta, \\ 0 & \text{otherwise.} \end{cases}$$

So that, the sum of polynomial-exponential series can be seen as a sum of polynomial-differential operators  $\omega_i(\partial)$  at  $\xi$  such that for  $\sigma = \sum_{i=1}^r \omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}(\mathbf{y})$

$$\forall p \in \mathbb{K}[\mathbf{x}], \langle \sigma | p \rangle = \sum_{i=1}^r \omega_i(\partial)(p)(\xi).$$

### 2.1.3 Differential Operators

An interesting property of the outer product is that the polynomial act as differential on the series:

**Lemma 2.1.2.**  $\forall p \in \mathbb{K}[\mathbf{x}], \forall \sigma \in \mathbb{K}[\mathbf{y}], p(\mathbf{x}) * \sigma(\mathbf{y}) = p(\partial)(\sigma)$

*Proof.* We prove  $x_i * (\mathbf{y})^\alpha = \partial_{y_i}(\mathbf{y}^\alpha)$ . See [Mou16]. □



**Lemma 2.1.3.**  $\forall p \in \mathbb{K}[\mathbf{x}], \forall \omega \in \mathbb{K}[\mathbf{y}], p(\mathbf{x}) * (\omega(\mathbf{y})\mathbf{e}_\xi(\mathbf{y})) = p(\xi_1 + \partial_{y_1}, \xi_2 + \partial_{y_2}, \dots, \xi_n + \partial_{y_n})((\omega(\mathbf{y}))\mathbf{e}_\xi(\mathbf{y})).$

*Proof.* By the previous lemma,  $x_i * (\omega(\mathbf{y}))\mathbf{e}_\xi(\mathbf{y}) = (\xi_i + \partial_{y_i})((\omega(\mathbf{y}))\mathbf{e}_\xi(\mathbf{y}))$  for  $i = 1, \dots, n$ . By repeated multiplications by the variables and linear combination, the equality is true for any polynomial  $p \in \mathbb{K}[\mathbf{x}]$ .  $\square$

**Definition 2.1.4.** For a subset  $D \subset \mathbb{K}[\mathbf{y}]$  the inverse system generated by  $D$  is the vector space generated by the formal power series of  $D$  and all their derivatives. For  $\omega(\mathbf{y}) \in \mathbb{K}[\mathbf{y}]$ , the inverse system of  $\omega(\mathbf{y})$  is generated by  $\omega(\mathbf{y})$  and all its derivatives  $\partial^\alpha(\omega), \alpha \in \mathbb{N}^n$ .

Its dimension is denoted by  $\mu(\omega)$ . We compute it using the following lemma:

**Lemma 2.1.5.** For  $\omega(\mathbf{y}) \in \mathbb{K}[\mathbf{y}]$ ,  $\mu(\omega)$  is the rank of the matrix  $\Theta = (\theta_{\alpha,\beta})_{\alpha \in A, \beta \in B}$  where  $\omega(\mathbf{y} + \mathbf{t}) = \sum_{\alpha \in A, \beta \in B} \theta_{\alpha,\beta} \mathbf{y}^\alpha \mathbf{t}^\beta$  for some finite subsets  $A, B$  of  $\mathbb{N}^n$ .

*Proof.* We prove it using the Taylor expansion of  $\omega(\mathbf{y} + \mathbf{t})$  at  $\mathbf{y}$ .  $\square$

**Lemma 2.1.6.** The series  $\mathbf{y}^{\alpha_{i,j}} \mathbf{e}_{\xi_i}(\mathbf{y})$  for  $i = 1, \dots, r$  and  $j = 1, \dots, \mu_i$  where  $\alpha_{i,1}, \alpha_{i,2}, \dots, \alpha_{i,\mu_i} \in \mathbb{N}^n$  and (resp.  $\xi_i \in \mathbb{K}^n$ ) distinct in pairs are linearly independent.

*Proof.* We suppose that there exists  $\omega_{i,j} \in \mathbb{K}$  such that

$$\sigma = \sum_{i=1}^r \omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}(\mathbf{y}) = \sum_{i=1}^r \sum_{j=1}^{\mu_i} \omega_{i,j} \mathbf{y}^{\alpha_{i,j}} \mathbf{e}_{\xi_i}(\mathbf{y}) = 0.$$

We need to prove that  $\omega_i(\mathbf{y}) = 0$  for all  $i = 1, \dots, r$ . If  $\sigma = 0$  then by lemma 2.1.3 and linearity we obtain  $p * \sigma = \sum_{i=1}^r p(\xi_i + \partial)((\omega_i(\mathbf{y}))\mathbf{e}_{\xi_i}(\mathbf{y}))$ .

We distinguish two cases:

If  $\deg(w_i(\mathbf{y})) = 0$ , then if we choose  $p$  as an interpolation polynomial at one of the distinct roots  $\xi_i$  then  $w_i(\mathbf{y}) = 0$  for  $i = 1, \dots, r$ .

If  $\deg(w_i(\mathbf{y})) \geq 1$ , then if we choose  $p(\mathbf{x}) = l(\mathbf{x}) - l(\xi_i)$  for a separating polynomial  $l$  of degree one ( $l(\xi_i) \neq l(\xi_j)$  if  $i \neq j$ ), so that at least one of  $w_i(\mathbf{y})$  has one degree less. By induction on the degree, we obtain  $\omega_i(\mathbf{y}) = 0$  for  $i = 1, \dots, r$ .  $\square$

### 2.1.4 Z-transform

We can replace  $\frac{y^\alpha}{\alpha!}$  by  $\mathbf{z}^\alpha$  where  $\mathbf{z} = (z_1, z_2, \dots, z_n)$  is a set of new variables. So that the sequence  $(\sigma_\alpha)_{\alpha \in \mathbb{N}^n}$  can be represented by the formal power series:

$$\sigma(\mathbf{y}) = \sum_{\alpha \in \mathbb{N}^n} \sigma_\alpha \mathbf{z}^\alpha \in \mathbb{K}[[\mathbf{z}]]$$

which is called the *Z-transform* of the sequence  $(\sigma_\alpha)_{\alpha \in \mathbb{N}^n}$  or the embedding in the ring of *divided powers*. The inverse transformation from a series in  $\mathbb{K}[[\mathbf{z}]]$  to a series in  $\mathbb{K}[[\mathbf{y}]]$  is called the *Borel Transform*. We then have the natural isomorphism between  $\mathbb{K}[\mathbf{x}]^*$  and  $\mathbb{K}[[\mathbf{z}]]$  and we can extend the properties of duality to any field which is not of characteristic 0.

$$\text{For } \alpha, \beta \in \mathbb{N}^n, \text{ we have } \mathbf{x}^\alpha * \mathbf{z}^\beta = \begin{cases} \mathbf{z}^{\beta-\alpha} & \text{if } \beta - \alpha \in \mathbb{N}^n \\ 0 & \text{otherwise} \end{cases}.$$

So that  $z_i$  plays the role of the inverse of  $x_i$  and  $x_i$  acts as a shift operators on the series such that:

$$x_i * \sigma(\mathbf{y}) = x_i * \sum_{\alpha \in \mathbb{N}^n} \sigma_\alpha \mathbf{z}^\alpha = \sum_{\alpha \in \mathbb{N}^n} \sigma_{\alpha + e_i} \mathbf{z}^\alpha \in \mathbb{K}[[\mathbf{z}]].$$

The evaluation  $\mathbf{e}_\xi$  is represented in  $\mathbb{K}[\mathbf{z}]$  by the rational function

$$\frac{1}{\prod_{j=1}^n (1 - \xi_j z_j)}.$$

The series  $\mathbf{y}^\alpha \mathbf{e}_\xi(\mathbf{y})$  in  $\mathbb{K}[\mathbf{y}]$  is then represented by the series

$$\frac{\alpha! \mathbf{z}^\alpha}{\prod_{j=1}^n (1 - \xi_j z_j)^{1+\alpha_j}} \in \mathbb{K}[\mathbf{z}].$$

### 2.1.5 Hankel Operators

#### Univariate Hankel Operators

*Hankel matrices* are structured matrices of the form

$$H = [\sigma_{i+j}]_{0 \leq i \leq l, 0 \leq j \leq m}$$

where the entry  $\sigma_{i+j}$  of the  $i^{\text{th}}$  row and the  $j^{\text{th}}$  columns depends only on the sum  $i + j$ . By reversing the order of the columns or the rows, we obtain

*Toeplitz matrices*, which entries depend on the difference of the row and column indices. Exploiting their structure leads to superfast methods for many linear algebra operations such as matrix-vector product, solution of linear systems, ... [BP12]

A Hankel matrix is a sub-matrix of the matrix of the *Hankel operator* associated to a sequence  $\sigma = (\sigma_k) \in \mathbb{K}^{\mathbb{N}}$ :

$$\begin{aligned} H_\sigma : L_0(\mathbb{K}^{\mathbb{N}}) &\rightarrow \mathbb{K}^{\mathbb{N}} \\ (p_k)_k &\mapsto \left( \sum_k p_k \sigma_{k+l} \right)_{l \in \mathbb{N}} \end{aligned}$$

where  $L_0(\mathbb{K}^{\mathbb{N}})$  is the set of sequences of  $\mathbb{K}^{\mathbb{N}}$  with a finite support.

### Multivariate Hankel Operators

The multivariate Hankel operator is nothing else than the operator of multiplication by  $\sigma$ :

$$\begin{aligned} H_\sigma : \mathbb{K}[\mathbf{x}] &\rightarrow \mathbb{K}[[\mathbf{y}]] \\ p &\mapsto p * \sigma \end{aligned}$$

The multivariate Hankel operator can be seen as an operator on sequences if we associate to each polynomial of finite support  $p = (p_\alpha)_{\alpha \in A \subset \mathbb{N}^n}$ , the sequence  $p * \sigma = (\sum_{\alpha \in \mathbb{N}^n} p_\alpha \sigma_{\alpha+\beta})_{\beta \in \mathbb{N}^n}$ . The kernel denoted by  $I_\sigma = \ker H_\sigma$  is the set of polynomials  $p = \sum_{\beta \in B} p_\beta \mathbf{x}^\beta$  such that  $\sum_{\beta \in B} p_\beta \sigma_{\alpha+\beta} = 0$  for all  $\alpha \in \mathbb{N}^n$  and it is called the set of linear recurrence relations of the sequence  $\sigma = (\sigma_\alpha)_{\alpha \in \mathbb{N}^n}$ .

Because of  $pq * \sigma = p * q * \sigma$  for all  $p, q \in \mathbb{K}[\mathbf{x}]$  we can easily check that  $I_\sigma$  is an ideal of  $\mathbb{K}[\mathbf{x}]$  and the quotient space  $A_\sigma = \mathbb{K}[\mathbf{x}]/I_\sigma$  defines an algebra.

The formal power series  $\sigma$  is called the symbol of  $H_\sigma$ .

**Definition 2.1.7.** *The rank of an element  $\sigma \in \mathbb{K}[[\mathbf{y}]]$  is the rank of the Hankel operator  $H_\sigma = r$ .*

The multivariate Hankel operator can also be interpreted using the *Z-transform* of the cross-correlation of  $p$  and  $\sigma$  by associating to it, the series  $\sigma(\mathbf{z}) = \sum_{\beta \in \mathbb{N}^n} (\sum_{\alpha \in \mathbb{N}^n} p_\alpha \sigma_{\alpha+\beta}) \mathbf{z}^\beta$  in  $\mathbb{K}[[\mathbf{z}]]$ .

The *Multivariate Hankel matrix* associated to the Hankel operator in the basis  $(\mathbf{x}^\alpha)_{\alpha \in \mathbb{N}^n}$  and the dual basis  $(\frac{\mathbf{y}^\beta}{\beta!})_{\beta \in \mathbb{N}^n}$  have a structure of the form

$$H = [\sigma_{\alpha+\beta}]_{\alpha, \beta \in \mathbb{N}^n} = (\langle \sigma | \mathbf{x}^{\alpha+\beta} \rangle)_{\alpha, \beta \in \mathbb{N}^n}.$$

**Example 2.1.8.** The series  $\sigma(\mathbf{y}) = \mathbf{e}_\xi(\mathbf{y}) = \sum_{\alpha \in \mathbb{N}^n} \xi^\alpha \frac{\mathbf{y}^\alpha}{\alpha!} = e^{\mathbf{y} \cdot \xi}$  for  $\xi \in \mathbb{K}^n$ , represents the linear functional corresponding to the evaluation  $\mathbf{e}_\xi$  at  $\xi$ . The Hankel operator  $H_{\mathbf{e}_\xi}$  is of rank 1, since its image is spanned by  $\mathbf{e}_\xi(\mathbf{y})$  and  $I_{\mathbf{e}_\xi} = (x_1 - \xi_1, x_2 - \xi_2, \dots, x_n - \xi_n)$ .

For  $A, B \subset \mathbb{N}^n$  subsets of multi-indices indexing respectively the rows and columns, the Hankel matrix of  $\mathbf{e}_\xi$  is denoted by  $H_\xi^{A,B} = [\xi^{\beta+\alpha}]_{\beta \in B, \alpha \in A}$ . If  $H_\xi^{A,B} \neq 0$ , it is a matrix of rank 1.

### Truncated Hankel Operators

Truncated Hankel operators are obtained by restriction of Hankel operators.

**Definition 2.1.9.** For two vector spaces  $U, V \subset \mathbb{K}[\mathbf{x}]$  and  $\sigma \in \langle U, V \rangle^* = \langle u.v \mid u \in U, v \in V \rangle^* \subset \mathbb{K}[[\mathbf{y}]]$ , we denote by  $H_\sigma^{U,V}$  the following map:

$$\begin{aligned} H_\sigma^{U,V} : V &\rightarrow U^* \\ p(\mathbf{x}) &\mapsto p(\mathbf{x}) * \sigma(\mathbf{y})|_U \end{aligned}$$

It is called the truncated Hankel operator on  $(U, V)$ . If  $U = V$ , the truncated Hankel operator is denoted by  $H_\sigma^V$ .

For  $U = \{u_1, \dots, u_l\} \subset \mathbb{K}[\mathbf{x}]$ ,  $V = \{v_1, \dots, v_m\} \subset \mathbb{K}[\mathbf{x}]$ , the Hankel matrix of  $\sigma$  on  $U, V$  is  $H_\sigma^{U,V} = (\langle \sigma \mid u_i v_j \rangle)_{1 \leq i \leq l, 1 \leq j \leq m}$ . We use the same notation  $H_\sigma^{U,V}$  for the truncated Hankel operator from  $\langle V \rangle$  to  $\langle U \rangle^*$ .

For  $A, B \subset \mathbb{N}^n$ , let  $\langle \mathbf{x}^B \rangle \subset \mathbb{K}[\mathbf{x}]$ ,  $\langle \mathbf{y}^A \rangle \subset \mathbb{K}[[\mathbf{y}]]$  be the vector spaces spanned respectively by the monomials  $\mathbf{x}^\beta$  for  $\beta \in B$  and  $\mathbf{y}^\alpha$  for  $\alpha \in A$ . The truncated Hankel operator of  $\sigma$  on  $A, B$  is

$$\begin{aligned} H_\sigma^{A,B} : \langle \mathbf{x}^B \rangle &\rightarrow \langle \mathbf{y}^A \rangle \\ p = \sum_{\beta \in B} p_\beta \mathbf{x}^\beta &\mapsto \sum_{\alpha \in A} (\sum_{\beta \in B} p_\beta \sigma_{\alpha+\beta}) \frac{\mathbf{y}^\alpha}{\alpha!} = p * \sigma|_{\langle \mathbf{x}^A \rangle} \end{aligned}$$

The matrix of  $H_\sigma^{A,B}$  in the bases  $(\mathbf{x}^\beta)_{\beta \in B}$  and  $(\frac{\mathbf{y}^\alpha}{\alpha!})_{\alpha \in A}$  is of the form:

$$H_\sigma^{A,B} = [\sigma_{\alpha+\beta}]_{\alpha \in A, \beta \in B}.$$

It is also called the *truncated moment matrix* of  $\sigma$ .

For  $d \in \mathbb{N}$ , we denote by  $\mathbb{K}[\mathbf{x}]_d$  the vector space of polynomials of total degree  $\leq d$ . Its dimension is  $s_d = \binom{n+d}{n}$ . For  $d, d' \in \mathbb{N}$ , we denote by  $H_\sigma^{d,d'}$  the

Hankel matrix of  $\sigma$  on the subset of monomials in  $\mathbf{x}$  respectively of degree  $\leq d$  and  $\leq d'$ . We also denote by  $H_\sigma^{d,d'}$  the corresponding truncated Hankel operator of  $H_\sigma$  from  $\mathbb{K}[\mathbf{x}]_{d'}$  to  $(\mathbb{K}[\mathbf{x}]_d)^*$ .

Multivariate Hankel matrices have a structure, which can be exploited to accelerate linear algebra operations.

**Example 2.1.10.** Consider the series  $\sigma = 1 + 2y_1 + 3y_2 + 4\frac{y_1^2}{2} + 5y_1y_2 + 6\frac{y_2^2}{2} + 7\frac{y_1^3}{6} + 8\frac{y_1^2y_2}{2} + \dots \in \mathbb{R}[[y_1, y_2]]$ . Its truncated Hankel matrix on  $A = [(0, 0), (1, 0), (0, 1)]$  (corresponding to the monomials  $1, x_1, x_2$ ),  $B = [(0, 0), (1, 0), (0, 1), (2, 0)]$  (corresponding to the monomials  $1, x_1, x_2, x_1^2$ ) is

$$H_\sigma^{A,B} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 5 & 7 \\ 3 & 5 & 6 & 8 \end{bmatrix}.$$

## 2.2 Structured decomposition of Hankel matrices

In this section, we show how the decomposition of the symbol  $\sigma$  of a Hankel operator  $H_\sigma$  as a sum of polynomial-exponential series reduces to the solution of polynomial equations. This corresponds to the decomposition of  $H_\sigma$  as a sum of Hankel matrices of low rank. We first recall classical techniques for solving polynomial systems and show how these methods can be applied on the Hankel matrix  $H_\sigma$ , to compute the decomposition.

### 2.2.1 Solving polynomial equations by eigenvector computation

**Definition 2.2.1.** A quotient algebra  $\mathcal{A} = \mathbb{K}[\mathbf{x}]/I$  is Artinian if it is of finite dimension over  $\mathbb{K}$ .

Notice that if  $\mathbb{K}$  is a subfield of  $\mathbb{L}$ ,  $I_L = I \otimes L$  is the ideal of  $\mathbb{L}[\mathbf{x}]$  generated by the elements of  $I$ , we denote by  $\mathcal{A}_L = \mathbb{L}[\mathbf{x}]/I_L = \mathcal{A} \otimes L$ . We have  $\dim_{\mathbb{K}}(\mathcal{A}) = \dim_{\mathbb{L}}(\mathcal{A}_L)$ , so if  $\mathbb{L} = \overline{\mathbb{K}}$  is the algebraic closure of  $\mathbb{K}$  we obtain  $\dim_{\mathbb{K}}(\mathcal{A}) = \dim_{\overline{\mathbb{K}}}(\mathcal{A})$ . In the following, we assume that  $\mathbb{K} = \overline{\mathbb{K}}$ .

**Theorem 2.2.2.**  $\dim_{\mathbb{K}}(\mathcal{A}) \leq \infty$  if and only if  $I$  defines a finite number of isolated points in  $\overline{\mathbb{K}}^n$ .

*Proof.* See [LCO92] [Theorem 6]. □

**Theorem 2.2.3.** *Let  $\mathcal{A} = \mathbb{K}[\mathbf{x}]/I$  be an Artinian algebra of dimension  $r$  defined by an ideal  $I$ . In the case that the ideal  $I$  defines a finite number of roots  $\mathcal{V}(I) = \{\xi_1, \dots, \xi_{r'}\} = \{\xi \in \mathbb{K}^n \mid \forall p \in I, p(\xi) = 0\}$  where  $r' \leq r$ , we have a decomposition of  $\mathcal{A}$  as a sum of sub-algebras:*

$$\mathcal{A} = \mathbb{K}[\mathbf{x}]/I = \mathcal{A}_1 \oplus \dots \oplus \mathcal{A}_{r'}$$

where

- $I = Q_1 \cap Q_2 \cap \dots \cap Q_{r'}$  is a minimal primary decomposition of  $I$  where  $Q_i$  is the  $\mathbf{m}_{\xi_i}$ -primary component of  $I$  associated to the root  $\xi_i \in \mathbb{K}^n$  such that  $\mathbf{m}_{\xi_i} = (x_1 - \xi_{i,1}, x_2 - \xi_{i,2}, \dots, x_n - \xi_{i,n})$
- The local algebra  $\mathcal{A}_i = \mathbf{u}_{\xi_i} \mathcal{A} \equiv \mathbb{K}[\mathbf{x}]/Q_i$  where  $\mathcal{A}_i \mathcal{A}_j \equiv 0$  for  $i \neq j$ .

The dimension of  $\mathcal{A}_i$  is the *multiplicity* of the point  $\xi_i$ . For more details, see [EM07][Chap. 4].

The projection of 1 on the local algebras  $\mathcal{A}_i$  yields the so-called idempotents  $\mathbf{u}_{\xi_i}$  associated to the roots  $\xi_i$ , which satisfy the relations:

$$\mathbf{u}_{\xi_i}(\mathbf{x})\mathbf{u}_{\xi_j}(\mathbf{x}) \equiv 0, \mathbf{u}_{\xi_i}^2(\mathbf{x}) \equiv \mathbf{u}_{\xi_i}(\mathbf{x}), \sum_{i=1}^{r'} \mathbf{u}_{\xi_i}(\mathbf{x}) \equiv 1.$$

for  $i \neq j = 1, \dots, r'$ .

$\mathcal{A}^*$  is identified with the subspace  $I^\perp$  of formal power series which are orthogonal to the polynomials of  $I$ . As  $I$  is stable by multiplication by  $x_i$ , then  $I^\perp$  is stable by derivation  $\frac{d}{dy_i}$  for  $i = 1, \dots, n$ .

**Proposition 2.2.4.** *Let  $Q$  be a primary ideal for the maximal ideal  $\mathbf{m}_\xi$  of the point  $\xi \in \mathbb{K}^n$  and let  $\mathcal{A}_\xi = \mathbb{K}[\mathbf{x}]/Q$  which is stable by derivations  $\frac{d}{dy_i}$  for  $i = 1, \dots, n$ . Then*

$$Q^\perp = \mathcal{A}_\xi^* = D_\xi(Q) \cdot \mathbf{e}_\xi(\mathbf{y}),$$

where  $D_\xi(Q) \subset \mathbb{K}[\mathbf{y}]$  is the set of polynomials  $\omega[\mathbf{y}] \in \mathbb{K}[\mathbf{y}]$  such that  $\forall q \in Q, \omega(\partial)(q)(\xi) = 0$ .

The vector space  $D_\xi(Q)$  is called inverse system of  $Q$ .

By the duality applied on the direct sum of sub-algebras which is equal to  $\mathcal{A}$ , any  $\sigma \in \mathcal{A}^*$  can be decomposed as  $\sum_{i=1}^{r'} \mathbf{u}_{\xi_i} * \sigma = \sigma$ . Because of  $(\mathbf{u}_{\xi_i} * \sigma)(\mathcal{A}_{\xi_i}) = 0$  we deduce that  $\mathbf{u}_{\xi_i} * \sigma \in \mathcal{A}_{\xi_i}^* = Q_i^\perp$  for  $i = 1, \dots, r$ .

$D_{\xi_i}(Q_i)$  is denoted by  $D_i$  for  $i = 1, \dots, r$ .

**Theorem 2.2.5.** *Let  $I = (p_1, p_2, \dots, p_s)$  be an ideal of  $\mathbb{K}[\mathbf{x}]/I$  which defines a finite number of roots  $\xi_1, \xi_2, \dots, \xi_r$ . The generating series of the sequence  $\sigma = (\sigma_\alpha)_{\alpha \in \mathbb{N}^n}$  which satisfies the system of difference equations*

$$p_1 * \sigma = 0, p_2 * \sigma = 0, \dots, p_s * \sigma = 0$$

are of the form

$$\sigma(\mathbf{y}) = \sum_{\alpha \in \mathbb{N}^n} \sigma_\alpha \frac{\mathbf{y}^\alpha}{\alpha!} = \sum_{i=1}^r \omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}(\mathbf{y}) \in \mathbb{K}[[\mathbf{y}]]$$

where  $\omega_i(\mathbf{y}) \in D_i = D(\xi_i)(I) = D(\xi_i)((p_1, p_2, \dots, p_s))$  such that  $\omega(\partial)(p_j)(\xi) = 0$  for all  $j = 1, \dots, s$  and  $i = 1, \dots, r$ .

The points  $\xi_i$  can be recovered using properties of the multiplicative structure of  $\mathcal{A}$ .

**Definition 2.2.6.** *For  $g \in \mathbb{K}[\mathbf{x}]$ , the multiplication operator  $\mathcal{M}_g$  is defined by*

$$\begin{aligned} \mathcal{M}_g: \mathcal{A} &\rightarrow \mathcal{A} \\ h &\mapsto \mathcal{M}_g(h) = g h. \end{aligned}$$

The transpose  $\mathcal{M}_g^\top$  of the multiplication operator  $\mathcal{M}_g$  is

$$\begin{aligned} \mathcal{M}_g^\top: \mathcal{A}^* &\rightarrow \mathcal{A}^* \\ \Lambda &\mapsto \mathcal{M}_g^\top(\Lambda) = \Lambda \circ \mathcal{M}_g = g * \Lambda. \end{aligned}$$

Let  $B = \{b_1, \dots, b_r\}$  is a basis of  $\mathcal{A}$ , and  $B^*$  its dual basis in  $\mathcal{A}^*$ . We denote by  $M_g$  (resp.  $M_g^\top$ ) is the matrix of  $\mathcal{M}_g$  (resp.  $\mathcal{M}_g^\top$ ) in the basis  $B$  (resp.  $B^*$ ) of  $\mathcal{A}$  (resp.  $\mathcal{A}^*$ ). We have  $M_g^\top = (M_g)^\top$  and they have the same eigenvalues.

The main property that we will use to recover the roots is the following [EM07][Thm. 4.23]:

**Proposition 2.2.7.** *Let  $I$  be an ideal of  $\mathbb{K}[\mathbf{x}]$  and suppose that  $\mathcal{V}(I) = \{\xi_1, \xi_2, \dots, \xi_{r'}\}$ . Then*

- for all  $g \in \mathcal{A}$ , the eigenvalues of  $M_g$  and  $M_g^\top$  are the values  $g(\xi_1), \dots, g(\xi_r)$  of the polynomial  $g$  at the roots with multiplicities  $\mu_i = \dim \mathcal{A}_{\xi_i}$ .
- The eigenvectors common to all  $M_g^\top$  with  $g \in \mathcal{A}$  are - up to a scalar - the evaluations  $\mathbf{e}_{\xi_1}, \dots, \mathbf{e}_{\xi_r}$ .

In the case of simple roots, we have the following property [EM07][Chap. 4]:

**Proposition 2.2.8.** *If the roots  $\{\xi_1, \xi_2, \dots, \xi_r\}$  of  $I$  are simple (i.e.  $\mu_i = \dim \mathcal{A}_i = 1$ ) then we have the following:*

- $\mathfrak{u} = \{\mathbf{u}_{\xi_1}, \dots, \mathbf{u}_{\xi_r}\}$  is a basis of  $\mathcal{A}$ .
- The polynomials  $\mathbf{u}_{\xi_1}, \dots, \mathbf{u}_{\xi_r}$  are interpolation polynomials at the roots  $\xi_i$ :  $\mathbf{u}_{\xi_i}(\xi_j) = 1$  if  $i = j$  and 0 otherwise.
- The matrix of  $M_g$  in the basis  $\mathfrak{u}$  is the diagonal matrix  $\text{diag}(g(\xi_1), \dots, g(\xi_r))$ .

This proposition tells us that if  $g$  is separating the roots, i.e.  $g(\xi_i) \neq g(\xi_j)$  for  $i \neq j$ , then the eigenvectors of  $M_g$  are, up to a scalar, interpolation polynomials at the roots.

The coefficient vector of the evaluation  $\mathbf{e}_{\xi_i} = \sum_{\beta \in \mathbb{N}^n} \xi_i^\beta \frac{\mathbf{y}^\beta}{\beta!} + \dots$  in the dual basis of  $B$  is  $[\langle \mathbf{e}_{\xi_i} | b_j \rangle]_{\beta \in B} = [b_j(\xi_i)]_{i=1 \dots r} = B(\xi_i)$ . The previous proposition says that

$$M_g^\top B(\xi_i) = g(\xi_i) B(\xi_i).$$

If moreover the basis  $B$  contains the monomials  $1, x_1, x_2, \dots, x_n$ , then the common eigenvectors of  $M_g^\top$  are of the form  $\mathbf{v}_i = c [1, \xi_{i,1}, \dots, \xi_{i,n}, \dots]$  and the root  $\xi_i$  can be computed from the coefficients of  $\mathbf{v}_i$  by taking the ratio of the coefficients of the monomials  $x_1, \dots, x_n$  by the coefficient of 1:  $\xi_{i,k} = \frac{\mathbf{v}_{i,k+1}}{\mathbf{v}_{i,1}}$ . Thus computing the common eigenvectors of all the matrices  $M_g^\top$  for  $g \in \mathcal{A}$  yields the roots  $\xi_i$  ( $i = 1, \dots, r$ ).

In practice, it is enough to compute the common eigenvectors of  $M_{x_1}^\top, \dots, M_{x_n}^\top$ , since  $\forall g \in \mathbb{K}[\mathbf{x}], M_g^\top = g(M_{x_1}^\top, \dots, M_{x_n}^\top)$ . Therefore, the common eigenvectors  $M_{x_1}^\top, \dots, M_{x_n}^\top$  are also eigenvectors of any  $M_g^\top$ .

The multiplicity structure, that is the dual  $Q_i^\perp$  of each primary component  $Q_i$  of  $I$ , also called the *inverse system* of the point  $\xi_i$  can be deduced by linear algebra tools (see e.g. [Mou97]).



**Definition 2.2.9.** A  $\mathbb{K}$ -algebra  $\mathcal{A}$  is Gorenstein if  $\exists \tau \in \mathcal{A}^*$  such that  $\forall \sigma \in \mathcal{A}^*, \exists p \in \mathcal{A}$  with  $\sigma = p * \tau$  and if  $p * \tau = 0$  implies  $p = 0$ .

In other words,  $\mathcal{A}$  is Gorenstein if and only if  $\mathcal{A}^*$  is free  $\mathcal{A}$ -module of rank one.

## 2.3 Generalized Kronecker Theorem

We are interested in structured decompositions of Hankel matrices (resp. operators) as sums of Hankel matrices (resp. operators) of low rank. This raises the question of describing the Hankel operators of finite rank and leads to the problem of decomposing them into indecomposable Hankel operators of low rank. We establish a correspondence among Hankel operators of finite rank, polynomial-exponential series and Artinian Gorenstein Algebra.

We associate to a Hankel operator  $H_\sigma$ , the quotient  $\mathcal{A}_\sigma = \mathbb{K}[\mathbf{x}]/I_\sigma$  of the polynomial ring  $\mathbb{K}[\mathbf{x}]$  modulo the kernel  $I_\sigma = \{p \in \mathbb{K}[\mathbf{x}] \mid \forall q \in R, \langle \sigma \mid pq \rangle = 0\}$  of  $H_\sigma$ . We have checked that  $I_\sigma$  is an ideal of  $\mathbb{K}[\mathbf{x}]$ , so that  $\mathcal{A}_\sigma$  is an algebra.

As  $\mathcal{A}_\sigma = \mathbb{K}[\mathbf{x}]/I_\sigma \sim \text{img } H_\sigma$ , the operator  $H_\sigma$  is of finite rank  $r$ , if and only if,  $\mathcal{A}_\sigma$  is Artinian of dimension  $\dim_{\mathbb{K}} \mathcal{A}_\sigma = r$ .

We recall the celebrated theorem of Kronecker [Kro81]. We will also assume hereafter that  $\overline{\mathbb{K}} = \mathbb{K}$  is algebraically closed.

**Theorem 2.3.1** (Kronecker Theorem). *The Hankel operator*

$$H_\sigma : (p_k) \in L_0(\mathbb{K}^{\mathbb{N}}) \mapsto \left( \sum_k p_k \sigma_{k+l} \right)_{l \in \mathbb{N}} \in \mathbb{K}^{\mathbb{N}}$$

*is of finite rank  $r$ , if and only if, there exist polynomials  $\omega_1, \dots, \omega_{r'} \in \mathbb{K}[y]$  and  $\xi_1, \dots, \xi_{r'} \in \mathbb{K}$  distinct s.t.*

$$\sigma_n = \sum_{i=1}^{r'} \omega_i(n) \xi_i^n$$

*with  $\sum_{i=1}^{r'} (\deg(\omega_i) + 1) = r$ .*

This results says that the Hankel operator  $H_\sigma$  is of finite rank, if and only if, its symbol  $\sigma$  is of the form

$$\sigma(y) = \sum_{n \in \mathbb{N}} \sigma_n \frac{y^n}{n!} = \sum_{i=1}^{r'} \tilde{\omega}_i(y) e^{\xi_i y}$$

for some univariate polynomials  $\tilde{\omega}_i(y) \in \mathbb{K}[y]$  and distinct complex numbers  $\xi_i$   $i = 1, \dots, r'$ . Moreover, the rank of  $H_\sigma$  is  $r = \sum_{i=1}^{r'} (\deg(\tilde{\omega}_i) + 1)$ .

The previous result admits a direct generalization to multivariate Hankel operators, using polynomial-exponential series.

The next theorem characterizes the multivariate Hankel operators of finite rank in terms of their symbol [Mou16]:

**Theorem 2.3.2** (Generalized Kronecker Theorem). *Let  $\sigma(\mathbf{y}) \in \mathbb{K}[[\mathbf{y}]]$ . Then  $\text{rank } H_\sigma = r < \infty$ , if and only if,  $\sigma(\mathbf{y}) = \sum_{i=1}^{r'} \omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}(\mathbf{y}) \in \mathcal{POLYEXP}(\mathbf{y})$  with  $\omega_i(\mathbf{y}) \in \mathbb{K}[\mathbf{y}] \setminus \{0\}$  and  $\xi_i \in \mathbb{K}^n$  pairwise distinct, with  $r = \sum_{i=1}^{r'} \mu(\omega_i)$  where  $\mu(\omega_i)$  is the dimension of the inverse system spanned by  $\omega_i(\mathbf{y})$  and all its derivatives  $\partial^\alpha \omega_i(\mathbf{y})$  for  $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{N}^n$ .*

*Proof.* See [Mou16]. □

The following proposition shows that the frequencies  $\xi_i$  and the weights  $\omega_i$  can be recovered from the ideal  $I_\sigma$  (see [Mou16] for more details):

**Proposition 2.3.3.** *If  $\sigma(\mathbf{y}) = \sum_{i=1}^{r'} \omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}(\mathbf{y})$  with  $\omega_i(\mathbf{y}) \in \mathbb{K}[\mathbf{y}] \setminus \{0\}$  and  $\xi_i \in \mathbb{K}^n$  pairwise distinct, then we have the following properties:*

- *The points  $\xi_1, \xi_2, \dots, \xi_{r'} \in \mathbb{K}^n$  are the common roots of the polynomials in  $I_\sigma = \ker H_\sigma = \{p \in \mathbb{K}[\mathbf{x}] \mid \forall q \in \mathbb{K}[\mathbf{x}], \langle \sigma | pq \rangle = 0\}$ .*
- *The series  $\omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}$  is a generator of the inverse system of  $Q_i^\perp$ , where  $Q_i$  is the primary component of  $I_\sigma$  associated to  $\xi_i$  such that  $\dim \mathbb{K}[\mathbf{x}]/Q_i = \mu(\omega_i)$ .*
- *The inner product  $\langle \cdot, \cdot \rangle_\sigma$  is non-degenerate on  $\mathcal{A}_\sigma$ .*

This result tells us that the problem of decomposing  $\sigma$  as a sum of polynomial-exponential series reduces to the solution of the polynomial equations  $p = 0$  for  $p$  in the kernel  $I_\sigma$  of  $H_\sigma$ .

**Proposition 2.3.4.** *Let  $\sigma(\mathbf{y}) \in \mathbb{K}[\mathbf{y}]$ . The following conditions are equivalent:*

- $\sigma = \sum_{i=1}^{r'} \omega_i \mathbf{e}_{\xi_i}(\mathbf{y})$  with  $\omega_i \in \mathbb{K}[\mathbf{y}]$ ,  $\xi_i \in \mathbb{K}^n$  and  $\sum_{i=1}^{r'} \mu(\omega_i) = r$ ,
- $H_\sigma$  is of rank  $r$ ,

- $\mathcal{A}_\sigma$  is Artinian Gorenstein algebra of dimension  $r$ .

A special case of interest is when the roots are simple.

**Proposition 2.3.5.** *Let  $\sigma(\mathbf{y}) \in \mathbb{K}[\mathbf{y}]$ . The following conditions are equivalent:*

- $\sigma = \sum_{i=1}^r \omega_i \mathbf{e}_{\xi_i}(\mathbf{y})$  with  $\omega_i \in \mathbb{K}$  are non-zeros,  $\xi_i \in \mathbb{K}^n$  are pairwise distinct,
- $H_\sigma$  is of rank  $r$ , and the multiplicity of  $\xi_i$  is 1.
- $(\mathbf{e}_{\xi_1}, \mathbf{e}_{\xi_2}, \dots, \mathbf{e}_{\xi_r})$  is a basis of  $\mathcal{A}_\sigma^*$ .

If  $\sigma$  is with real coefficients, we consider the following property of positivity:

**Proposition 2.3.6.** *The formal power series  $\sigma \in \mathbb{R}[[\mathbf{y}]]$  of finite rank is semi-definite positive if and only if the constant weights in the decomposition are positive and the points belongs to  $\mathbb{R}^n$ .*

*Proof.* The proof relies on the fact that the ideal  $I_\sigma$  is real radical. For more details, see [Mou16].  $\square$

It is saying that a positive measure on  $\mathbb{R}^n$  with an Hankel operator of finite rank  $r$  is a convex combination of  $r$  distinct Dirac measures of  $\mathbb{R}^n$ .

### 2.3.1 Undecomposable Series

The Hankel operators associated to evaluations  $\mathbf{e}_\xi$  are of rank 1. As shown in the next example, a Hankel operator of rank  $> 1$  is not necessarily the sum of Hankel operators of rank 1.

**Example 2.3.7.** *For  $n = 1$  and  $\sigma = y$ , we check that  $H_y$  is of rank 2, but it cannot be decomposed as a sum of two rank-one Hankel operators. If  $A = \{1, x, x^2\}$ , we have*

$$H_y^{A,A} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \neq \lambda_1 \begin{bmatrix} 1 & \xi_1 & \xi_1^2 \\ \xi_1 & \xi_1^2 & \xi_1^3 \\ \xi_1^2 & \xi_1^3 & \xi_1^4 \end{bmatrix} + \lambda_2 \begin{bmatrix} 1 & \xi_2 & \xi_2^2 \\ \xi_2 & \xi_2^2 & \xi_2^3 \\ \xi_2^2 & \xi_2^3 & \xi_2^4 \end{bmatrix}$$

for  $\lambda_1, \lambda_2, \xi_1, \xi_2 \in \mathbb{K}$ . This shows that the symbol  $y$  is indecomposable as a sum of polynomial-exponential series, though it defines an Hankel operator of rank 2.

**Definition 2.3.8.** For  $\sigma \in \mathbb{K}[[\mathbf{y}]]$ , we say that  $\sigma$  is indecomposable if  $\sigma$  cannot be written as a sum  $\sigma = \sigma_1 + \sigma_2$  with  $\text{img } H_\sigma = \text{img } H_{\sigma_1} \oplus \text{img } H_{\sigma_2}$ .

**Proposition 2.3.9.** The series  $\omega(\mathbf{y}) \mathbf{e}_\xi(\mathbf{y})$  with  $\omega(\mathbf{y}) \in \mathbb{K}[\mathbf{y}] \setminus \{0\}$  and  $\xi \in \mathbb{K}^n$  is indecomposable.

*Proof.* Let  $\sigma = \omega \mathbf{e}_\xi$  and  $r = \mu(\omega)$  be the rank of  $H_\sigma$ . Suppose that  $\sigma = \sigma_1 + \sigma_2$  with  $\text{img } H_\sigma = \text{img } H_{\sigma_1} \oplus \text{img } H_{\sigma_2}$ . We assume that the rank of  $H_{\sigma_1}$  is minimal. By the Generalized Kronecker Theorem 2.3.2,  $\sigma_1 = \sum_{i=1}^{r_1} \omega_{1,i} \mathbf{e}_{\xi_{1,i}}$ ,  $\sigma_2 = \sum_{i=1}^{r_2} \omega_{2,i} \mathbf{e}_{\xi_{2,i}}$  with  $\omega_{l,i} \in \mathbb{K}[\mathbf{x}]$ ,  $\xi_{l,i} \in \mathbb{K}^n$  and

$$\omega \mathbf{e}_\xi = \sum_{i=1}^{r_1} \omega_{1,i} \mathbf{e}_{\xi_{1,i}} + \sum_{i=1}^{r_2} \omega_{2,i} \mathbf{e}_{\xi_{2,i}}.$$

By the independence of the polynomial-exponential series (lemma 2.1.6), we can assume that  $\xi_{1,1} = \xi_{2,1} = \xi$  and that  $\omega = \omega_{1,1} + \omega_{2,1}$  (possibly with  $\omega_{2,1} = 0$ ) and that  $\omega_{1,i} = -\omega_{2,i}$  for  $i = 2, \dots, r_1 = r_2$ . As  $\text{rank } H_{\sigma_1} = \sum_{i=1}^{r_1} \mu(\omega_{1,i})$  is minimal, we can assume moreover that  $\omega_{1,i} = 0$  for  $i = 2, \dots, r_1$ , that is,  $r_1 = r_2 = 1$ . Then, we have  $\sigma = \omega \mathbf{e}_\xi$ ,  $\sigma_1 = \omega_1 \mathbf{e}_\xi$ ,  $\sigma_2 = \omega_2 \mathbf{e}_\xi$  with  $\omega = \omega_1 + \omega_2$ . As  $\text{img } H_{\sigma_i} = \langle \partial^\alpha(\omega_i) \mathbf{e}_\xi \rangle$ ,  $i = 1, 2$ , we have  $\text{img } H_{\sigma_1} \cap \text{img } H_{\sigma_2} \ni \mathbf{e}_\xi$  and  $\text{img } H_\sigma$  is not the direct sum of  $\text{img } H_{\sigma_1}$  and  $\text{img } H_{\sigma_2}$ . This shows that  $\sigma$  is indecomposable.  $\square$

### 2.3.2 Decomposition of Formal Power Series

Our goal is to compute the decomposition of the series using eigenvalue et eigenvector structure of non-zero minors of maximal size of the matrix of  $H_\sigma$ . Hereafter  $\mathcal{A} = \mathbb{K}[\mathbf{x}]/I$  is the quotient algebra of  $\mathbb{K}[\mathbf{x}]$  by any ideal  $I$  and  $\mathcal{A}^*$  is the dual of  $\mathcal{A}$ . It is naturally identified with the orthogonal  $I^\perp$ . In the reconstruction problem, we will take  $I = \ker(H_\sigma) = I_\sigma$ .

**Proposition 2.3.10.** Let  $B, B'$  be basis of  $\mathcal{A}_\sigma$  and  $g \in \mathbb{K}[\mathbf{x}]$ . We have

$$H_{g^* \sigma}^{B, B'} = (M_g^B)^\top H_\sigma^{B, B'} = H_\sigma^{B, B'} M_g^{B'}. \quad (2.5)$$

where  $M_g^B$  (resp.  $M_g^{B'}$ ) is the matrix of the multiplication by  $g$  in the basis  $B$  (resp.  $B'$ ) of  $\mathcal{A}_\sigma$ .

*Proof.* Let  $B = \{b_1, \dots, b_r\}$ ,  $B' = \{b'_1, \dots, b'_r\}$  be two bases of  $\mathcal{A}_\sigma$ . We have  $g b_j = \sum_{i=1}^r m_{i,j} b'_i + \kappa$  where  $m_{i,j}$  is the  $(i, j)$  entry of the matrix  $M_g^B$  of multiplication by  $g$  in the basis  $B$  and  $\kappa \in I_\sigma$ . Then,

$$(H_{g^* \sigma}^{B, B'})_{[i,j]} = \langle \sigma \mid g b_j b'_i \rangle = \langle \sigma \mid \sum_{l=1}^r m_{l,i} b_l b'_j \rangle + \langle \sigma \mid \kappa b_j \rangle = \sum_{l=1}^r m'_{l,i} \langle \sigma \mid b_l b'_j \rangle = ((M_g^B)^\top H_\sigma^{B, B'})_{[i,j]}$$

Similarly, we have  $g b'_j = \sum_{i=1}^r m'_{i,j} b_i + \kappa'$  where  $m'_{i,j}$  is the  $(i, j)$  entry of the matrix  $M_g^{B'}$  of multiplication by  $g$  in the basis  $B'$  and  $\kappa' \in I_\sigma$ . For  $1 \leq i, j \leq r$ , the entry  $(i, j)$  of  $H_{g^* \sigma}^{B, B'}$  is

$$(H_{g^* \sigma}^{B, B'})_{[i,j]} = \langle \sigma \mid b_i g b'_j \rangle = \langle \sigma \mid \sum_{l=1}^r m_{l,j} b_i b'_l \rangle + \langle \sigma \mid b_i \kappa' \rangle = \sum_{l=1}^r \langle \sigma \mid b_i b'_l \rangle m_{l,j} = (H_\sigma^{B, B'} M_g^B)_{[i,j]}.$$

This concludes the proof of the relations (2.5).  $\square$

We deduce the following property:

**Proposition 2.3.11.** *Let  $\sigma(\mathbf{y}) = \sum_{i=1}^r \omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}(\mathbf{y})$  with  $\omega_i \in \mathbb{K}[\mathbf{y}] \setminus \{0\}$  and  $\xi_i \in \mathbb{K}^n$  distinct and let  $B, B'$  be bases of  $\mathcal{A}_\sigma$ . We have the following properties:*

- For  $g \in \mathbb{K}[\mathbf{x}]$ ,  $M_g^{B'} = (H_\sigma^{B, B'})^{-1} H_{g^* \sigma}^{B, B'}$ ,  $(M_g^B)^\top = H_{g^* \sigma}^{B, B'} (H_\sigma^{B, B'})^{-1}$ .
- For  $g \in \mathbb{K}[\mathbf{x}]$ , the generalized eigenvalues of  $(H_{g^* \sigma}^{B, B'}, H_\sigma^{B, B'})$  are  $g(\xi_i)$  with multiplicity  $\mu_i = \mu(\omega_i)$ ,  $i = 1, \dots, r$ .
- The generalized eigenvectors common to all  $(H_{g^* \sigma}^{B, B'}, H_\sigma^{B, B'})$  for  $g \in \mathbb{K}[\mathbf{x}]$  are - up to a scalar -  $(H_\sigma^{B, B'})^{-1} B(\xi_i)$ ,  $i = 1, \dots, r$ .

*Proof.* The two first points are direct consequences of Propositions 2.3.10 and 2.2.7. The third point is also a consequence of Proposition 2.2.7, since the coordinate vector of the evaluation  $\mathbf{e}_{\xi_i}$  in the dual basis of  $B$  is  $B(\xi_i)$  for  $i = 1, \dots, r$ .  $\square$

This proposition shows that the matrices of multiplication by an element  $g$  in  $\mathcal{A}$ , and thus the roots  $\{\xi_1, \dots, \xi_r\}$  and their multiplicity structure, can be computed from truncated Hankel matrices, provided we can determine bases  $B, B'$  of  $\mathcal{A}_\sigma$ . In practice, it is enough to compute the generalized eigenvectors common to  $(H_{x_i^* \sigma}^{B, B'}, H_\sigma^{B, B'})$  for  $i = 1, \dots, n$  to recover the roots. As  $H_{x_i^* \sigma}^{B, B'} =$

$H_\sigma^{x_i B, B'} = H_\sigma^{B, x_i B'}$ , the decomposition can be computed from sub-matrices of  $H_\sigma^{B, B'^+}$  or  $H_\sigma^{B^+, B'}$  where  $B^+ = B \cup x_1 B \cup \dots \cup x_n B$ ,  $B'^+ = B' \cup x_1 B' \cup \dots \cup x_n B'$ .

Another property that will be helpful to determine a basis of  $\mathcal{A}_\sigma$  is the following:

**Lemma 2.3.12.** *Let  $B = \{b_1, \dots, b_r\}$ ,  $B' = \{b'_1, \dots, b'_r\} \subset \mathbb{K}[\mathbf{x}]$ . If the matrix  $H_\sigma^{B, B'} = (\langle \sigma | b_i b'_j \rangle)_{1 \leq i, j \leq r}$  is invertible, then  $B$  and  $B'$  are linearly independent in  $\mathcal{A}_\sigma$ .*

*Proof.* Suppose that  $H_\sigma^{B, B'}$  is invertible. If there exists  $p = \sum_i \lambda_i b_i$  ( $\lambda_i \in \mathbb{K}$ ) such that  $p \equiv 0$  in  $\mathcal{A}_\sigma$ . Then  $p \star \sigma = 0$  and  $\forall q \in R$ ,  $\langle \sigma | pq \rangle = 0$ . In particular, for  $j = 1, \dots, r$  we have

$$\sum_{i=1}^r \langle \sigma | b_i b'_j \rangle \lambda_i = 0.$$

As  $H_\sigma^{B, B'}$  is invertible,  $\lambda_i = 0$  for  $i = 1, \dots, r$  and  $B$  is a family of linearly independent elements in  $\mathcal{A}_\sigma$ . Since we have  $(H_\sigma^{B, B'})^\tau = H_\sigma^{B', B}$ , we prove by a similar argument that  $H_\sigma^{B', B}$  invertible also implies that  $B'$  is linearly independent in  $\mathcal{A}_\sigma$ .  $\square$

The converse is not necessarily true. If  $\sigma = y$ , then  $I_\sigma = (x^2)$  and  $B = B' = \{1\}$  are linearly independent in  $\mathcal{A}_\sigma$ , but  $H_\sigma^{B, B'} = (\langle \sigma | 1 \rangle) = (0)$  is not invertible.

### 2.3.3 Decomposition Algorithm

We are given the first moments  $\sigma_\alpha, |\alpha| \leq d$  of the series  $\sigma(\mathbf{y}) = \sum_{i=1}^r \omega_i \mathbf{e}_{\xi_i}(\mathbf{y})$  with  $\omega_i \in \mathbb{C} \setminus (0)$  and  $\xi_i \in \mathbb{C}^n$ . The goal is to recover the number of terms  $r$ , the constant weights  $\omega_i$  and the frequencies  $\xi_i$  of the series  $\sigma(\mathbf{y})$ .

#### Computation of the basis

The first problem is to find automatically bases  $B_1$  and  $B_2$  of the quotient algebra  $\mathcal{A}_\sigma$ , of maximal sizes such that  $H_\sigma^{B_1, B_2}$  is invertible. Using Proposition 2.3.11, we will compute the multiplication matrices  $M_g^{B_2}$  for  $g = x_i, i = 1, \dots, n$ . The frequencies  $\xi_j$  and the weights  $\omega_j, j = 1, \dots, r$  will be deduced from their eigenvectors, as described in section 2.3.3.

Given the set of moments  $(\sigma_\alpha)_{|\alpha| \leq d}$ , we create two sets  $A_1 = (\mathbf{x}^\alpha)_{|\alpha| \leq d_1}$  and  $A_2 = (\mathbf{x}^\beta)_{|\beta| \leq d_2}$  of monomials such that  $\alpha$  and  $\beta$  are multi-indices in  $\mathbb{N}^n$  with

$|\alpha| \leq d_1$  and  $|\beta| \leq d_2$ . The degrees  $d_1$  and  $d_2$  are chosen such that  $d_1 + d_2 < d$ . Let  $N_1 = |A_1|$  and  $N_2 = |A_2|$ . The truncated Hankel operator associated to  $\sigma$  is:

$$\begin{aligned} H_\sigma^{d_1, d_2} : \mathbb{K}[\mathbf{x}]_{d_2} &\rightarrow (\mathbb{K}[\mathbf{x}]_{d_1})^* \\ p &\mapsto p \star \sigma \end{aligned}$$

The Hankel matrix in these two monomial sets  $A_1$  and  $A_2$  is defined by  $H_\sigma^{d_1, d_2} = [\sigma_{(\alpha+\beta)}]_{\substack{|\alpha| \leq d_1 \\ |\beta| \leq d_2}}$ .

Computing the singular value decomposition of  $H_\sigma^{d_1, d_2}$ , we obtain

$$H_\sigma^{d_1, d_2} = USV^\top$$

where  $S$  is the diagonal matrix of all singular values of  $H_\sigma^{d_1, d_2}$  arranged in a decreasing order,  $U$  is a unitary matrix whose columns are the left singular vectors of  $H_\sigma^{d_1, d_2}$ ,  $V$  is a unitary matrix whose columns are the right singular vectors of  $H_\sigma^{d_1, d_2}$ . We denote by  $U^H$  the hermitian transpose of  $U$  and  $\bar{V}$  the conjugate of  $V$ .

Let  $u_i = [u_{\alpha, i}]_{\alpha \in A_1}$  and  $v_j = [v_{\beta, j}]_{\beta \in A_2}$  be respectively the  $i^{\text{th}}$  and  $j^{\text{th}}$  columns of  $U^H$  and  $\bar{V}$ . They are vectors respectively in  $\mathbb{C}^{N_1}$  and  $\mathbb{C}^{N_2}$ . We denote by  $u_i(\mathbf{x}) = u_i^\top A_1 = \sum_{|\alpha| \leq d_1} u_{\alpha, i} \mathbf{x}^\alpha$  and  $v_j(\mathbf{x}) = v_j^\top A_2 = \sum_{|\beta| \leq d_2} v_{\beta, j} \mathbf{x}^\beta$  the corresponding polynomials. The bases formed by these first  $r$  polynomials are denoted  $\mathcal{U}_r^H := (u_i(\mathbf{x}))_{i=1, \dots, r}$  and  $\bar{\mathcal{V}}_r := (v_j(\mathbf{x}))_{j=1, \dots, r}$ . We will also denote by  $U_r^H$  (resp.  $\bar{V}_r$ ) the corresponding coefficient matrix, formed by the first rows (resp. columns) of  $U^H$  (resp.  $\bar{V}$ ). We denote by  $S_r$  the diagonal matrix of the first  $r$  rows and columns of  $S$ , formed by the first  $r$  singular values.

**Proposition 2.3.13.** *Let  $\sigma = \sum_{i=1}^{r'} \omega_i(\mathbf{y}) \mathbf{e}_{\xi_i}$  with  $\omega_i \in \mathbb{C}[\mathbf{y}]$ ,  $\xi_i \in \mathbb{C}^n$  and  $\sum_{i=1}^{r'} \mu(\omega_i) = r$ . If  $\text{rank } H_\sigma^{d_1, d_2} = r$ , then the sets of polynomials  $\mathcal{U}_r^H$  and  $\bar{\mathcal{V}}_r$  are bases of  $\mathcal{A}_\sigma$ . The matrix  $M_{x_i}^{\bar{\mathcal{V}}_r}$  associated to the multiplication operator by  $x_i$  in the basis  $\bar{\mathcal{V}}_r$  of  $\mathcal{A}_\sigma$  is  $M_{x_i}^{\bar{\mathcal{V}}_r} = S_r^{-1} U_r^H H_{x_i \star \sigma}^{d_1, d_2} \bar{V}_r$   $i = 1, \dots, n$ .*

*Proof.* The  $(i, j)$  entry of the matrix  $H_\sigma^{\mathcal{U}_r^H, \bar{\mathcal{V}}_r}$  of the truncated Hankel operator

of  $\sigma$  with respect to  $\mathcal{U}_r^H$  and  $\overline{\mathcal{V}}_r$  is equal to:

$$\begin{aligned}
(H_\sigma^{\mathcal{U}_r^H, \overline{\mathcal{V}}_r})_{[i,j]} &= \langle \sigma | u_i(\mathbf{x}) v_j(\mathbf{x}) \rangle \\
&= \langle \sigma | \left( \sum_{|\alpha| \leq d_1} u_{\alpha,i} \mathbf{x}^\alpha \right) \left( \sum_{|\beta| \leq d_2} v_{\beta,j} \mathbf{x}^\beta \right) \rangle = \sum_{|\alpha| \leq d_1} u_{\alpha,i} \sum_{|\beta| \leq d_2} \langle \sigma | \mathbf{x}^\alpha \mathbf{x}^\beta \rangle v_{\beta,j} \\
&= [U_r^H H_\sigma^{d_1, d_2} \overline{V}_r]_{[i,j]}.
\end{aligned} \tag{2.6}$$

Using the SVD decomposition of  $H_\sigma^{d_1, d_2}$ , we have

$$H_\sigma^{\mathcal{U}_r^H, \overline{\mathcal{V}}_r} = U_r^H H_\sigma^{d_1, d_2} \overline{V}_r = U_r^H U S V^T \overline{V}_r = S_r,$$

since  $U^H U = \text{Id}_{N_1}$ ,  $V^T \overline{V} = \text{Id}_{N_2}$ . As  $r = \text{rank } H_\sigma^{d_1, d_2}$ ,  $S_r$  is invertible and by Lemma 2.3.12,  $\mathcal{U}_r^H$  and  $\overline{\mathcal{V}}_r$  are linearly independent in  $\mathcal{A}_\sigma$ , which is a vector space of dimension  $r$ . Thus they are bases of  $\mathcal{A}_\sigma$ .

Let  $H_{x_i \star \sigma}^{\mathcal{U}_r^H, \overline{\mathcal{V}}_r}$  be the matrix of the truncated Hankel operator of  $x_i \star \sigma$  on the two bases  $U_r^H$  and  $\overline{V}_r$ . A similar computation yields  $H_{x_i \star \sigma}^{\mathcal{U}_r^H, \overline{\mathcal{V}}_r} = U_r^H H_{x_i \star \sigma}^{d_1, d_2} \overline{V}_r$ , where  $H_{x_i \star \sigma}^{d_1, d_2}$  is the matrix of the truncated Hankel operator of  $x_i \star \sigma$  in the bases  $A_1$  and  $A_2$  for all  $i = 1, \dots, n$ . Since  $S_r$  is an invertible matrix, by Proposition 2.3.11 we obtain  $M_{\mathbf{x}_i}^{\overline{\mathcal{V}}_r} = (H_\sigma^{\mathcal{U}_r^H, \overline{\mathcal{V}}_r})^{-1} H_{x_i \star \sigma}^{\mathcal{U}_r^H, \overline{\mathcal{V}}_r} = S_r^{-1} U_r^H H_{x_i \star \sigma}^{d_1, d_2} \overline{V}_r$ .  $\square$

By this proposition  $\mathcal{U}_r^H$  and  $\overline{\mathcal{V}}_r$  are bases of  $\mathcal{A}_\sigma$ . By Proposition 2.3.11, the eigenvalues of  $M_{\mathbf{x}_i}^{\overline{\mathcal{V}}_r}$  are the  $i^{\text{th}}$  coordinates  $x_i(\xi_j) = \xi_{j,i}$  of the roots  $\xi_j$  for  $i = 1, \dots, n, j = 1, \dots, r$ .

### Computation of weights

The weight  $\omega_i, i = 1, \dots, r$  of the decomposition of  $\sigma$  can be easily computed using the eigenvectors of all  $M_{\mathbf{x}_j}^{\overline{\mathcal{V}}_r}, j = 1, \dots, n$  as follows.

**Proposition 2.3.14.** *Let  $\sigma = \sum_{i=1}^r \omega_i \mathbf{e}_{\xi_i}$  with  $\omega_i \in \mathbb{C} \setminus \{0\}$ ,  $\xi_i = (\xi_{i,1}, \dots, \xi_{i,n}) \in \mathbb{C}^n$  distinct and let  $M_{\mathbf{x}_j}^{\overline{\mathcal{V}}_r}$  be the matrix of multiplication by  $x_j$  in the basis  $\overline{\mathcal{V}}_r$ . Let  $\mathbf{v}_i$  be a common eigenvector of  $M_{\mathbf{x}_j}^{\overline{\mathcal{V}}_r}, j = 1, \dots, n$  for the eigenvalues  $\xi_{i,j}$ . Then the weight of  $\mathbf{e}_{\xi_i}$  in the decomposition of  $\sigma$  is*

$$\omega_i = \frac{[1]^\top H_\sigma^{d_1, d_2} \overline{V}_r \mathbf{v}_i}{[\xi_i^\alpha]_{\alpha \in A_2}^\top \overline{V}_r \mathbf{v}_i}. \tag{2.7}$$



*Proof.* According to Proposition 2.2.8, the eigenvectors of the multiplication operator  $\mathcal{M}_{x_i}$  are, up to a scalar, the interpolation polynomials  $\mathbf{u}_i(\mathbf{x})$  at the roots. Let  $\mathbf{u}_{\xi_i}$  be the coefficient vector associated to  $\mathbf{u}_{\xi_i}(\mathbf{x})$  in the basis  $\overline{\mathcal{V}}_r$  of  $\mathcal{A}_\sigma$ . Let  $\mathbf{v}_i = \lambda \mathbf{u}_{\xi_i}$  be the eigenvector of  $M_{x_i}^{\overline{\mathcal{V}}_r}$  associated to the eigenvalue  $\xi_{j,i}$  for  $j = 1, \dots, r, i = 1, \dots, n$  such that  $\mathbf{v}_i(\mathbf{x}) = A_2^\top \tilde{\mathbf{v}}_i = \sum_{|\beta| \leq d_2} \tilde{\mathbf{v}}_{i\beta} \mathbf{x}^\beta$  where  $\tilde{\mathbf{v}}_i = \overline{\mathcal{V}}_r \mathbf{v}_i$ . Applying the series on all the idempotents, we obtain

$$\langle \sigma \mid \mathbf{u}_{\xi_i}(\mathbf{x}) \rangle = \left\langle \sum_{j=1}^r \omega_j \mathbf{e}_{\xi_j}(\mathbf{y}) \mid \mathbf{u}_{\xi_i}(\mathbf{x}) \right\rangle = \omega_i \mathbf{u}_{\xi_i}(\xi_i) = \omega_i.$$

Therefore, we have  $\omega_i = \frac{\langle \sigma \mid \lambda \mathbf{u}_{\xi_i}(\mathbf{x}) \rangle}{\lambda} = \frac{\langle \sigma \mid \mathbf{v}_i(\mathbf{x}) \rangle}{\lambda} = \frac{\langle \sigma \mid \mathbf{v}_i(\mathbf{x}) \rangle}{\mathbf{v}_i(\xi_i)}$  because of  $\mathbf{v}_i(\xi_i) = (\lambda \mathbf{u}_{\xi_i})(\xi_i) = \lambda$ . Then

$$\langle \sigma \mid \mathbf{v}_i(\mathbf{x}) \rangle = [1]^\top H_\sigma^{d_1, d_2} \tilde{\mathbf{v}}_i = [1]^\top H_\sigma^{d_1, d_2} \overline{\mathcal{V}}_r \mathbf{v}_i,$$

where  $[1]$  is the vector of coefficients of the polynomial 1 in the monomial basis  $A_1 = (\mathbf{x}^\alpha)_{|\alpha| \leq d_1}$  and

$$\mathbf{v}_i(\xi_i) = [\xi_i^\alpha]_{\alpha \in A_2}^\top \tilde{\mathbf{v}}_i = [\xi_i^\alpha]_{\alpha \in A_2}^\top \overline{\mathcal{V}}_r \mathbf{v}_i.$$

We deduce that  $\omega_i = \frac{[1]^\top H_\sigma^{d_1, d_2} \overline{\mathcal{V}}_r \mathbf{v}_i}{[\xi_i^\alpha]_{\alpha \in A_2}^\top \overline{\mathcal{V}}_r \mathbf{v}_i}$ . □

### Algorithm

We describe now the algorithm to recover the sum

$$\sigma(\mathbf{y}) = \sum_{j=1}^r \omega_j \mathbf{e}_{\xi_j}(\mathbf{y}),$$

$\omega_j \in \mathbb{C} \setminus \{0\}, \xi_j \in \mathbb{C}^n$ , from the first coefficients  $(\sigma_\alpha)_{|\alpha| \leq d}$  of the formal power series  $\sigma(\mathbf{y}) = \sum_{\alpha} \sigma_{\alpha} \frac{\mathbf{y}^{\alpha}}{\alpha!}$ .

---

**Algorithm 1** : Decomposition of polynomial-exponential series with constant weights

---

**Input:** the moments  $\sigma_{\alpha}$  of  $\sigma$  for  $|\alpha| \leq d$ .

Let  $d_1$  and  $d_2$  be positive integers such that  $d_1 + d_2 + 1 = d$ , for example  $d_1 := \lceil \frac{d-1}{2} \rceil$  and  $d_2 := \lfloor \frac{d-1}{2} \rfloor$ .

1. Compute the Hankel matrix  $H_{\sigma}^{d_1, d_2} = [\sigma_{(\alpha+\beta)}]_{\substack{|\alpha| \leq d_1 \\ |\beta| \leq d_2}}$  of  $\sigma$  in for the monomial sets  $A_1 = (\mathbf{x}^{\alpha})_{|\alpha| \leq d_1}$  and  $A_2 = (\mathbf{x}^{\beta})_{|\beta| \leq d_2}$ .
2. Compute the singular value decomposition of  $H_{\sigma}^{d_1, d_2} = USV^{\top}$  with singular values  $s_1 \geq s_2 \geq \dots \geq s_m \geq 0$ .
3. Determine its numerical rank, that is, the largest integer  $r$  such that  $\frac{s_r}{s_1} \geq \epsilon$ .
4. Form the matrices  $M_{x_i}^{\bar{V}_r} = S_r^{-1} U_r^{\text{H}} H_{x_i \star \sigma}^{d_1, d_2} \bar{V}_r, i = 1, \dots, n$ , where  $H_{x_i \star \sigma}^{d_1, d_2}$  is the Hankel matrix associated to  $x_i \star \sigma$ .
5. Compute the eigenvectors  $\mathbf{v}_j$  of  $\sum_i^n l_i M_{x_i}$  for a random choice of  $l_i$  in  $[-1, 1], i = 1, \dots, n$  and for each  $j = 1, \dots, r$  do the following:
  - Compute  $\xi_{j,i}$  such that  $M_i \mathbf{v}_j = \xi_{j,i} \mathbf{v}_j$  for  $i = 1, \dots, n$  and deduce the point  $\xi_j := (\xi_{j,1}, \dots, \xi_{j,n})$ .
  - Compute  $\omega_j = \frac{\langle \sigma | \mathbf{v}_j(\mathbf{x}) \rangle}{\mathbf{v}_j(\xi_j)} = \frac{[1]^{\top} H_{\sigma}^{d_1, d_2} \bar{V}_r \mathbf{v}_j}{[\xi_j^{\alpha}]_{\alpha \in A_2}^{\top} \bar{V}_r \mathbf{v}_j}$  where  $[1]$  is the coefficient vector of 1 in the basis  $A_1$ .

**Output:**  $r \in \mathbb{N}, \omega_j \in \mathbb{C} \setminus (0), \xi_j \in \mathbb{C}^n, j=1, \dots, r$  such that  $\sigma(\mathbf{y}) = \sum_{j=1}^r \omega_j \mathbf{e}_{\xi_j}(\mathbf{y})$  up to degree  $d$ .

---

## 2.4 Experimentation

In this section, we present numerical experimentations for the decomposition of  $\sigma = \sum_{\alpha \in \mathbb{N}} \sigma_{\alpha} \frac{\mathbf{y}^{\alpha}}{\alpha!}$  from its moments  $\sigma_{\alpha}$ . For a given number of variables  $n$

and a fixed degree  $d$ , we compute the coefficients  $\sigma_\alpha = \sigma(\mathbf{x}^\alpha) = \sum_{i=1}^r \omega_i \xi_i^\alpha$  such that  $|\sigma_\alpha| \leq d$  where  $\omega_j \in \mathbb{C} \setminus (0)$  and  $\xi_i = (\xi_{i,1}, \dots, \xi_{i,n}), i = 1, \dots, r$  have random uniform distributions such that  $0.5M \leq |\xi_{i,j}| \leq 1.5M$ ,  $-\pi \leq \arg(\xi_{i,j}) \leq \pi$ ,  $0.5 \leq |\omega_i| \leq 1$  and  $-\pi \leq \arg(\omega_i) \leq \pi$ , for  $M \geq 1$ . To analyse the numerical behaviour of our method, we compare the results with the known frequencies and weights used to compute the moments of  $\sigma$ .

We use Maple 16 to implement the algorithms. The arithmetic operations are performed on complex numbers, with a numerical precision fixed to  $Digits = 15$ .

### 2.4.1 Reconstruction using Fourier Coefficients

Let  $T = (T_1, \dots, T_n) \in \mathbb{R}_+^n$  and  $\Omega = \prod_{i=1}^n [-\pi T_i, \pi T_i] \in \mathbb{R}^n$ , one application of decomposition algorithm is the reconstruction of the measure  $\mu$  as a weighted sum of Dirac measures with support in  $\Omega$  such that:

$$\mu = \sum_{i=1}^{r'} \sum_{\alpha \in A_i \subset \mathbb{N}^n} \omega_{i,\alpha} \delta_{\xi_i}^\alpha$$

using its Fourier coefficients  $\sigma = (\sigma_\gamma)_{\gamma \in \mathbb{Z}^n}$ . For  $\gamma = (\gamma_1, \dots, \gamma_n) \in \mathbb{Z}^n$ , the  $\gamma^{\text{th}}$  Fourier coefficients of the measure  $\mu$  is  $\sigma_\gamma = \frac{1}{\prod_{j=1}^n T_j} \int \mu(\mathbf{x}) e^{-2\pi i \sum_{j=1}^n \frac{\gamma_j x_j}{T_j}}$ .

### 2.4.2 Numerical example

In this section, we present a detailed example of decomposition of formal power series of degree  $d = 4$  into weighted sum of 2 evaluations such that the weights are non-zero and the points belongs to  $\mathbb{C}^2$ .

Let  $\sigma(x_1, x_2)$  a formal power series of the form:

$$\begin{aligned} \sigma(x_1, x_2) = & 0.6633 + 0.7783I + (1.1840 - 1.5203I)x_1^2 x_2 + (1.5962 - 5.1489I)x_1^4 + \\ & (1.3504 - 1.1120I)x_1 x_2^3 + (-0.6461 - 0.0905I)x_1 x_2 + (-0.0175 - 1.1328I)x_1 + \\ & (0.7029 - 1.6456I)x_2^2 + (0.3992 + 2.0175I)x_1 x_2^2 + (-0.1485 + 1.1147I)x_2 + (-0.9206 + \\ & 2.1364I)x_1^3 x_2 + (-1.0318 - 0.1351I)x_2^3 + (-1.8318 - 0.9192I)x_1^2 x_2^2 + (0.2513 + \\ & 2.5979I)x_1^3 + (-0.9557 + 1.4759I)x_2^4 + (-0.6061 + 0.1348I)x_1^2. \end{aligned}$$

We suppose that  $d_1 = 1$  and  $d_2 = 2$  so that the truncated Hankel matrix  $H_\sigma^{d_1, d_2}$  associated to  $\sigma$  in the basis  $A_1 = \{1, x_1, x_2\}$  and  $A_2 = \{1, x_1, x_2, x_1^2, x_1 x_2, x_2^2\}$  is equal to:

$$H_{\sigma}^{d_1, d_2} = \begin{bmatrix} 0.2513 + 2.5979I & 1.1840 - 1.5203I & -0.6061 + 0.1348I \\ 1.1840 - 1.5203I & 0.3992 + 2.0175I & -0.6461 - 0.0905I \\ 0.3992 + 2.0175I & -1.0318 - 0.1351I & 0.7029 - 1.6456I \\ -0.6061 + 0.1348I & -0.6461 - 0.0905I & -0.0175 - 1.1328I \\ -0.6461 - 0.0905I & 0.7029 - 1.6456I & -0.1485 + 1.1147I \\ -0.0175 - 1.1328I & -0.1485 + 1.1147I & 0.6633 + 0.7783I \end{bmatrix}$$

The truncated Hankel matrices  $H_{x_i \star \sigma}^{d_1, d_2}$  associated to  $x_i \star \sigma$  in the basis  $A_1 = \{1, x_1, x_2\}$  and  $A_2 = \{1, x_1, x_2, x_1^2, x_1x_2, x_2^2\}$  for  $i = 1, 2$  are equal to:

$$H_{x_1 \star \sigma}^{d_1, d_2} = \begin{bmatrix} 1.5962 - 5.1489I & -0.9206 + 2.1364I & 0.2513 + 2.5979I \\ -0.9206 + 2.1364I & -1.8318 - 0.9192I & 1.1840 - 1.5203I \\ -1.8318 - 0.9192I & 1.3504 - 1.1120I & 0.3992 + 2.0175I \\ 0.2513 + 2.5979I & 1.1840 - 1.5203I & -0.6061 + 0.1348I \\ 1.1840 - 1.5203I & 0.3992 + 2.0175I & -0.6461 - 0.0905I \\ -0.6061 + 0.1348I & -0.6461 - 0.0905I & -0.0175 - 1.1328I \end{bmatrix}$$

$$H_{x_2 \star \sigma}^{d_1, d_2} = \begin{bmatrix} -0.9206 + 2.1364I & -1.8318 - 0.9192I & 1.1840 - 1.5203I \\ -1.8318 - 0.9192I & 1.3504 - 1.1120I & 0.3992 + 2.0175I \\ 1.3504 - 1.1120I & -0.9557 + 1.4759I & -1.0318 - 0.1351I \\ 1.1840 - 1.5203I & 0.3992 + 2.0175I & -0.6461 - 0.0905I \\ 0.3992 + 2.0175I & -1.0318 - 0.1351I & 0.7029 - 1.6456I \\ -0.6461 - 0.0905I & 0.7029 - 1.6456I & -0.1485 + 1.1147I \end{bmatrix}$$

The multiplication matrices by  $x_1$  and  $x_2$  in the orthogonal basis are respectively equal to:

$$M_{x_1} = \begin{bmatrix} -1.1271 + 0.5514I & 0.50211 + 0.4100I & 0.1024 - 0.0712I \\ -0.4788 + 1.0304I & -0.6477 - 0.5095I & -0.1769 - 0.0549I \\ -1.5150 - 1.2786I & 1.6611 + 0.1639I & -0.6651 + 0.2514I \end{bmatrix}$$

$$M_{x_2} = \begin{bmatrix} 0.0475 - 0.9528I & -0.4267 - 0.2658I & -0.0999 + 0.0546I \\ 0.2043 - 0.9326I & -0.5251 + 0.2362I & -0.0451 - 0.0528I \\ 0.8754 + 1.7504I & 0.5753 + 0.7663I & -0.5222 + 0.3172I \end{bmatrix}$$

The frequencies  $\xi_1, \xi_2, \xi_3$  in  $\mathbb{C}^2$  are respectively equal to:

$$\xi_1 = [-0.8091 - 1.2527I, -0.1258 + 0.4860I]$$

$$\xi_2 = [-0.8870 + 1.0111I, -0.1398 - 1.2992I]$$

$$\xi_3 = [-0.7437 + 0.5348I, -0.7341 + 0.4139I]$$

Their associated weights are:

$$\omega_1 = 0.3325 + 0.5946I, \omega_2 = -0.4439 + 0.4465I, \omega_3 = 0.7747 - 0.2628I.$$

### 2.4.3 Numerical behavior against perturbation

We apply random perturbations on the moments of the form  $\sigma_\alpha + \epsilon(p_\alpha + i q_\alpha)$  where  $p_\alpha$  and  $q_\alpha$  are two random numbers in  $[-1, 1]$  with a uniform distribution, and  $\epsilon = 10^{-e}$  where  $e$  is a fixed positive integer.

To measure the consistency of our algorithm, we compute the maximum error between the input frequencies  $\xi_i$  and the output frequencies  $\xi'_i$ , and between the input weights  $\omega_i$  and the output weights  $\omega'_i$ :

$$\text{err} = \max(\text{err}(\xi_i, \xi'_i), \text{err}(\omega_i, \omega'_i)) \text{ where } \text{err}(\omega_i, \omega'_i) = \max_{1 \leq i \leq r} |\omega_i - \omega'_i| \text{ and } \text{err}(\xi_i, \xi'_i) = \max_{1 \leq i \leq r} \|\xi_i - \xi'_i\| \quad (2.8)$$

In each computation, we compute the average of the maximum errors resulting from 10 tests.

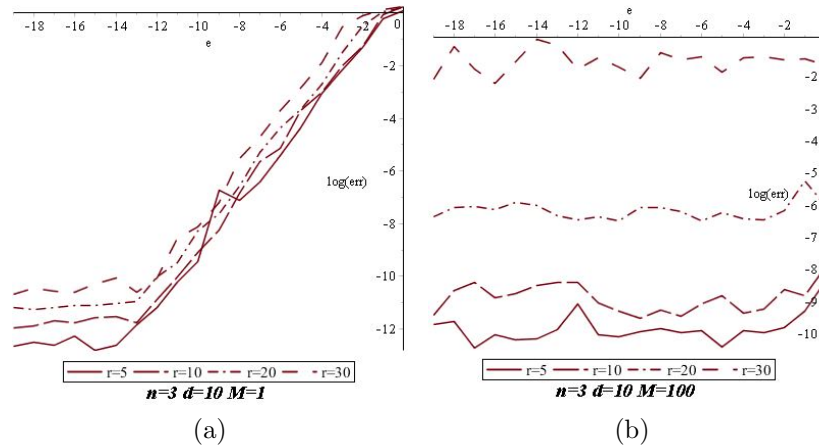


Figure 2.1: The influence of the amplitude of the frequencies on the maximum error.

In Figures 2.1a and 2.1b, we study the evolution of the error in terms of the perturbation  $\epsilon = 10^{(-e)}$ , for a fixed degree  $d = 10$ , a number of variables  $n = 3$ , different ranks  $r = 5, 10, 20, 30$  and for two different amplitudes of the frequencies  $M = 1$  and  $M = 100$ .

In Figure 2.1a for  $M = 1$ , the lower error is for the lower rank  $r = 5$ . Between  $\epsilon \approx 10^{-12}$  and  $\epsilon = 1$ , the error  $\text{err}$  increases in terms of the perturbation as  $\text{err} = \exp(te)$  for some slope  $t \approx 1$ . The slope  $t$  remains approximately constant but the error increases slightly with the rank  $r$ . Before  $\epsilon = 10^{-13}$ , it

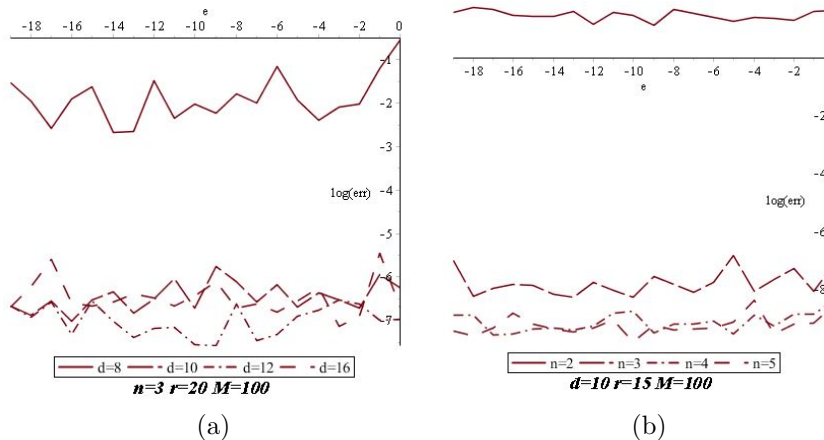


Figure 2.2: The influence of the degree and dimension on the maximum error.

is approximately constant (approximately  $10^{-12}$  for  $r = 5$ ). This is due to the fact, in this range, the perturbation is lower than the numerical precision.

In Figure 2.1b for  $M = 100$ , the lower error is also for the lower rank. The error has almost a constant value when  $e$  varies. It is bigger than for  $M = 1$  for small perturbations. For  $r = 5, 10$  the error slightly increases between  $e = -2$  and  $e = 0$ , with a similar slope. This figure clearly shows that the error degrades significantly from  $M = 1$  to  $M = 100$  and that the degradation increases rapidly with the rank  $r$ .

In Figure 2.2a, we fix the number of variables  $n = 3$ , the rank  $r = 20$ ,  $M = 100$  and we change the degree  $d$  which induces a change in the dimensions of the Hankel matrices. For  $e \in [-19, 0]$ , the error decreases when we increase the degree from  $d = 8$  to  $d = 10$ . It is slightly lower when  $d = 12$  than when  $d = 10$ , and error is similar for  $d = 10$  and  $d = 16$ . This increase of the precision with the degree can be related to ratio of number of moments by the number of values to recover in the decomposition.

In Figure 2.2b, we fix the degree  $d = 10$ , the rank  $r = 15$ ,  $M = 100$  and we change the number of variables  $n = 2, 3, 4, 5$ . The dimension of the matrices increases polynomially with  $n$ . We observe that the error decreases quickly with  $n$ . It shows that the precision improves significantly with the dimension.

### 2.4.4 Rescaling

As we have seen in Figures 2.1a and 2.1b, the error increases significantly with the amplitude  $M$ . To remedy this issue, we present a rescaling technique and its numerical impact. It's done like this:

- For a chosen non-zero constant  $\lambda$ , we transform the input moments of the series as follows:

$$\sigma(\mathbf{y}) := \sum_{\alpha \in N^n} \sigma_\alpha \frac{\mathbf{y}^\alpha}{\alpha!} \longrightarrow \tilde{\sigma}(\mathbf{y}) := \sigma(\lambda \mathbf{y}) = \sum_{\alpha \in N^n} \lambda^{|\alpha|} \sigma_\alpha \frac{\mathbf{y}^\alpha}{\alpha!},$$

which corresponds to the scaling on the frequencies  $e_\xi(\lambda \mathbf{y}) = e_{\lambda \xi}(\mathbf{y})$ .

- We compute decomposition of  $\tilde{\sigma}(\mathbf{y}) = \sigma(\lambda \mathbf{y})$  from the moments  $\tilde{\sigma}_\alpha = \lambda^{|\alpha|} \sigma_\alpha$ .
- We apply the inverse scaling on the computed frequencies  $\tilde{\xi}_i$  which gives  $\xi_i = \frac{\tilde{\xi}_i}{\lambda} = \left( \frac{\tilde{\xi}_{i,1}}{\lambda}, \dots, \frac{\tilde{\xi}_{i,n}}{\lambda} \right)$ .

To determine the scaling factor  $\lambda$ , we use  $\lambda := \frac{1}{m}$  where  $m = \frac{\max_{|\alpha|=d} |\sigma_\alpha|}{\max_{|\alpha|=d-1} |\sigma_\alpha|}$ . This is justified as follows: If  $|\omega_j| \leq 1, j = 1, \dots, r$ , then  $|\sigma_\alpha| = \left| \sum_{j=1}^r \omega_j \xi_j^{|\alpha|} \right| \simeq M^d$  for  $|\alpha| = d$  big and for  $M$  is the highest modulus of frequencies. Similarly  $|\sigma_{\alpha'}| \simeq M^{d-1}$  for  $|\alpha'| = d - 1$ . Then we have  $m = \frac{\max_{|\alpha|=d} |\sigma_\alpha|}{\max_{|\alpha|=d-1} |\sigma_\alpha|} \approx M$ .

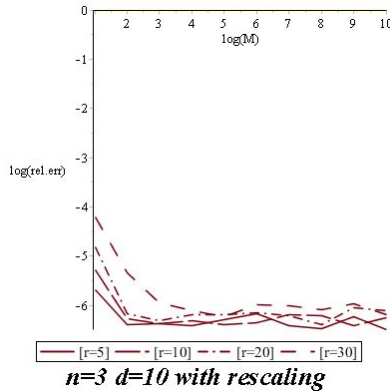


Figure 2.3: The rescaling influence

To study the numerical influence of the rescaling, we compute the maximum relative error between the input frequencies  $\xi_i$  and the output frequencies  $\tilde{\xi}_i$ , and the maximum error between the input weights  $\omega_i$  and the output

weights  $\tilde{\omega}_i$ , and we take their maximum:

$$\text{rel.err} = \max(\text{rel.err}(\xi_i, \tilde{\xi}_i), \text{err}(\omega_i, \tilde{\omega}_i)) \quad (2.9)$$

where  $\text{err}(\omega_i, \tilde{\omega}_i) = \max_{1 \leq i \leq r} |\omega_i - \tilde{\omega}_i|$  and  $\text{rel.err}(\xi_i, \tilde{\xi}_i) = \max_{1 \leq i \leq r} \frac{\|\xi_i - \tilde{\xi}_i\|_2}{\|\xi_i\|_2}$ .

In Figure 2.3, we see the influence of the rescaling on the maximum relative error. The perturbation on the moments is of the order  $\varepsilon = 10^{-6}$ . Each curve for  $r = 5, 10, 20, 30$ , has almost a constant evolution with the increasing values of  $M$  between  $10^2$  and  $10^{10}$ . The maximum relative error is lower when  $M = 100$  than when  $M = 1$  which is confirmed with the results shown in Figures 2.1a and 2.1b. When we increase  $r$  the maximum relative error decreases slightly.

In conclusion, the rescaling has an important influence on the computation of the maximum relative error when the modulus  $M$  of points is quite big.

The scaling of moments by some computed factor  $\lambda$  also enhances the computation of the numerical rank  $r$  and leads to a better decomposition as explained in Section 2.4.4.

## 2.4.5 Numerical Rank

To compute the number  $r$  of terms in the decomposition of  $\sigma$ , we arrange the diagonal entries in the decreasing order  $s_1 \geq s_2 \geq \dots \geq s_r > s_{r+1} \geq \dots \geq s_{N_2}$  and we determine the numerical rank of  $H_\sigma^{d_1, d_2}$  by fixing the largest integer  $r$  such that  $s_r/s_1 \geq \epsilon$ .

It is known that the ill-conditioning of the Hankel matrix associated to Prony's method is in the origin of a numerical instability with respect to perturbed measurements

$$\tilde{\sigma}_{\alpha+\beta} = \sigma_{\alpha+\beta} + \varepsilon_{\alpha+\beta} \quad |\alpha + \beta| \leq d.$$

In our algorithm the computation of the numerical rank can be affected by this instability. We can explain this instability, using a reasoning close to [Sau17], as follows.

We denote by  $s_j$  (resp.  $\tilde{s}_j$ ) the  $j^{\text{th}}$  largest singular value of  $H := H_\sigma^{d_1, d_2}$  (resp.  $\tilde{H} := H_{\tilde{\sigma}}^{d_1, d_2}$ ). The perturbation result for singular values satisfies the estimate (see [Gol96])

$$|s_j - \tilde{s}_j| \leq s_1(\varepsilon) = \|\varepsilon\|_2.$$



Then, as long as the perturbation is small relative to the conditioning of the problem, that is

$$\|\epsilon\|_2 \leq \frac{1}{2}s_r \quad \text{provided that } r = \text{rank}(H),$$

then  $\|s_j - \tilde{s}_j\| \leq \frac{1}{2}s_r \quad \forall j$  and therefore  $\tilde{s}_r \geq \frac{1}{2}s_r$  and  $\tilde{s}_{r+1} \leq \frac{1}{2}s_r$ . Hence by taking  $\epsilon \leq \frac{1}{2}s_r$  as a threshold level we will be sure that the rank is calculated correctly.

But the problem may be badly ill-conditioned and then such a level will not be reasonable. In fact

$$H = (\sigma_{\alpha+\beta})_{\substack{|\alpha| \leq d_1 \\ |\beta| \leq d_2}} = \left( \sum_{i=1}^r \omega_i \xi_i^{\alpha+\beta} \right)_{\substack{|\alpha| \leq d_1 \\ |\beta| \leq d_2}} = \sum_{i=1}^r \omega_i v_{i,d_1} v_{i,d_2}^T,$$

where  $v_{i,d_1} = (\xi_i^{|\alpha|})_{|\alpha| \leq d_1}$  (resp.  $v_{i,d_2} = (\xi_i^{|\beta|})_{|\beta| \leq d_2}$ ) is the  $i^{\text{th}}$  column of the Vandermonde matrix  $V_{d_1} = (\xi_i^\alpha)_{\substack{1 \leq i \leq r \\ |\alpha| \leq d_1}}$  (resp.  $V_{d_2} = (\xi_i^\beta)_{\substack{1 \leq i \leq r \\ |\beta| \leq d_2}}$ ). Then

$$H = \sum_{i=1}^r \omega_i V_{d_1} e_i e_i^T V_{d_2}^T = V_{d_1} \left( \sum_{i=1}^r \omega_i e_i e_i^T \right) V_{d_2}^T = V_{d_1} C V_{d_2}^T$$

where  $C = \text{diag}((\omega_i)_{1 \leq i \leq r})$  is the diagonal matrix with  $\omega_i$  on the diagonal.

Now, using the fact that

$$s_r(H) = \min_{\substack{\|x\|=1 \\ Hx \neq 0}} \|Hx\|_2 = \min_{\substack{\|x\|=1 \\ Hx \neq 0}} \|V_{d_1} C V_{d_2}^T x\|_2,$$

we remark that if  $V_{d_2}$  (resp.  $V_{d_1}$ ) is ill-conditioned then  $\|V_{d_2}x\|_2$  (resp.  $\|V_{d_1} C V_{d_2}^T x\|_2$ ) may be very small and  $s_r(H)$  is small as well. This situation can also be produced if  $\max_{1 \leq i \leq r} \omega_i$  is very small. In our numerical experiments, the  $\omega_i$  are chosen randomly in  $[0.5, 1]$  and then they don't seem to cause any numerical instability.

On the other hand, the  $\xi_i$  vary in such a way that their amplitude can be large, which can generate very ill-conditioned Vandermonde matrices. In fact, it is known (see [Pan16]), that for a nonsingular univariate Vandermonde matrix  $V = (a_i^j)_{0 \leq i, j \leq n-1}$ , where  $(a_i)_{0 \leq i \leq n-1}$  denotes a vector of  $n$  distinct knots, the condition number of  $V$  is exponential in  $n$  if  $\max_{0 \leq i \leq n-1} |a_i| > 1$  or in  $k$  if  $|a_i| < 1$  for at least  $k$  knots  $a_i$ . Therefore an  $n \times n$  Vandermonde matrix is badly

ill-conditioned unless all knots lie in or near the disc  $D(0, 1) = \{x : |x| \leq 1\}$  and unless they lie mostly on or near its boundary  $C(0, 1)$ .

In the multivariate case, it appears that the condition number of multivariate Vandermonde matrices has the same behavior as in the univariate case. That is, it is exponential in the highest degree of the entries.

According to the foregoing, when the amplitude  $M$  of the frequencies increases (even for moderate values of  $M$ ) the numerical rank calculated by truncating the singular values of  $H$  will be different from the exact rank of  $H$ . As shown in the following section, rescaling remedy the problem of frequencies  $\xi_i$  of high amplitude in order to obtain points with coordinates close to the unitary circle  $C(0, 1)$ .

# Chapter 3

## Tensor Decomposition Problem

In the following, we study the general class of multi symmetric tensor decomposition problem, which contains these two classes of symmetric and non symmetric tensors. We show the correlation between the dual of a tensor, formal power series and then the Hankel matrices associated to them. We use the singular value decomposition of Hankel matrices to compute the decomposition of a tensor of low rank. We exploit the properties of Artinian Gorenstein Algebra to find out some multiplication matrices which help to know the eigen-structure of points associated to linear forms and their weights. We slice variables into bunches of sub-variables and we adapt the description of Artinian Gorenstein Algebra to this case. We adapt the method of decomposition of Hankel matrices of low rank described in [HKM18] to a decomposition of multi linear tensors method which is based on the decomposition of a formal power series as a weighted sum of exponential described in [Mou16]. The computation of multiplication matrices depend on the dimension of tensor, and the number of given moments or coefficients. We describe the algorithm in 3 dimensional space and we give its numerical implementation using MAPLE. This description gives an idea about the constraints and difficulties of the problem in n dimensional space. We show a numerical example of the decomposition of a tensor of rank 3 with order one in each bunch of 3 variables in 3 dimensional space.

## 3.1 Symmetric and Multi-Symmetric Tensors

In this section, we give the definition of a multi-symmetric tensor as a multi-homogeneous polynomial with different positive degrees for each collection of variables. This tensor can also be defined as a multi symmetric array of coefficients. Given a multi symmetric array of coefficients, we can define a multi-homogeneous polynomial. We also give the definition of a symmetric tensor as a particular case of multi-symmetric tensor.

We recall the basic definitions of outer product of vectors and tensor product in Euclidean spaces. We show the relationship between the non symmetric tensor and the  $k$ -way array using the so called Segré map.

### 3.1.1 Symmetric and Multi-Symmetric tensors

In the following we recall the definition of a multi-symmetric tensor as a multi-homogeneous polynomial. We show that the non symmetric tensor and the  $d$ -th order symmetric tensor are particular cases of multi-symmetric tensor.

Let  $\mathbb{K}$  be an algebraically closed field of characteristic 0.

### 3.1.2 Symmetric Tensor

**Definition 3.1.1.** *Let  $E$  be a  $\mathbb{K}$  vector space, a symmetric tensor of  $\mathcal{S}^d(E)$  can be interpreted as a symmetric array of coefficients  $[T] = [t_{\alpha'}]_{|\alpha'|=d}$  such that each  $\alpha' = (\alpha'_j)_{0 \leq j \leq n} \in \mathbb{N}^{n+1}$ .*

For  $\alpha \in \mathbb{N}^n$  with  $|\alpha| \leq d$ , we denote  $\bar{\alpha} = (d - |\alpha|, \alpha_1, \dots, \alpha_n)$ . The symmetric tensor can be written as  $[T] = [t_{\bar{\alpha}}]_{\substack{|\bar{\alpha}|=d \\ \bar{\alpha} \in \mathbb{N}^{n+1}}}$ .

Such tensor is identified with the multi-homogeneous polynomial

$$T(\mathbf{x}) = \sum_{\substack{|\bar{\alpha}|=d \\ \bar{\alpha} \in \mathbb{N}^{n+1}}} t_{\bar{\alpha}} \mathbf{x}^{\bar{\alpha}}$$

where  $\mathbf{x} = (x_0, x_1, \dots, x_n)$

If we let  $x_0 = 1$  we get

$$\underline{T}(\underline{\mathbf{x}}) = \sum_{\substack{|\alpha| \leq d \\ \alpha \in \mathbb{N}^n}} t_\alpha \underline{\mathbf{x}}^\alpha$$

where  $\underline{\mathbf{x}} = (x_1, \dots, x_n)$  because of  $\underline{\mathbf{x}}^\alpha = \mathbf{x}^{\bar{\alpha}}$  when  $x_0 = 1$ .

We denote by  $R_d$  the space obtained by deshomogenisation of elements in  $\mathcal{S}^d(E)$  by setting  $x_0 = 1$  where  $R = \mathbb{K}[\underline{\mathbf{x}}]$  is the space of polynomials in the variables  $\underline{\mathbf{x}} = (x_1, \dots, x_n)$ .

### 3.1.3 Non symmetric Tensor

A non symmetric tensor is defined when  $|\bar{\alpha}_j| = 1$  for  $j = 1, \dots, k$ , then by abuse of notation we obtain  $\bar{\alpha}_j[i_j] = 1$  for some  $0 \leq i_j \leq n_j$  and 0 elsewhere.

Let  $E_j$  be  $n_j$ -dimensional  $\mathbb{K}$  vector spaces for  $j = 1, \dots, k$ . Given a basis  $e_j = \{e_{i_j, j}\}_{0 \leq i_j \leq n_j}$  of  $E_j$  for  $j = 1, \dots, k$ , tensor  $T$  in the basis  $\{e_{i_1, 1} \otimes e_{i_2, 2} \otimes \dots \otimes e_{i_k, k} \mid 0 \leq i_1 \leq n_1, 0 \leq i_2 \leq n_2, \dots, 0 \leq i_k \leq n_k\}$  is

$$T = \sum_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}} t_{i_1, i_2, \dots, i_k} e_{i_1, 1} \otimes e_{i_2, 2} \otimes \dots \otimes e_{i_k, k}$$

**Definition 3.1.2.** *The vector space of arrays of coefficients in the basis  $\{e_{i_1, 1} \otimes e_{i_2, 2} \otimes \dots \otimes e_{i_k, k} \mid 0 \leq i_1 \leq n_1, 0 \leq i_2 \leq n_2, \dots, 0 \leq i_k \leq n_k\}$  is defined as the set of elements of the form  $[A] = [a_{i_1, i_2, \dots, i_k}]_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}}$ .*

*The vector space is denoted by  $\mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$  where  $[n_i] = \{0, 1, \dots, n_i\}$  for  $i = 1, \dots, k$ .*

Using the universal property of the tensor product and because of the  $\dim(E_1 \otimes E_2 \otimes \dots \otimes E_k) = \dim(\mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]})$ , there exists an isomorphism  $\theta$  between the space of non symmetric tensors and the space of arrays of coefficients with respect to the basis  $\{e_{i_1, 1} \otimes e_{i_2, 2} \otimes \dots \otimes e_{i_k, k} \mid 0 \leq i_1 \leq n_1, 0 \leq i_2 \leq n_2, \dots, 0 \leq i_k \leq n_k\}$ .

$\theta$  is explicitly defined as:

$$\theta : E_1 \otimes E_2 \otimes \dots \otimes E_k \rightarrow \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$$

$$T = \sum_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}} t_{i_1, i_2, \dots, i_k} e_{i_1, 1} \otimes e_{i_2, 2} \otimes \dots \otimes e_{i_k, k} \mapsto [t_{i_1, i_2, \dots, i_k}]_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}}$$

So that the multi symmetric array associated to the non symmetric tensor is defined as  $[T] = [t_{i_1, i_2, \dots, i_k}]_{\substack{0 \leq i_j \leq n_j \\ 1 \leq j \leq k}} \in \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$ .

If  $\mathbb{K} = \mathbb{R}$  (resp.  $\mathbb{K} = \mathbb{C}$ ) then the array of coefficients is called real-valued (resp. complex-valued) array of coefficients.

**Definition 3.1.3.** *The outer product of  $k$  vectors  $\mathbf{u}_1 \in E_1, \mathbf{u}_2 \in E_2, \dots, \mathbf{u}_k \in E_k$ , is defined as an array of coefficients  $[u_{i_1, 1} u_{i_2, 2} \dots u_{i_k, k}]_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}}$  in the basis*

$e_1 \otimes e_2 \otimes \dots \otimes e_k$  where each  $u_{i_j, j}$  is the  $i_j^{\text{th}}$  element of the vector  $\mathbf{u}_j \in E_j$  in the basis  $e_j$  of  $E_j$ . It is denoted by  $\mathbf{u}_1 \otimes \mathbf{u}_2 \otimes \dots \otimes \mathbf{u}_k$ .

If  $k = 3$ , the outer product of  $\mathbf{u}$  and  $\mathbf{v}$  and  $\mathbf{w}$  defines an array  $[u_i v_j w_k]_{\substack{0 \leq i \leq n \\ 0 \leq j \leq m \\ 0 \leq k \leq l}}$ .

If  $k = 2$ , the outer product of  $\mathbf{u}$  and  $\mathbf{v}$  defines a matrix  $[u_i v_j]_{\substack{0 \leq i \leq n \\ 0 \leq j \leq m}}$ .

To understand the relationship between the non symmetric tensor and the array of coefficients, we recall the existence of the so called Segré  $\mathbb{K}$ -linear map which associates to each  $k$ -tuple of vectors an outer product:

$$\phi_k : E_1 \times E_2 \times \dots \times E_k \rightarrow \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$$

$$(\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_k) \mapsto \mathbf{u}_1 \otimes \mathbf{u}_2 \otimes \dots \otimes \mathbf{u}_k$$

Such a tensor can be identified with the multi-homogeneous polynomial  $T(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k) =$

$$\sum_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}} t_{i_1, i_2, \dots, i_k} x_{1, i_1} x_{2, i_2} \dots x_{k, i_k} \text{ because of } \mathbf{x}_j^{\bar{\alpha}_j} = x_{j, i_j} \text{ where } \mathbf{x}_j = (x_{j, 0}, x_{j, 1}, x_{j, n_j})$$

for some  $0 \leq i_j \leq n_j$  and for all  $1 \leq j \leq k$ .

$$\text{The dual of the tensor is } T^*(\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_k) = \sum_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}} t_{i_1, i_2, \dots, i_k} y_{1, i_1} y_{2, i_2} \dots y_{k, i_k}$$

because of  $\mathbf{y}_j^{\bar{\alpha}_j} = y_{j, i_j}$  for some  $0 \leq i_j \leq n_j$  and for all  $1 \leq j \leq k$ .

## Flattening a Tensor

In this section, we describe the flattening space of tensor product of  $k$  vector spaces  $E_1, E_2, \dots, E_k$  and we show that the  $d$ -minors of a tensor  $T$  are equal to the  $d$ -minors of its image  $T'$  in the flattening space using a linear map.

**Definition 3.1.4.** *The flattening space of  $E_1 \otimes E_2 \otimes \dots \otimes E_k$  is defined after partitioning the set  $\{1, \dots, k\}$  onto two sets of indices  $J_1 = \{p_1, \dots, p_s\}$  and  $J_2 = \{q_1, \dots, q_{k-s}\}$  and then associating to each one of them the tensor product  $E_{J_1} = E_{p_1} \otimes \dots \otimes E_{p_s}$  and  $E_{J_2} = E_{q_1} \otimes \dots \otimes E_{q_{k-s}}$  and finally computing the tensor product space of  $E_{J_1}$  and  $E_{J_2}$  which is denoted by  $E_{J_1} \otimes E_{J_2}$  and equal to  $(E_{p_1} \otimes \dots \otimes E_{p_s}) \otimes (E_{q_1} \otimes \dots \otimes E_{q_{k-s}})$ .*

**Definition 3.1.5.** *The linear map which associates to a tensor, its corresponding flattening tensor according to the partition  $J_1, J_2$  of  $\{1, \dots, k\}$  is defined as follows:*

$$\begin{aligned} f_{J_1, J_2} : E_1 \otimes E_2 \otimes \dots \otimes E_k &\rightarrow E_{J_1} \otimes E_{J_2} \\ T &\mapsto f_{J_1, J_2}(T) = T' \end{aligned}$$

If

$$[T] = [t_{i_1, i_2, \dots, i_k}]_{\substack{0 \leq i_j \leq n_j \\ 1 \leq j \leq k}} \in \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$$

then

$$\begin{aligned} [T'] &= [t'_{i_1, i_2, \dots, i_s, j_1, j_2, \dots, j_{k-s}}]_{\substack{0 \leq i_l \leq p_l \\ 1 \leq l \leq s \\ 0 \leq j_h \leq q_h \\ 1 \leq h \leq k-s}} \\ &= [t_{i_1, i_2, \dots, i_s}]_{\substack{0 \leq i_l \leq p_l \\ 1 \leq l \leq s}} \otimes [t_{j_1, j_2, \dots, j_{k-s}}]_{\substack{0 \leq j_h \leq q_h \\ 1 \leq h \leq k-s}} \\ &= [t_{i_1, i_2, \dots, i_s} t_{j_1, j_2, \dots, j_{k-s}}]_{\substack{0 \leq i_l \leq p_l \\ 1 \leq l \leq s \\ 0 \leq j_h \leq q_h \\ 1 \leq h \leq k-s}} \end{aligned}$$

For  $d \in \mathbb{N}^*$ , if  $[T]$  is the array of coefficients associated to the tensor  $T$  and  $[A_{J_1, J_2}]$  is the array of coefficients associated to the tensor  $T'$  then the  $d$ -minors of  $[A_{J_1, J_2}]$  are the  $d$ -minors of  $[T]$ .

## Tensor operations

In this subsection, we recall some basic operations on non symmetric tensors such as how to convert a tensor to a vector and vice versa. We define also the non symmetric tensor decomposition problem. Let  $T \in \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$  to be a non symmetric tensor.

**Definition 3.1.6.** *The mode  $m$ -fiber of  $T$  can be obtained by fixing all elements but one index  $i_m$  then the corresponding fiber is denoted by  $t_{i_1, i_2, \dots, i_{m-1}, :, i_{m+1}, \dots, i_k}$*

We denote by  $I := [n_1]$ ,  $J := [n_2]$  and  $K := [n_3]$  the three sets of indices used to represent the non symmetric tensor of order 3.

**Example 3.1.7.** *The 3 mode  $m$ -fibers of non symmetric tensor of dimension 3  $T \in \mathbb{K}^{I \times J \times K}$  are  $t_{:,j,k}$ ,  $t_{i,:,k}$  and  $t_{i,j,:}$ . They are respectively the collections of fibers parallel to the  $Ox$ ,  $Oy$  and  $Oz$ .*

**Definition 3.1.8.** *The slices of  $T$  can be obtained by fixing all elements but two indices  $i_t$  and  $i_s$  then the corresponding slice is denoted by  $t_{i_1, i_2, \dots, i_{t-1}, :, i_{t+1}, \dots, i_{s-1}, :, i_{s+1}, \dots, i_k}$*

**Example 3.1.9.** *The 3 slices of non symmetric tensor  $T \in \mathbb{K}^{I \times J \times K}$  are  $t_{i,:,:,}$ ,  $t_{:,j,:}$  and  $t_{:, :, k}$ . They are respectively the collections of slices parallel to the  $yOz$ ,  $xOz$  and  $xOy$ .*

To convert a tensor into a matrix, we introduce the mode  $m$ -flattening concept as follows:

**Definition 3.1.10.** *The mode  $m$ -flattening of a tensor  $T$  is obtained by concatenating the mode  $m$ -fibers into a matrix denoted by  $T_{[m]} := [t_{i_1, i_2, \dots, i_{m-1}, :, i_{m+1}, \dots, i_k}]_{\substack{i_j \in [n_j] \\ j \in [k] \\ j \neq m}}$*

*It is flattening with the two tensor space products  $E_{J_1}$  and  $E_{J_2}$  associated to the sets of indices  $J_1 = \{i_1, \dots, i_{m-1}\}$  and  $J_2 = \{i_{m+1}, \dots, i_k\}$ .*

**Definition 3.1.11.** *The  $m$ -rank tensor is the rank of mode  $m$ -flattening of a tensor for  $m = 1, \dots, k$ .*

The different  $m$ -ranks of a tensor are not necessarily the same.

**Example 3.1.12.** *The mode 1-flattening of a non symmetric tensor  $T$  is obtained by concatenating the mode 1-fibers into a matrix which is denoted*



by  $T_{[1]} := [t_{:,j,k}]_{\substack{j \in J \\ k \in K}} \in \mathbb{K}^{I \times J \times K}$ . Similarly the mode 2-flattening and mode 3-flattening are respectively  $T_{[2]} := [t_{i,:k}]_{\substack{i \in I \\ k \in K}} \in \mathbb{K}^{J \times I \times K}$  and  $T_{[3]} := [t_{i,j,:}]_{\substack{i \in I \\ j \in J}} \in \mathbb{K}^{K \times I \times J}$ .

**Definition 3.1.13.** *The mode  $m$ -product of a tensor  $[T] \in \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$  and a matrix  $A \in \mathbb{K}^{P \times n_m}$  denoted by  $[T] \times_m A$  is the  $k$ -th order tensor  $[V]$  of size  $[n_1] \times [n_2] \times \dots \times [n_{m-1}] \times P \times [n_{m+1}] \times \dots \times [n_k]$ . The element of the new tensor is  $[T \times_m A]_{i_1, i_2, \dots, i_{m-1}, j, i_{m+1}, \dots, i_k} = \sum_{i_m \in [n_m]} t_{i_1, i_2, \dots, i_{m-1}, i_m, i_{m+1}, \dots, i_k} a_{j, i_m}$ .*

By mode  $m$ -flattening we can get  $[V] = [T] \times_m A \iff [V]_{[m]} = [T]_{[m]} A$ . The mode  $m$ -product satisfies these two properties:

**Proposition 3.1.14.** *Given the tensor  $[T] \in \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$  and the matrices  $A \in \mathbb{R}^{[p_m] \times [n_m]}$  and  $B \in \mathbb{R}^{[p_{m'}] \times [n_{m'}]}$  so that  $[T] \times_{m'} A \times_m B = [T] \times_m B \times_{m'} A$ .*

**Proposition 3.1.15.** *Given the tensor  $[T] \in \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$  and the matrices  $A \in \mathbb{R}^{[p_m] \times [n_m]}$  and  $B \in \mathbb{R}^{[q_m] \times [p_m]}$  so that  $[T] \times_m A \times_m B = [T] \times_m (BA)$ .*

For the proofs see [DL97].

We deduce that:

**Proposition 3.1.16.** *If  $A$  and  $B$  are two tensors and  $X \in \mathbb{R}^{I \times r}$ ,  $Y \in \mathbb{R}^{J \times r}$  and  $Z \in \mathbb{R}^{K \times r_3}$  we have  $A = B \times_1 X \times_2 Y \times_3 Z$  if and only if  $a_{i,j,k} = \sum_{r_1, r_2, r_3} x_{i, r_1} y_{j, r_2} z_{k, r_3} b_{r_1, r_2, r_3}$ .*

To convert a matrix into a vector, we introduce the vectorization concept as follows:

**Definition 3.1.17.** *The vectorization of a matrix  $M = [l_1, l_2, \dots, l_n] \in \mathbb{K}^{m \times n}$  where each  $l_j$  is a column of  $M$  is obtained by concatenating the columns into*

*a vector denoted by  $vec(M) := \begin{bmatrix} l_1 \\ l_2 \\ \dots \\ l_n \end{bmatrix}$ .*

The vectorization of a tensor  $T$  is the vectorization of its mode 1-flattening denoted by  $vec(T)$ .

Now, we show how to convert a vector into a matrix or a vector into a tensor as follows:

**Definition 3.1.18.** Given a vector  $v \in \mathbb{K}^{I.I}$ , the matrix associated to  $v$  denoted by  $\text{unvec}(v) \in \mathbb{K}^{I \times I}$  is obtained by dividing  $v$  into  $I$  vectors such that each one of them is of dimension  $I$  and then we concatenate them as the columns of resulting matrix.

**Definition 3.1.19.** Given a vector  $v \in \mathbb{K}^{I.I.I}$ , the tensor associated to  $v$  denoted by  $\text{ten}(v) \in \mathbb{K}^{I \times I \times I}$  is obtained by dividing  $v$  into  $I$  vectors such that each one of them is of dimension  $I.I$  and then unvectorize each vector  $v$  into a matrix  $\text{unvec}(v) \in \mathbb{K}^{I.I}$ . Each matrix is a frontal slice of resulting tensor such that

$$\text{ten}(v)[:, :, i] := \text{unvec}(v_i)$$

for  $i = 1, \dots, I$ .

**Example 3.1.20.** For a vector  $v := \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \end{bmatrix} \in \mathbb{K}^8$ , the  $w_1 := \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}$  and the

$w_2 := \begin{bmatrix} v_5 \\ v_6 \\ v_7 \\ v_8 \end{bmatrix}$  and the  $\text{ten}(v)[:, :, 1] := \text{unvec}(w_1) = \begin{bmatrix} v_1 & v_3 \\ v_2 & v_4 \end{bmatrix}$  and  $\text{ten}(v)[:, :, 2] := \text{unvec}(w_2) = \begin{bmatrix} v_5 & v_7 \\ v_6 & v_8 \end{bmatrix}$

Now we recall the Khatri-Rio product of two matrices and two vectors

**Definition 3.1.21.** Let  $A = [a_1, a_2, \dots, a_k] \in \mathbb{K}^{I \times K}$  and  $B = [b_1, b_2, \dots, b_k] \in \mathbb{K}^{J \times K}$  be two matrices. The Khatri-Rio product denoted by  $A \odot B$  is the matrix  $\in \mathbb{K}^{I.J \times K}$  obtained by the outer product of each column of  $A$  by each column of  $B$  such that:

$$A \odot B = [\text{vec}(a_1 \otimes b_1), \text{vec}(a_2 \otimes b_2), \dots, \text{vec}(a_k \otimes b_k)]$$

If  $a$  and  $b$  are vectors then the Khatri-Rio product is the same of the outer product.

**Example 3.1.22.** If  $A = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \in \mathbb{K}^{2 \times 2}$  and  $B = \begin{bmatrix} b_{1,1} & b_{1,2} \\ b_{2,1} & b_{2,2} \\ b_{3,1} & b_{3,2} \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 0 & 1 \end{bmatrix} \in \mathbb{K}^{3 \times 2}$  then the Khatri-Rio product is defined by:

$$A \odot B = [\text{vec}(a_1 \otimes b_1), \text{vec}(a_2 \otimes b_2)]$$

where  $a_1 \otimes b_1 = \begin{bmatrix} a_{1,1}b_{1,1} & a_{1,1}b_{2,1} & a_{1,1}b_{3,1} \\ a_{2,1}b_{1,1} & a_{2,1}b_{2,1} & a_{2,1}b_{3,1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 2 & 2 & 1 \end{bmatrix}$  and  $a_2 \otimes b_2 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

**Definition 3.1.23.** The Frobenius norm of a tensor  $[T] \in \mathbb{K}^{[n_1] \times [n_2] \times \dots \times [n_k]}$  is the square root of the sum of the squares of all its elements such that:

$$\|T\|_F^2 = \sum_{\substack{i_1 \in [n_1] \\ i_2 \in [n_2] \\ \vdots \\ i_n \in [n_k]}} t_{i_1, i_2, \dots, i_k}^2.$$

### 3.1.4 Multi-Symmetric Tensor

**Definition 3.1.24.** Let  $(E_j)_{1 \leq j \leq k}$  be a family of  $n_j + 1$  dimensional vector spaces over the field  $\mathbb{K}$  for  $j = 1, \dots, k$ . Each one of them is of basis  $\mathbf{x}_j$  such that  $E_j = \langle \mathbf{x}_j \rangle = \langle x_{j,0}, \dots, x_{j,n_j} \rangle$ .

**Definition 3.1.25.**  $\mathcal{S}^{\delta_j}(E_j)$  is the vector space of homogeneous polynomials in the variables  $\mathbf{x}_j$  of degree  $\delta_j$ .

**Definition 3.1.26.**  $\mathcal{S}^{\delta_1}(E_1) \otimes \mathcal{S}^{\delta_2}(E_2) \otimes \dots \otimes \mathcal{S}^{\delta_k}(E_k)$  is the vector space of multi-homogeneous polynomials of degree  $\delta_j$  in each subset of variables  $\mathbf{x}_j$  for  $j = 1, \dots, k$ . An element  $T$  of this vector space is called a multi symmetric tensor. The vector space is denoted hereafter as  $\mathcal{S}^\delta(E)$ .

**Definition 3.1.27.** A multi symmetric tensor

$$T(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k) = \sum_{\substack{|\alpha_j| = \delta_j \\ \alpha'_j \in \mathbb{N}^{n_j+1}}} t_{\alpha'_1, \alpha'_2, \dots, \alpha'_k} (\mathbf{x}_1)^{\alpha'_1} (\mathbf{x}_2)^{\alpha'_2} \dots (\mathbf{x}_k)^{\alpha'_k}$$

of  $\mathcal{S}^{\delta_1}(E_1) \otimes \mathcal{S}^{\delta_2}(E_2) \otimes \dots \otimes \mathcal{S}^{\delta_k}(E_k)$  can be interpreted as a multi symmetric array  $[T]$  of coefficients  $[t_{\alpha'_1, \alpha'_2, \dots, \alpha'_k}]_{\substack{|\alpha'_j|=\delta_j \\ \alpha'_j \in \mathbb{N}^{n_j+1}}}$  such that each  $\alpha'_j = (\alpha'_{j,p_j})_{0 \leq p_j \leq n_j}$  is a multi-index for  $1 \leq j \leq k$ .

For  $\alpha_j \in \mathbb{N}^{n_j}$  with  $|\alpha_j| \leq \delta_j$ , we denote  $\bar{\alpha}_j = (\delta_j - |\alpha_j|, \alpha_{j,1}, \dots, \alpha_{j,n_j})$  for  $j = 1, \dots, k$ . The multi symmetric array of coefficients associated to this multi-index is defined as  $[T] = [t_{\bar{\alpha}_1, \bar{\alpha}_1, \dots, \bar{\alpha}_k}]_{\substack{|\alpha_j| \leq \delta_j \\ \alpha_j \in \mathbb{N}^{n_j}}}$ .

If we let  $x_{j,0} = 1$  for  $j = 1, \dots, k$  we get

$$\underline{T}(\underline{\mathbf{x}}_1, \underline{\mathbf{x}}_2, \dots, \underline{\mathbf{x}}_k) = \sum_{\substack{|\alpha_j| \leq \delta_j \\ \alpha_j \in \mathbb{N}^{n_j}}} t_{\alpha_1, \alpha_2, \dots, \alpha_k} \underline{\mathbf{x}}_1^{\alpha_1} \underline{\mathbf{x}}_2^{\alpha_2} \dots \underline{\mathbf{x}}_k^{\alpha_k}$$

where  $\underline{\mathbf{x}}_j = (x_{j,1}, \dots, x_{j,n_j})$  for  $j = 1, \dots, k$  because of  $\underline{\mathbf{x}}_j^{\alpha_j} = \underline{\mathbf{x}}_j^{\bar{\alpha}_j}$  when  $x_{j,0} = 1$  for  $j = 1, \dots, k$ .

We denote  $R_{\delta_1, \delta_2, \dots, \delta_k}$  the space obtained by deshomogenisation of elements in  $\mathcal{S}^\delta(E)$  by setting  $x_{j,0} = 1$  for  $j = 1, \dots, k$  where  $R = \mathbb{K}[\underline{\mathbf{x}}_1, \underline{\mathbf{x}}_2, \dots, \underline{\mathbf{x}}_k]$  is the space of polynomials in the variables  $\underline{\mathbf{x}}_j = (x_{j,1}, \dots, x_{j,n_j})$  for  $j = 1, \dots, k$ .

## 3.2 Rank One Tensors from a Geometric Point of View

The real projective space denoted by  $\mathbb{P}(\mathbb{R}^n)$  is the space of lines in  $\mathbb{R}^{n+1}$  passing through the origin. It is defined as  $\mathbb{P}(\mathbb{R}^n) \simeq (\mathbb{R}^{n+1} - \{0\}) / \sim$  where the equivalence relation satisfies the property:  $\forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^{n+1}$ ,  $\mathbf{u} = (u_0, u_1, \dots, u_n) \sim \mathbf{v} = (v_0, v_1, \dots, v_n)$  if and only if  $\exists \lambda \neq 0$  such that  $(u_0, u_1, \dots, u_n) = \lambda(v_0, v_1, \dots, v_n)$ .  $\bar{\mathbf{u}}$  is named the projective class of the vector  $\mathbf{u}$  which is one dimensional tensor.

The projective space of a  $n + 1$  dimensional vector space  $E := \mathbb{K}^{n+1}$  over the field  $\mathbb{K}$  is denoted by  $\mathbb{P}^n$ .

Let  $\mathbb{P}^{n_1}, \mathbb{P}^{n_2}, \dots, \mathbb{P}^{n_k}$  be respectively the associated projective spaces to  $E_1, E_2, \dots, E_k$  which are  $n_1 + 1, n_2 + 1, \dots, n_k + 1$  dimensional vector spaces over  $\mathbb{K}$ .

In this section, we describe the set of rank one tensors so called indecomposable tensors which form projective varieties such as Segré and Veronese varieties.

### 3.2.1 Segré variety

We first define the geometric object which describes the projective variety of non symmetric tensors of rank one named Segré variety.

**Definition 3.2.1.** *The Segré variety of  $k$  factors is the image of the following map:*

$$\begin{aligned} \phi_k : \mathbb{P}^{n_1} \times \mathbb{P}^{n_2} \times \dots \times \mathbb{P}^{n_k} &\rightarrow \mathbb{P}(E_1 \otimes E_2 \otimes \dots \otimes E_k) \\ (\overline{\mathbf{u}}_1, \overline{\mathbf{u}}_2, \dots, \overline{\mathbf{u}}_k) &\mapsto \overline{\mathbf{u}}_1 \otimes \overline{\mathbf{u}}_2 \otimes \dots \otimes \overline{\mathbf{u}}_k \end{aligned}$$

It is denoted by  $\Xi(E_1 \otimes E_2 \otimes \dots \otimes E_k)$ . As  $\phi_k$  is well defined and  $\mathbf{u}_1 \otimes \mathbf{u}_2 \otimes \dots \otimes \mathbf{u}_k$  is a tensor of rank one 3.4.2, then the Segré variety describes the projective classes of tensors of rank one.

**Definition 3.2.2.** *The array of coefficients  $[H] := [h_{i_1, i_2, \dots, i_k}]_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}}$  is said to be generic array of indeterminates if its entries are independent variables of  $\mathbb{K}[\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_k]$ .*

**Definition 3.2.3.** *A  $d$ -minor of a matrix  $A$  is the determinant of some smaller square matrix  $\in \mathbb{K}^{d \times d}$ , cut down from  $A$  by removing one or more of its rows or columns. A  $d$ -minor of a tensor  $T$  is a  $d$ -minor of its mode-1 flattening of it.*

In the following theorem, we describes how to compute the set of equations of the Segré variety:

**Theorem 3.2.4.**  $[H02]$

*If  $[H]$  is a generic array of coefficients in  $\mathbb{K}[\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_k]$  and  $I_d(H)$  is the ideal generated by the  $d$ -minors of  $H$  then  $I_2(H)$  is a prime ideal, therefore:*

$$I(\Xi(E_1 \otimes E_2 \otimes \dots \otimes E_k)) = I_2([H]).$$

### 3.2.2 Veronese variety

We define the geometric object which describes the projective variety of  $d$ -order symmetric tensors of rank one named Veronese variety.

**Definition 3.2.5.** *The Veronese variety is the image of the following map:*

$$\begin{aligned} v_d : \mathbb{P}^n &\rightarrow \mathbb{P}(\mathcal{S}^d(E)) \\ \bar{\mathbf{u}} &\mapsto \overline{\mathbf{u}^{\otimes d}} \end{aligned}$$

It is denoted by  $\Xi(\mathcal{S}^d(E))$ . As  $v_d$  is well defined and  $\mathbf{u}^{\otimes d}$  is a  $d$ -order symmetric tensor of rank one, then the Veronese variety describes the projective class of  $d$ -order symmetric tensors of rank one.

As  $\mathcal{S}^d(E)$  is isomorphic to  $\mathbb{P}^{\binom{n+d}{d}-1}$ , so the Veronese variety can be seen as the  $d$ -embedding of  $\mathbb{P}^n$  into  $\mathbb{P}^{\binom{n+d}{d}-1}$ .

In the following theorem, we describe how to compute the set of equations of the Veronese variety:

**Theorem 3.2.6.** *Let  $[H] := [h_{i_1, \dots, i_d}]_{\substack{0 \leq i_j \leq n \\ 1 \leq j \leq d}}$  is a generic symmetric array of indeterminates in  $\mathbb{K}[\mathbf{h}]$ , then the variety defined by symmetric tensors of rank one is the ideal generated by 2-minors of  $H$ ,  $I_2([H])$ :*

$$I(\Xi(\mathcal{S}^d(E))) = I_2([H])$$

In [Puc98], the autor proved that the  $I(\Xi(\mathcal{S}^d(E)))$  is generated by the 2-minors of catalecticant matrix called also Hankel matrix.

### 3.2.3 Segré-Veronese variety

We denote by  $\mathcal{S}^{\delta_1}(E_1) \otimes \mathcal{S}^{\delta_2}(E_2) \otimes \dots \otimes \mathcal{S}^{\delta_k}(E_k)$  the vector space of multi-symmetric tensors. We define the geometric object which describes the projective variety of the partial symmetric tensors of rank one named Segré-Veronese variety.

**Definition 3.2.7.** *The Segré-Veronese variety is the image of composition of two maps. The first one is the product of  $k$  Veronese maps defined as follows:*

$$\begin{aligned} v_{\delta_1} \times v_{\delta_2} \times \dots \times v_{\delta_k} : \mathbb{P}^{n_1} \times \mathbb{P}^{n_2} \times \dots \times \mathbb{P}^{n_k} &\rightarrow \mathbb{P}(\mathcal{S}^{\delta_1}(E_1)) \times \mathbb{P}(\mathcal{S}^{\delta_1}(E_2)) \times \dots \times \mathbb{P}(\mathcal{S}^{\delta_1}(E_k)) \\ (\bar{\mathbf{u}}_1, \bar{\mathbf{u}}_2, \dots, \bar{\mathbf{u}}_k) &\mapsto (\overline{\mathbf{u}_1^{\otimes \delta_1}}, \overline{\mathbf{u}_2^{\otimes \delta_2}}, \dots, \overline{\mathbf{u}_k^{\otimes \delta_k}}) \end{aligned}$$

Each  $\mathcal{S}^{\delta_j}(E_j)$  is isomorphic to  $\mathbb{P}^{\binom{n_j+\delta_j}{\delta_j}-1}$ , so that each Veronese variety is the  $\delta_j$ -embedding of  $\mathbb{P}^{n_j}$  into  $\mathbb{P}^{\binom{n_j+\delta_j}{\delta_j}-1}$  for  $j = 1, \dots, k$ . The second one is the Segré map of  $k$ -factors defined as follows:

$$s : \mathbb{P}(\mathcal{S}^{\delta_1}(E_1)) \times \mathbb{P}(\mathcal{S}^{\delta_1}(E_2)) \times \dots \times \mathbb{P}(\mathcal{S}^{\delta_1}(E_k)) \rightarrow \mathbb{P}(\mathcal{S}^{\delta_1}(E_1) \otimes \mathcal{S}^{\delta_1}(E_2) \otimes \dots \otimes \mathcal{S}^{\delta_1}(E_k))$$

$$(\overline{\mathbf{v}}_1, \overline{\mathbf{v}}_2, \dots, \overline{\mathbf{v}}_k) \mapsto \overline{\mathbf{v}}_1 \otimes \mathbf{v}_2 \otimes \dots \otimes \mathbf{v}_k$$

As  $\mathbb{P}(\otimes_{j=1}^k \mathcal{S}^{\delta_j}(E_j))$  is isomorphic to  $\mathbb{P}^{\binom{\sum_{j=1}^k (n_j+\delta_j)}{\sum_{j=1}^k \delta_j}-1}$  then  $s$  can be defined from  $\prod_{j=1}^k \mathbb{P}^{\binom{n_j+\delta_j}{\delta_j}-1}$  to  $\mathbb{P}^{\binom{\sum_{j=1}^k (n_j+\delta_j)}{\sum_{j=1}^k \delta_j}-1}$ .

Therefore the Segré-Veronese variety is the Segré embedding variety of the product of  $k$  Veronese varieties.

Notice that, if  $\delta_j = 1$  for all  $j = 1, \dots, k$ , the Segré-Veronese variety is nothing else than the Segré variety and the mutli-symmetric tensor can be seen as a non symmetric tensor.

If  $k = 1$ , the Segré-Veronese variety is nothing else than the Veronese variety and the mutli-symmetric tensor can be seen simply as a symmetric tensor.

In the following theorem, we describes how to compute the set of equations of the Segré-Veronese variety:

**Theorem 3.2.8.** *Let  $[X]$  is a generic symmetric array of indeterminates in  $\mathbb{K}[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$  where each  $\mathbf{x}_j = [x_{i_j,j}]_{0 \leq i_j \leq n_j}$  for  $j = 1, \dots, k$ , then the variety defined by partial symmetric tensors of rank one is the ideal generated by 2-minors of  $X$  such that:*

$$I(\Xi(\mathcal{S}^{\delta_1}(E_1) \otimes \mathcal{S}^{\delta_1}(E_2) \otimes \dots \otimes \mathcal{S}^{\delta_1}(E_k))) = I_2([X])$$

*Proof.* See [Ber08]. □

An element of the Segré-Veronese variety can be seen as a multi-homogeneous polynomial which is equal to a product of power of linear forms.

### 3.2.4 Rank and Border Rank

We recall that the minimal number of terms in the decomposition of  $T(x)$  is called the rank of  $T$ . We define also the border rank of a tensor using the secant variety.

**Definition 3.2.9.** Let  $X$  be a projective variety and define  $X_r^0$  to be the union of  $r$  projective classes of points such that  $X_r^0 := \bigcup_{\overline{P}_1, \overline{P}_2, \dots, \overline{P}_r \in X} \langle \overline{P}_1, \overline{P}_2, \dots, \overline{P}_r \rangle$ . The  $r$ -th secant variety  $X_r$  of  $X$  is the Zariski closure of  $X_r^0$ .

Observe that if  $\overline{P} \in X_r$  is a generic element so  $\overline{P} \in X_r^0$  therefore  $\overline{P} = \sum_{i=1}^r l_i \overline{Q}_i$  where  $\overline{Q}_i \in X$  and  $l_i \in \mathbb{K}$ .

If  $X = \Xi(\mathcal{S}^{\delta_1}(E_1) \otimes \mathcal{S}^{\delta_1}(E_2) \otimes \dots \times \mathcal{S}^{\delta_1}(E_k))$  is a Segré-Veronese variety therefore a generic element of  $X_r = \Xi_r(\mathcal{S}^{\delta_1}(E_1) \otimes \mathcal{S}^{\delta_1}(E_2) \otimes \dots \times \mathcal{S}^{\delta_1}(E_k))$  is the projective class  $\overline{T}$  of a partial symmetric tensor  $T$  that can be written as linear combination of linearly independent elements of Segré-Veronese variety which are partial symmetric tensors of rank one.

**Definition 3.2.10.** The minimal integer  $r$  such that a projective class of a tensor  $\overline{T} \in \mathbb{P}(\mathcal{S}^{\delta_1}(E_1) \otimes \mathcal{S}^{\delta_1}(E_2) \otimes \dots \times \mathcal{S}^{\delta_1}(E_k))$  is in the secant Segré-Veronese variety is the border rank of  $T$ .

If  $\overline{T} \in X_r \setminus X_r^0$  then the rank of  $T$  is strictly bigger than  $r$ .

In other terms, the border rank of  $T$  is the minimal number of equations in the secant variety.

### 3.3 Symmetric and Non Symmetric Tensor Decomposition problem in the literature

Given a general  $k$ -order tensor  $T$  of rank  $r$ , the goal is to decompose it into a sum of  $r$  tensors of rank one which is called parallel factors decomposition of  $T$ . We replace the decomposition problem by the minimization of the following cost function

$$\underset{\tilde{T}}{\text{minimize}} \quad \|T - \tilde{T}\|_F^2 \quad (3.1)$$

such that

$$\tilde{T} = \sum_{i=1}^r u_{i,1} \otimes u_{i,2} \otimes \dots \otimes u_{i,k}$$

The tensorial decomposition is the generalization of the diagonalisation of a matrix by equivalence transformation (non symmetric case) or by congruence transformation (symmetric case). The rank of a tensor can be larger than



the minimal of the dimensions of vector spaces. The uniqueness on tensor rank bounds comparing to dimensions of vector spaces are more detailed in [Ste10] and [Ste11]. The decomposition is stable by permutation of rank one tensors and scaling of them.

### 3.3.1 Power Method

#### Matrix case

In this section we show how to deduce eigenvalues and eigenvectors of a symmetric matrix using the Rayleigh quotient algorithm. We introduce the power iteration, the inverse iteration and the Rayleigh quotient iteration in order to compute the eigendecomposition of a rank  $r$  symmetric matrix. We first introduce the notion of a Rayleigh quotient and we describe intuitively why it can be used to compute the eigendecomposition of a symmetric matrix.

#### Eigenvalues and eigenvectors of a decomposable Matrix

In this section, we recall some basic definitions and properties of eigenvalues and eigenvectors of a decomposable matrix of rank  $r$ .

**Definition 3.3.1.** *The eigendecomposition of symmetric matrix  $M$  of rank  $r$  is defined as follows:*

$$M = V\Omega V^\top.$$

where  $V = [v_1 | v_2 | \dots | v_r]$  is the matrix with orthonormal vectors as columns and  $\omega = \text{diag}(\omega_1, \omega_2, \dots, \omega_r)$  is the diagonal matrix with non-zero eigenvalues. In other words, the decomposition can be expressed as follows:

$$M = \sum_{i=1}^r \omega_i v_i v_i^\top. \tag{3.2}$$

For a symmetric matrix  $M$ , the orthogonal decomposition is guaranteed.

**Definition 3.3.2.** *Let  $M$  be a matrix, the vector-valued map associated to it is*

$$u \mapsto Mu = \sum_{i=1}^n \sum_{j=1}^n m_{i,j} (ue_j^\top) ue_i.$$

**Definition 3.3.3.** We say that a vector  $u \in \mathbb{R}^n$  with  $\|u\| = 1$  is an eigenvector of  $M$  associated to an eigenvalue  $\omega \in \mathbb{R}$  if  $Mu = \omega u$ .

**Proposition 3.3.4.** Each vector  $v_i$  of the rank  $r$  orthogonal decomposable matrix (3.2) is an eigenvector of  $M$  associated to the eigenvalue  $\omega_i$  of  $M$ .

The linear combination of two eigenvectors  $v_1$  and  $v_2$  associated to a multiple eigenvalue  $\omega_1 = \omega_2 = \omega$  of  $M$  is an eigenvector of  $M$ . With the multiple eigenvalue  $\omega$ , any linear combination of  $v_1$  and  $v_2$  is fixed under  $M$ . However, in this case, the decomposition is not unique.

Nevertheless, the decomposition is unique when the weights  $\omega_1, \omega_2, \dots, \omega_r$  are distinct, whereupon the  $v_1, v_2, \dots, v_r$  are the only directions fixed under  $u \mapsto Mu$  up to non-trivial scaling.

### Rayleigh Quotient

**Definition 3.3.5.** The Rayleigh quotient of a vector  $u \in \mathbb{R}^n$  is the scalar defined as follows:

$$r(u) := \frac{u^\top M u}{\|u\|^2}.$$

Notice that if  $u$  is an eigenvector of  $M$  so that  $r(u)$  is the corresponding eigenvalue.

The optimization problem which leads to compute the scalar  $\alpha$  which "acts most like an eigenvalue" for a fixed vector  $u \in \mathbb{R}^n$  is then:

$$\underset{\alpha}{\text{minimize}} \quad \|Mu - \alpha u\|^2. \tag{3.3}$$

To make these ideas quantitative, it is fruitful to see  $u \in \mathbb{R}^n$  as a variable and  $u \mapsto r(u)$  as a function from  $\mathbb{R}^n$  to  $\mathbb{R}$ .

The gradient of  $r(u)$  with respect to  $u$  denoted by  $\nabla(r(u))$  is equal to:

$$\nabla(r(u)) = \frac{2}{\|u\|^2} (Mu - r(u)u).$$

From this formula, we deduce that if  $u$  is an eigenvector of  $M$  then  $\nabla(r(u)) = 0$ . Conversely, if  $\nabla(r(u)) = 0$  with  $u \neq 0$ , then  $u$  is an eigenvector and  $r(u)$  is the corresponding eigenvalue. So that the eigenvectors of  $M$  are the stationary points of  $r(u)$ . Since  $r(u)$  is a continuous function on the unit

sphere so that the normalized eigenvectors of  $M$  are the stationary points of  $r(u)$ . In addition, if the eigenvalues are all simple then the normalized eigenvectors are isolated points.

If  $q_i$  is an eigenvector of  $M$ , then we deduce by smoothness of the function  $r(u)$  that the Rayleigh quotient is a quadratically estimate of eigenvalue of  $M$  such that:

$$r(u) - r(q_i) = O(\|u - q_i\|^2) \text{ if } u \rightarrow q_i. \quad (3.4)$$

### Power Iteration

This power iteration algorithm leads to a sequence  $v_i$  of eigenvectors which converges to the eigenvector corresponding to the largest eigenvalue of  $M$ . If we write  $v_0$  as a linear combination of  $q_i$  such that

$$v_0 = \sum_{i=1}^r a_i q_i$$

Since,  $v_k$  is a multiple of  $M^k v_0$  we obtain:

$$v_k = c_k M^k v_0 = c_k \sum_{i=1}^r a_i \omega_i^k q_i = c_k \omega_1^k \left( a_1 q_1 + \sum_{i=2}^r a_i \left( \frac{\omega_i}{\omega_1} \right)^k q_i \right). \quad (3.5)$$

So that, we obtain the following theorem:

**Theorem 3.3.6.** *Suppose  $|\omega_1| > |\omega_2| \geq \dots \geq |\omega_r| \geq 0$  and  $q_1^\top v_0 \neq 0$ , then the iterates of the Algorithm 2 satisfy*

$$\|v_k - (\pm q_1)\| = O\left(\left|\frac{\omega_2}{\omega_1}\right|^k\right),$$

$$\|\omega_k - (\pm \omega_1)\| = O\left(\left|\frac{\omega_2}{\omega_1}\right|^{2k}\right).$$

*Proof.* The first equation follows from (3.5), since  $a_1 = q_1^\top v_0 \neq 0$  by assumption. The second follows from this and (3.4). If  $\omega_1 > 0$ , then the  $\pm$  signs are all + or all -, whereas if  $\omega_1 < 0$ , they alternate.  $\square$

The power iteration helps us to compute only the eigenvector corresponding to the largest eigenvalue. If the two largest eigenvalue are close in magnitude the convergence will be very slow.

---

**Algorithm 2 : Power iteration**

---

Some vector  $v_0 \in \mathbb{R}^n$  with  $\|v_0\| = 1$ ;

**for**  $k=1,2,\dots$  **do**

$$\left[ \begin{array}{l} b = Mv_{k-1}; \\ v_k = \frac{b}{\|b\|}; \\ \omega_k = v_k^T M v_k \end{array} \right.$$


---

**Inverse Iteration**

The idea is to overcome the problem of two closed eigenvalues in magnitude which leads to a slow convergence. This process leads to a sequence of  $v_i$  which converge to the eigenvector corresponding to  $\mu$  where  $\mu$  is chosen to be closed to an eigenvalue of  $M$ .

**Theorem 3.3.7.** *For  $\mu \in \mathbb{R}$  is not an eigenvalue of  $M$ , the eigenvectors of  $(M - \mu I)^{-1}$  are the same of  $M$  and the corresponding eigenvalues are  $(\omega_i - \mu)^{-1}$  where  $\omega_i$  are the eigenvalues of  $M$ .*

Suppose that  $\mu$  is close to an eigenvalue  $\omega_{i_0}$  of  $A$ . So  $(\omega_{i_0} - \mu)^{-1}$  may be much larger than  $(\omega_i - \mu)^{-1}$  for all  $i \neq i_0$ . If we apply the power iteration to  $(M - \mu I)^{-1}$ , the process will converge rapidly to the largest eigenvalue  $(\omega_{i_0} - \mu)^{-1}$ .

So that, we obtain the following theorem:

**Theorem 3.3.8.** *Suppose that  $\omega_{i_0}, \omega_{j_0}$  is the first two closest eigenvalue to  $\mu$  and  $v_{i_0}^T v_0 \neq 0$ , then the iterates of the Algorithm 3 satisfy*

$$\|v_k - (\pm v_{i_0})\| = O\left(\left|\frac{\mu - \omega_{i_0}}{\mu - \omega_{j_0}}\right|^k\right),$$

$$\|\omega_k - (\pm \omega_{i_0})\| = O\left(\left|\frac{\mu - \omega_{i_0}}{\mu - \omega_{j_0}}\right|^{2k}\right).$$

The inverse iteration helps us to compute the eigenvectors if the eigenvalues are already known. The convergence of the iteration method is linear.

---

**Algorithm 3 :** Inverse iteration

---

Some vector  $v_0 \in \mathbb{R}^n$  with  $\|v_0\| = 1$ ;

**for**  $k=1,2,\dots$  **do**

$$\left[ \begin{array}{l} b = (M - \mu I)^{-1}v_{k-1}; \\ v_k = \frac{b}{\|b\|}; \\ \omega_k = v_k^T M v_k \end{array} \right.$$


---

### Rayleigh quotient iteration

The idea is to improve the eigenvalue estimation at each step of inverse iteration in order to accelerate the convergence.

So that, we obtain the following theorem,

**Theorem 3.3.9.** *If  $\omega_{i_0}$  is an eigenvalue of  $M$  and  $v_0$  is sufficiently closed to the eigenvector corresponding to  $\omega_{i_0}$ , then the iterates of the Algorithm 4 satisfy*

$$\begin{aligned} \|v_{k+1} - (\pm v_{i_0})\| &= O(\|v_k - v_{i_0}\|^3), \\ \|\omega_{k+1} - (\pm \omega_{i_0})\| &= O(|\omega_k - \omega_{i_0}|^3). \end{aligned}$$

The Rayleigh quotient converges cubically to an eigenvalue-eigenvector pair for all except a set of measure zero of starting vector.

---

**Algorithm 4 :** Rayleigh quotient iteration

---

Some vector  $v_0 \in \mathbb{R}^n$  with  $\|v_0\| = 1$ ;

$$\omega_0 = v_0^T M v_0;$$

**for**  $k=1,2,\dots$  **do**

$$\left[ \begin{array}{l} b = (M - \omega_{k-1} I)^{-1}v_{k-1}; \\ v_k = \frac{b}{\|b\|}; \\ \omega_k = v_k^T M v_k \end{array} \right.$$


---

## Tensor Case

### Eigenvalues and eigenvectors of a decomposable Tensor

In this section, we recall some basic definitions and properties of eigenvalues and eigenvectors of a decomposable tensor of rank  $r$ .

**Definition 3.3.10.** Let  $T = [t_{i,j,k}]_{\substack{0 \leq i \leq n \\ 0 \leq j \leq n \\ 0 \leq k \leq n}}$  be a symmetric tensor of dimension 3, the vector-valued map associated to this tensor  $T$  is

$$u \mapsto T(I, u, u) = \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n t_{i,j,k} (e_j^\top u) (e_k^\top u) e_i.$$

where  $I$  is  $n \times n$  identity matrix.

This is not a linear map.

**Definition 3.3.11.** We say that a vector  $u \in \mathbb{R}^n$  with  $\|u\| = 1$  is an eigenvector of  $T$  associated to an eigenvalue  $\omega \in \mathbb{R}$  if  $T(I, u, u) = \omega u$ .

To simplify the discussion, we assume throughout that eigenvectors have unit norm; otherwise, for scaling reasons, we replace the above equation with  $T(I, u, u) = \omega \|u\| u$ . This concept was originally introduced by Lim [Lim05] and [Qi05].

**Proposition 3.3.12.** For a symmetric tensor  $T$  of dimension 3

$$T = \sum_{i=1}^r \omega_i v_i \otimes v_i \otimes v_i,$$

the eigenvector of  $T$  verified the following expression:

$$T(I, u, u) = \sum_{i=1}^r \omega_i (u^\top v_i)^2 v_i.$$

Each orthogonal vector  $v_i$  of the decomposition is an eigenvector associated to the eigenvalue  $\omega_i$ .

Notice that, the orthogonal decomposition is not guaranteed for a symmetric tensor.

First, the linear combination of two eigenvectors associated to a multiple eigenvalue of  $T$  may not be an eigenvector of  $T$ . Second, the eigenvectors associated to simple eigenvalues are not the only eigenvectors of  $T$ . For example, set  $u := (\frac{1}{\omega_1})v_1 + (\frac{1}{\omega_2})v_2$  then

$$T(I, u, u) = \omega_1 \left(\frac{1}{\omega_1}\right)^2 v_1 + \omega_2 \left(\frac{1}{\omega_2}\right)^2 v_2 = u.$$

so  $\frac{u}{\|u\|}$  is an eigenvector.

### Rayleigh quotient

**Definition 3.3.13.** *The generalized Rayleigh quotient of a vector  $u \in \mathbb{R}^n$  associated to a third order tensor is*

$$r(u) := \frac{T(I, u, u)}{(u^\top u)^{1.5}}.$$

The following theorem ([NW99]) shows that a non-zero vector  $u \in \mathbb{R}^n$  is an isolated local maximizer of the generalized Rayleigh quotient if and only if it is equal to some vector  $v_i$  of the decomposition.

**Theorem 3.3.14.** *Let  $T$  be a symmetric tensor of rank  $r$  decomposed as:*

$$T(u, u, u) = \sum_{i=1}^r \omega_i (v_i^\top u)^3.$$

*Consider the optimization problem*

$$\underset{u \in \mathbb{R}^n, \|u\| \leq 1}{\text{maximize}} \quad T(u, u, u) = \underset{u \in \mathbb{R}^n, \|u\| \leq 1}{\text{maximize}} \quad (3.6)$$

*Then,*

- *The stationary points are the eigenvectors of  $T$ .*
- *A stationary point  $u$  is an isolated local maximizer if and only if  $u$  is equal to a vector  $v_i$  of the orthogonal decomposition.*

For the proof, see Appendix A.2 in [AGH<sup>+</sup>15].

In the following, we set up the condition which guarantee the uniqueness of orthogonal decomposition of rank  $r$  third order tensor.

**Definition 3.3.15.** We say that a unit vector  $u \in \mathbb{R}^n$  is a robust eigenvector of  $T$  if for all points in the ball  $B_{u,\xi}$  of center  $u$  and of radius  $\xi$ , repeated iteration of the map

$$\theta \mapsto \frac{T(I, \theta, \theta)}{\|T(I, \theta, \theta)\|} \quad (3.7)$$

starting from some fixed  $\theta_0 \in B_{u,\xi}$  converges to  $u$ .

The following theorem implies that the set of robust eigenvectors of  $T$  are the orthogonal vectors of the decomposition. In this case, the decomposition is unique.

**Theorem 3.3.16** ([AGH<sup>+</sup>15]). *Let  $T$  be a symmetric decomposable tensor of rank  $r$ , then we have the following*

- *The set of  $\theta \in \mathbb{R}^n$  which do not converge to some  $v_i$  under some repeated iteration (3.7) of has measure 0.*
- *The set of robust eigenvectors of  $T$  is equal to  $\{v_1, v_2, \dots, v_r\}$ .*

## Robust Tensor power Method

---

**Algorithm 5 :** Robust tensor power iteration

---

Multi-linear tensor  $T \in \mathbb{R}^{r \times r \times r}$ , number of iterations  $L, N$ ;

**for**  $t=1, \dots, L$  **do**

$u_0^{(t)} \in \mathbb{R}^r / \|u_0^{(t)}\| = 1$   
**for**  $k=1, \dots, N$  **do**  

 $b = \frac{T(I, u_{k-1}^{(t)}, u_{k-1}^{(t)})}{\|T(I, u_{k-1}^{(t)}, u_{k-1}^{(t)})\|};$

Let  $t^* := \operatorname{argmax}_{1 \leq t \leq L} T(u_N^{(t)}, u_N^{(t)}, u_N^{(t)})$

Do  $N$  power iteration updates starting from  $u_N^{(t^*)}$  to obtain  $\hat{u}$  and set  $\hat{\omega} := T(\hat{u}, \hat{u}, \hat{u})$

---

So that, we obtain the following theorem,

**Theorem 3.3.17.** *Let  $T$  is a symmetric decomposable tensor. For a vector  $u_0 \in \mathbb{R}^n$ , we suppose that the set of numbers  $\{|\omega_i v_i^T u_0|, 1 \leq i \leq r\}$  has a unique largest element. Without loss of generality, we say that  $|\omega_1 v_1^T u_0|$  and  $|\omega_2 v_2^T u_0|$  are the two largest values.*



For  $k = 1, 2, \dots$ , let

$$u_k := \frac{T(I, u_{k-1}, u_{k-1})}{\|T(I, u_{k-1}, u_{k-1})\|}$$

$$\|u_k - v_1\|^2 \leq \left(2\omega_1^2 \sum_{i=2}^r \omega_i^{-2}\right) \left| \frac{\omega_2 v_2^\top u_0}{\omega_1 v_1^\top u_0} \right|^{2k+1}.$$

The repeated iteration of (3.7) starting from  $u_0$  converges quadratically to  $v_1$ .

### 3.3.2 ALS algorithm

We compute the decomposition of non symmetric tensors of rank  $r$ . We replace the decomposition problem by the following minimization problem. The optimization problem is the following:

$$\underset{\tilde{T}}{\text{minimize}} \quad \|T - \tilde{T}\|_F^2 \quad (3.8)$$

where

$$\tilde{T} = \sum_{i=1}^r a_i \otimes b_i \otimes c_i \quad (3.9)$$

Using the Khatri-Product and the mode  $n$  flattening, we obtain  $\tilde{T}_{[1]} = A(C \odot B)^\top$ ,  $\tilde{T}_{[2]} = B(C \odot A)^\top$  and  $\tilde{T}_{[3]} = C(B \odot A)^\top$  where the so called factor matrices:  $A = [a_i]_{1 \leq i \leq r} \in \mathbb{K}^{I \times r}$ ,  $B = [b_i]_{1 \leq i \leq r} \in \mathbb{K}^{J \times r}$  and  $C = [c_i]_{1 \leq i \leq r} \in \mathbb{K}^{K \times r}$  represent respectively the  $x$ 's,  $y$ 's and  $z$ 's coordinates of  $\tilde{T}$ .

For example, if  $r = 2$ ,  $\tilde{T}_{[1]} = [a_1 \otimes b_1 \otimes c_1]_{[1]} + [a_2 \otimes b_2 \otimes c_2]_{[1]} = [a_1[:].b_1[j].c_1[k]]_{\substack{j \in J \\ k \in K}} + [a_2[:].b_2[j].c_2[k]]_{\substack{j \in J \\ k \in K}} = a_1[:][b_1[j].c_1[k]]_{\substack{j \in J \\ k \in K}} + a_2[:][b_2[j].c_2[k]]_{\substack{j \in J \\ k \in K}} = A(C \odot B)^\top \in \mathbb{K}^{I \times J \cdot K}$  and similarly for  $\tilde{T}_{[2]}$ .

Using 3 modes of flattening for Problem (3.8) for both  $T$  and  $\tilde{T}$  we obtain these three expressions

$$\underset{A, B, C}{\text{minimize}} \quad \|\tilde{T}_{[1]} - A(C \odot B)^\top\|_F^2, \quad (3.10)$$

$$\underset{A, B, C}{\text{minimize}} \quad \|\tilde{T}_{[2]} - B(C \odot A)^\top\|_F^2,$$

$$\underset{A, B, C}{\text{minimize}} \quad \|\tilde{T}_{[3]} - C(B \odot A)^\top\|_F^2$$

Fixing all factor matrices but one each time, the problem reduces to three linear least square problems:

$$\begin{aligned}
A^{k+1} &= \underset{\hat{A} \in \mathbb{K}^{I \times r}}{\operatorname{argmin}} \quad \|T_{[1]} - \hat{A}(C^k \odot B^k)^\top\|_F^2, \\
B^{k+1} &= \underset{\hat{B} \in \mathbb{K}^{J \times r}}{\operatorname{argmin}} \quad \|T_{[2]} - \hat{B}(C^k \odot A^{k+1})^\top\|_F^2, \\
C^{k+1} &= \underset{\hat{C} \in \mathbb{K}^{K \times r}}{\operatorname{argmin}} \quad \|T_{[3]} - \hat{C}(B^{k+1} \odot A^{k+1})^\top\|_F^2.
\end{aligned} \tag{3.11}$$

We start by initial values of factor matrices  $A^0, B^0$  and  $C^0$ . The ALS approach fixes  $C$  and  $B$  to solve  $A$ , then fixes  $C$  and  $A$  to obtain  $B$ , and then fixes  $B$  and  $A$  to obtain  $C$  until a convergence criterion is satisfied.

The linear independency of rank one tensors in (3.9) is sufficient to guarantee the uniqueness of decomposition in the tensor case under some conditions explained in section 1 of [DLDMV04]. This linear dependency has physical meaning in the higher order tensors decomposition. In the matrix case which consists of decomposing a matrix  $M$  into a sum of matrices of rank one, we need more stronger conditions to satisfy the uniqueness such as the orthogonality of rank one tensors.

To guarantee the convergence of ALS algorithm to the local optimum, we should repeat the optimization for a number of randomly chosen initial values of factor matrices. This process is consuming in time and the ALS iterations can be very slow [DLDMV04].

In [HK13] using the Gauss-Newton method, the factor matrices are updated simultaneously, but solving the resulting set of equations is computationally hard as well even if we add a regularized constraint on the entries to guarantee the determinacy of the decomposition.

The conditions for which the uniqueness is guaranteed are the following [Ste11]:

- $r \leq \min\{I_1, I_2\}$
- $\exists!$   $m$ -rank  $r_m$  for  $m \geq 3$  which satisfies  $r_m \geq 2$
- There is no single vector which is a linear combination for the other vectors in the set of columns of the factor matrix corresponding to  $x$ 's coordinates.

- There is no single vector which is a linear combination for the other vectors in the set of columns of the factor matrix corresponding to  $y$ 's coordinates. corresponding to  $x'_j$ 's coordinates for every  $j \geq 2$ .
- There is no single vector which is a scalar multiple for any other vector in the set of columns of each factor matrix.

For more details on the uniqueness conditions see [DLDMV04].

### 3.3.3 Simultaneous Eigenvalue Decomposition Method

In this section we describe another low rank decomposition method which is based on the diagonalization of a collection of similar matrices obtained from the given tensor. Another similar approach is analyzed in the context of latent variable model discussed in section 4. For more details, see [AGH<sup>+</sup>15].

Furthermore, a number of robust numerical methods for (approximately) simultaneously diagonalization collections of matrices have been proposed and used successfully in the literature (e.g. [ZLNMA04]).

We show that the CANDECOP can be reformulated as an orthogonal simultaneous matrix decomposition. The reformulation in terms of orthogonal unknowns allows for the application of typical numerical procedures that involve orthogonal matrices. This technique is applied to non-symmetric, instead of symmetric matrices. This generalization may raise some confusion. It might, for instance, be tempting to consider also a simultaneous lower triangularization, in addition to a simultaneous upper triangularization.

Given  $T \in \mathbb{R}^{I \times J \times K}$  a 3-order non symmetric tensor of rank  $r$  with the canonical decomposition

$$T = \sum_{i=1}^r \omega_i a_i \otimes b_i \otimes c_i.$$

We assume that  $I = J = r$  where  $r = \text{rank}(T)$ .

We associate to  $T$  a linear transformation  $f_T$  from the vector space  $\mathbb{R}^K$  to the matrix space  $\mathbb{R}^{I \times J}$  which associates to a vector  $P = (p_k)_{k \in K}$  a matrix  $V = (v_{i,j})_{i \in I, j \in J}$  such that:

$$V = f_T(P) = T \times_3 P \iff v_{i,j} = \sum_{k \in K} t_{i,j,k} p_k$$

for all  $i \in I, j \in J$ .

By substituting the expression of  $V$  in the canonical decomposition of  $T$ , we obtain:

$$V = ADB^\top.$$

in which

$$D = \text{diag}\{(\omega_1, \omega_2, \dots, \omega_r)\} \text{diag}\{C^\top P\}.$$

and

$$A = [a_1, a_2, \dots, a_r],$$

$$B = [b_1, b_2, \dots, b_r],$$

$$C = [c_1, c_2, \dots, c_r].$$

are the factor matrices.

If the range of the mapping  $f_T$  is spanned by the matrices  $V_1, V_2, \dots, V_L$ , we diagonalize by equivalence each one of them with the matrices  $A$  and  $B$  and we obtain the following simultaneous decomposition:

$$\begin{cases} V_1 = AD_1B^\top, \\ \vdots \\ V_L = AD_LB^\top. \end{cases}$$

where  $D_1, D_2, \dots, D_L$  are diagonal matrices where the diagonal elements of each one of them are the corresponding eigenvalues of  $V_1, V_2, \dots, V_L$ .

If  $V_l$  for  $l = 1, \dots, L$  consists of the matrix slices  $t_{:, :, k}$  of the tensor  $T$ , their corresponding vectors  $P_k$  are the canonical unit vectors  $e_k$  for  $k \in K$  such that:

$$f_T(e_k) = t_{:, :, k} \text{ for } k \in K.$$

We define  $\tilde{C} = (\tilde{c}_{l, :})_{1 \leq l \leq L}$  where each row  $\tilde{c}_{l, :}$  consists of the elements of diagonal matrix  $D_l$  such that  $\tilde{c}_{l, :} = (D_l(i, i))_{1 \leq i \leq r}$  for  $l = 1, \dots, L$ .

Therefore

$$\tilde{C} = [P_1 P_2 \dots P_L]^\top C \text{diag}\{(\omega_1, \omega_2, \dots, \omega_r)\}.$$

It is possible by a generic choice of  $P_1$  to assume that  $V_1$  is of full rank. We multiply  $V_2$  by the inverse of  $V_1$  which gives the following eigenvalue decomposition:

$$V_2 V_1^{-1} = A(D_2 D_1^{-1})A^{-1}.$$

We assume that  $C$  doesn't contain any collinear columns so that the diagonal elements of  $D_2 D_1^{-1}$  are mutually different and the eigenvalue decomposition problem reveals the columns of  $A$ , up to irrelevant scaling and/or permutation.

The simultaneous eigenvalue decomposition problem reduces to solving the system of equations:

$$\{V_l V_l^{-1} = A D_l D_1^{-1} A^{-1}\}_{2 \leq l \leq L}.$$

Once  $A$  is known,  $B$  can be obtained up to scaling of its columns, as follows:

We have

$$\begin{cases} V_1^T A^{-T} = B D_1, \\ \vdots \\ V_L^T A^{-T} = B D_L. \end{cases}$$

Hence, if we denote the  $i^{th}$  column of  $V_l^T A^{-T}$  by  $b_{l,i}$ , the columns  $B_i$  of  $B$  for  $i = 1, \dots, r$  can be estimated by the dominant left singular vectors of  $[b_{1,i} b_{2,i} \dots b_{L,i}]$ . See [DLDMV04]. Finally the matrix  $C \text{diag}\{\omega_1, \omega_2, \dots, \omega_r\}$  is found by solving a linear set of equations for a given matrices  $A$  and  $B$ .

### Algorithm

In this section we describe decomposition algorithm based on simultaneous eigenvalue decomposition of matrices obtained from a given tensor (see [DLDMV04] for more details) which is used to approximate a non symmetric tensor  $T$  of known rank  $r$  by its non symmetric decomposition which consists of computing the factor matrices  $A, B$  and  $C$  of the decomposition. A reduction step from a tensor  $T \in \mathbb{R}^{I \times J \times K}$  to a tensor  $S \in \mathbb{R}^{r \times r \times r_3}$  is required at the beginning of the algorithm.

We assume that  $r_1 = \text{rank}(T_{[1]}) = r$ ,  $r_2 = \text{rank}(T_{[2]}) = r$  and  $r_3 = \text{rank}(T_{[3]}) = \text{rank}(C)$ . We have to perform a best rank  $r_1, r_2$  and  $r_3$ -approximation of respectively mode 1, mode 2 and mode 3-flattening of  $T = \sum_{i=1}^r \omega_i a_i \otimes b_i \otimes c_i$ .

In the following, we assume that:

- $A$  is invertible

- $B$  is invertible
- $C$  does not contain collinear vectors.

These assumptions are required to guarantee the uniqueness of the solution. See [DLDMV04] for more details.

If  $r < \max(I, J)$  or  $r_3 < K$ , then we apply a dimensionality reduction step on  $T \in \mathbb{R}^{I \times J \times K}$  to a tensor  $S \in \mathbb{R}^{r \times r \times r_3}$ .

The reduction problem is to optimize the cost function such that

$$\underset{X, Y, Z}{\text{maximize}} \quad \|T \times_1 X^\top \times_2 Y^\top \times_3 Z^\top\|_F^2 \quad (3.12)$$

where  $X \in \mathbb{R}^{I \times r}$ ,  $Y \in \mathbb{R}^{J \times r}$ ,  $Z \in \mathbb{R}^{K \times r_3}$  are respectively column-wise orthonormal.

$T$  and the reduced tensor  $S$  are related by the formula:  $T = S \times_1 X \times_2 Y \times_3 Z$ .

If  $X, Y$  and  $Z$  are all column-wise orthonormal matrices then the optimal approximation of rank  $r$  of  $T$  and of its associated reduced tensor are related using the same formula:  $\tilde{T} = \tilde{S} \times_1 X \times_2 Y \times_3 Z$ , since the quadratic cost function  $\|T \times_1 X^\top \times_2 Y^\top \times_3 Z^\top\|_F^2$  does not change under the mode  $m$ -product

by column-wise orthonormal matrices.

---

**Algorithm 6 :** Decomposition of non symmetric tensor into CANDECOMP components

---

**Begin**

**Input:** the moments  $t_{i,j,k}$  of 3-order multi linear tensor  $T \in \mathbb{R}^{I \times J \times K}$  with known rank  $r \in \mathbb{N}$ .

- Compute the reduced tensor  $S$  as follows:
  - Any orthonormal basis of the mode 1, mode 2 and mode 3 flattening of  $T$  gives respectively  $X, Y$  and  $Z$ .
  - The  $r$  and  $r_3$  are computed using the singular values of the mode 1 and mode 3 flattening of  $T$ .
  - Compute the reduced tensor  $S$  using the orthonormal basis  $X, Y$  and  $Z$ .

- Compute  $V_1, V_2, \dots, V_L$  which generate the image of the linear transformation

$$f_T : \mathbb{R}^K \rightarrow \mathbb{R}^{I \times J}$$

$$Im(f_T) = rank(T) = \langle V_1, V_2, \dots, V_L \rangle.$$

- Find the maximum of the following cost function:

$$\sum_{i=1}^L \|Q \cdot V_i \cdot Z\|_{UF}^2$$

where  $\|\cdot\|_{UF}$  represents the Frobenius norm of the upper triangular part of a matrix.

This is equivalent to solve the right part of the set of matrix equations so called *simultaneous generalized Schur decomposition*

$$\begin{cases} QV_1Z = R_1 = R'D_1R'' , \\ QV_2Z = R_2 = R'D_2R'' , \\ \vdots \\ QV_LZ = R_L = R'D_LR'' . \end{cases}$$

in order to obtain the orthogonal matrices  $Q$  and  $Z$  which makes  $R_i$  for  $i = 1, \dots, L$  are upper triangular. A more detailed description is in Section 5 of [DLDMV04].

- 
- 
- We assume that  $\text{diag}(R') = \text{diag}(R'') = I$ . Then  $D_i = \text{diag}(R_i)$  for  $1 \leq i \leq L$ .
  - The strictly upper diagonal elements of  $R'$  and  $R''$  can be estimated by subsequently solving the set of equations related to the entries of  $R_i$  for  $1 \leq i \leq L$  described in the Section 8 of [DLDMV04].
  - Compute  $A$  and  $B$  using the formulas  $A = Q^\top R'$  and  $B^\top = R'' Z$
  - Compute a tensor  $V \in \mathbb{R}^{r \times r \times L}$  with entries  $v_{i,j,l} = (V_l)_{ij}$  such that  $V_l = A D_l B^\top$  for  $1 \leq i \leq r, 1 \leq j \leq r$  and  $1 \leq l \leq L$ 
    - Compute the matrix representation  $V_{(r^2 \times L)} \in \mathbb{R}^{r^2 \times L}$  of  $V$ .
    - Deduce  $\tilde{C}$  such that  $V_{(r^2 \times L)} = (A \odot B)(\tilde{C})^\top$ .
    - Compute  $C$  up to a scaling of its columns using this formula

$$\tilde{C} = [P_1 P_2 \dots P_L]^\top C \text{diag}\{(\omega_1, \omega_2, \dots, \omega_r)\}.$$

**Output:** The factor matrices  $A \in \mathbb{R}^{I \times r}$ ,  $B \in \mathbb{R}^{J \times r}$  and  $C \in \mathbb{R}^{K \times r}$  which represent respectively the  $x$ 's,  $y$ 's and  $z$ 's coordinates of the approximated tensor.

---



In the symmetric case we know that  $A = B = C = U$ . So we use the so called AD-DC algorithm to find out the components defined by  $U$ .

### 3.4 Multi Symmetric Tensor Decomposition Problem

We recall the definition of minimal affine decomposition of a multi symmetric tensor as a weighted sum of product of power of linear forms. We show the relationship between the dual of deshomogenized tensor and the formal power series associated to it using the apolar product. Each moment of the formal power series can be deduced from its corresponding one of the associated tensor by dividing by the product of binomial coefficients. Furthermore, each moment of the formal power series is associated to a coefficient in the associated Hankel matrix. Then, after scaling by the linear form of the decomposition and multiplying the weights by the scaling factor we deduce by linearity that the dual of the Tensor can be decomposed as a weighted sum of evaluations.

**Definition 3.4.1.** *The tensor decomposition problem of  $T(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k)$  is the decomposition of  $T$  as a sum of product of power of linear forms such that  $T(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k) = \sum_{p=1}^r \omega_p \mathbf{u}_{p,1}^{\delta_1}(\mathbf{x}_1) \mathbf{u}_{p,2}^{\delta_2}(\mathbf{x}_2) \dots \mathbf{u}_{p,k}^{\delta_k}(\mathbf{x}_k)$  where  $\mathbf{u}_{p,j}(\mathbf{x}_j) = u_{p,j,0}x_{j,0} + u_{p,j,1}x_{j,1} + \dots + u_{p,j,n_j}x_{j,n_j}$  and*

$$\mathbf{u}_p = (u_{p,j,p_j})_{\substack{0 \leq p_j \leq n_j \\ 1 \leq j \leq k}} \\ = (u_{p,1}, u_{p,1,1}, \dots, u_{p,1,n_1}, u_{p,2}, u_{p,2,1}, \dots, u_{p,2,n_2}, \dots, u_{p,k}, u_{p,k,1}, \dots, u_{p,k,n_k}) \in \mathbb{K}^{\sum_{j=1}^k (n_j+1)}$$

*is the coefficient vector associated to the linear forms  $\mathbf{u}_{p,j}(\mathbf{x}_j)$  in the basis  $\mathbf{x}_j$  for  $j = 1, \dots, k$ .*

**Definition 3.4.2.** *The minimal number of terms in a decomposition of  $T(\mathbf{x})$  is called the rank of  $T$ .*

We say that  $T(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k)$  has an affine minimal decomposition of the previous form if  $u_{p,j} \neq 0$  for  $p = 1, \dots, r$  and  $j = 1, \dots, k$  where  $r$  is the rank of  $T$ .

**Lemma 3.4.3.** *By a generic change of coordinates in each  $E_j$ , we may assume that  $u_{p,j} \neq 0$  and that  $T$  has a minimal affine decomposition of the previous*

form. Then by scaling  $\mathbf{u}_p(\mathbf{x})$  and multiplying  $\omega_p$  by the  $d^{\text{th}}$  power of the scaling factor we may assume that  $u_{p,j} = 1$  for  $p = 1, \dots, r$  and  $j = 1, \dots, k$ . Thus the polynomial  $\underline{T}(\underline{\mathbf{x}}) = \sum_{p=1}^r \omega'_p \mathbf{u}'_p{}^\delta(\underline{\mathbf{x}}) = \sum_{p=1}^r \omega'_i \mathbf{u}'_{p,1}{}^{\delta_1}(\underline{\mathbf{x}}_1) \mathbf{u}'_{p,2}{}^{\delta_2}(\underline{\mathbf{x}}_2) \dots \mathbf{u}'_{p,k}{}^{\delta_k}(\underline{\mathbf{x}}_k)$

**Definition 3.4.4.** Let  $T_1(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k)$  and  $T_2(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k)$  be two tensors of  $\mathcal{S}^\delta(E)$ . The apolar product of  $\underline{T}_1(\underline{\mathbf{x}}_1, \underline{\mathbf{x}}_2, \dots, \underline{\mathbf{x}}_k)$  and  $\underline{T}_2(\underline{\mathbf{x}}_1, \underline{\mathbf{x}}_2, \dots, \underline{\mathbf{x}}_k)$  is defined as

$$\langle \underline{T}_1(\underline{\mathbf{x}}_1, \underline{\mathbf{x}}_2, \dots, \underline{\mathbf{x}}_k), \underline{T}_2(\underline{\mathbf{x}}_1, \underline{\mathbf{x}}_2, \dots, \underline{\mathbf{x}}_k) \rangle = \sum_{\substack{|\alpha_j| \leq \delta_j \\ \alpha_j \in \mathbb{N}^{n_j}}} \sigma_{\alpha_1, \alpha_2, \dots, \alpha_k}^{(1)} \bar{\sigma}_{\alpha_1, \alpha_2, \dots, \alpha_k}^{(2)} \binom{\delta}{\alpha} \text{ where} \\ \binom{\delta}{\alpha} = \binom{\delta_1}{\alpha_1} \binom{\delta_2}{\alpha_2} \dots \binom{\delta_k}{\alpha_k}.$$

**Definition 3.4.5.** The dual operator of a tensor is defined as

$$T^* : (R_{\delta_1, \delta_2, \dots, \delta_k}) \rightarrow (R_{\delta_1, \delta_2, \dots, \delta_k})^* \quad (3.13)$$

$$\underline{T}_2 \mapsto T^*(\underline{T}_2) = \langle \underline{T}(\underline{\mathbf{x}}), \underline{T}_2(\underline{\mathbf{x}}) \rangle \quad (3.14)$$

**Definition 3.4.6.** For  $T = (t_{\alpha_1, \alpha_2, \dots, \alpha_k})_{\substack{|\alpha_j| \leq \delta_j \\ \alpha_j \in \mathbb{N}^{n_j}}} \in \mathcal{S}^\delta(E)$  we denote

$\sigma_{\alpha_1, \alpha_2, \dots, \alpha_k}(T) = \sigma_{\alpha_1, \alpha_2, \dots, \alpha_k} = t_{\alpha_1, \alpha_2, \dots, \alpha_k} \binom{\delta_1}{\alpha_1}^{-1} \binom{\delta_2}{\alpha_2}^{-1} \dots \binom{\delta_k}{\alpha_k}^{-1}$ . The dual of the tensor

$T(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k) \in \mathcal{S}^\delta(E)$  is defined via the formal power series as

$$\sigma(\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_k) = T^*(\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_k) = \sum_{\substack{|\alpha_j| \leq \delta_j \\ \alpha_j \in \mathbb{N}^{n_j}}} \sigma_{\alpha_1, \alpha_2, \dots, \alpha_k} \frac{(\mathbf{y}_1)^{\bar{\alpha}_1}}{\bar{\alpha}_1!} \frac{(\mathbf{y}_2)^{\bar{\alpha}_2}}{\bar{\alpha}_2!} \dots \frac{(\mathbf{y}_k)^{\bar{\alpha}_k}}{\bar{\alpha}_k!}$$

where

$$(\mathbf{y}_j)^{\bar{\alpha}_j} = (y_j, y_{j,1}, \dots, y_{j,n_j})^{(\alpha_j, \alpha_{j,1}, \dots, \alpha_{j,n_j})} = \prod_{p_j=0}^{n_j} (y_{j,p_j})^{\alpha_{j,p_j}} \text{ for } j = 1, \dots, k$$

**Proposition 3.4.7.** The dual of the product of powers of linear forms  $\mathbf{u}_1^{\delta_1} \mathbf{u}_2^{\delta_2} \dots \mathbf{u}_k^{\delta_k}$  is the evaluation  $\mathbf{e}_{\mathbf{u}}$  at  $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_k)$ .

*Proof.* For  $T = \mathbf{u}_1^{\delta_1} \mathbf{u}_2^{\delta_2} \dots \mathbf{u}_k^{\delta_k}$  and any  $T' \in R_{\delta_1, \delta_2, \dots, \delta_k}$ , we check that  $\langle \underline{T}(\underline{\mathbf{x}}), \underline{T}'(\underline{\mathbf{x}}) \rangle = T'(\mathbf{u})$ . This shows that  $T^*$  coincides with the evaluation  $\mathbf{e}_{\mathbf{u}}$ .  $\square$

Thus if  $T = \sum_i \omega_i \mathbf{u}_{i,1}^{\delta_1} \mathbf{u}_{i,2}^{\delta_2} \dots \mathbf{u}_{i,k}^{\delta_k}$ , then  $T^*$  coincides with the weighted sum of evaluations  $T^* = \sum_i \omega_i \mathbf{e}_{\mathbf{u}_i}$  on  $R_{\delta_1, \delta_2, \dots, \delta_k}$ . We reduce the decomposition problem of  $T$  to the decomposition of  $T^*$  as a weighted sum of evaluations  $T^* = \sum_i \omega_i \mathbf{e}_{\mathbf{u}_i}$ .

### 3.5 Non Symmetric Tensor Decomposition Problem

In this section, we analyze the easiest case of multi symmetric tensor where it is of degree one in each group of sub-variables. Our goal is to decompose  $\sigma = T^*$  as a weighted sum of evaluations such that  $T^* = \sum_i \omega_i \mathbf{e}_{\mathbf{u}_i}$  by computing the eigen-structure of the quotient algebra  $\mathcal{A}_\sigma = \mathbb{K}[\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]/I_\sigma$  of the ring of polynomials by the kernel  $I_\sigma$  of the Hankel operator  $H_\sigma$ , using the multiplication operators. We simplify notation by using subscripts of variables and coefficients instead of multi-index exponents. We compute the truncated Singular Value Decomposition of a generic linear combination of shifted Hankel matrices by the first collection of variables. By linearity and properties of the multiplication operators by one variable described in section 2, we deduce the multiplication operators by linear combination of variables which could be used to compute weights and points.

We choose two monomial bases  $B_1$  and  $B_2$  indexing respectively rows and columns of the Hankel matrix  $H_{T^*}^{B_1, B_2}$  associated to the tensor  $T$ , such that the set of monomials  $\{B_1 * B_2 x_{j, i_j}, 0 \leq i_j \leq n_j, 1 \leq j \leq k\}$  span the set of deshomogenized polynomials  $R_{\delta_1, \delta_2, \dots, \delta_k}$ .

The matrix of the truncated Hankel operator in the basis  $B_1$  and the dual basis of  $B_2$  is

$$H_{T^*}^{B_1, B_2} = [t_{i_1, i_2, \dots, i_k}]_{\substack{0 \leq i_1 \leq n_1 \\ 0 \leq i_2 \leq n_2 \\ \vdots \\ 0 \leq i_k \leq n_k}}$$

The Hankel matrix associated to the tensor  $x_{1, i_1} * T^*$  is defined as  $H_{1, i_1} = H_{x_{1, i_1} * T^*}^{B_1, B_2} = H_{T^*}^{x_{1, i_1} * B_1, B_2} = [t_{\alpha + \beta}]_{\alpha \in x_{1, i_1} * B_1, \beta \in B_2}$ , all the elements of the matrix are divisible in  $x_{1, i_1}$  and of degree  $\delta$ .

The Hankel matrix associated to  $T^*$  in the monomials basis  $B_1$  and  $B_2$  is denoted by  $H_0$ . Let  $\lambda(\mathbf{x}_1) = \lambda_0 + \lambda_1 x_{1,1} + \dots + \lambda_{n_1} x_{1, n_1}$  is a linear form with generic chosen coefficients  $\lambda_{i_1}, i_1 = 0, \dots, n_1$ , we build a linear combination of  $H_{1, i_1}, i_1 = 0, \dots, n_1$  such that  $\widehat{H}_0 = \sum_{i_1=0}^{n_1} \lambda_{i_1} H_{1, i_1}$  we compute its singular value decomposition.

Computing the singular value decomposition of  $\widehat{H}_0$ , we obtain

$$\widehat{H}_0 = USV^\top$$

where  $S$  is the diagonal matrix of all singular values of  $\widehat{H}_0$  arranged in a

decreasing order,  $U$  is an unitary matrix whose columns are the left singular vectors of  $\widehat{H}_0$ ,  $V$  is an unitary matrix whose columns are the right singular vectors of  $\widehat{H}_0$ . We denote by  $U^H$  the hermitian transpose of  $U$  and  $\overline{V}$  the conjugate of  $V$ . We denote by  $U_r$  and  $V_r$  the truncated matrices of the first  $r$  columns of  $U$  and  $V$  and  $S_r$  the diagonal matrix of the first  $r$  rows and  $r$  columns of  $S$ .

We denote  $\overline{B}_1 = \langle 1, x_{1,1}, \dots, x_{1,n_1} \rangle$  and  $\overline{B}_2 = \langle 1, x_{k,1}, \dots, x_{k,n_k} \rangle$ . Let  $u_i = [u_{\alpha,i}]_{\alpha \in \overline{B}_1}$  and  $v_j = [v_{\beta,j}]_{\beta \in \overline{B}_2}$  be respectively the  $i^{\text{th}}$  and  $j^{\text{th}}$  columns of  $U^H$  and  $\overline{V}$ . We denote by  $u_i(\mathbf{x}) = u_i^\top \overline{B}_1$  and  $v_j(\mathbf{x}) = v_j^\top \overline{B}_2$  the corresponding polynomials. The bases formed by these first  $r$  polynomials are denoted  $\mathcal{U}_r^H := (u_i(\mathbf{x}_1))_{i=1,\dots,r}$  and  $\overline{\mathcal{V}}_r := (v_j(\mathbf{x}_1))_{j=1,\dots,r}$ . We denote by  $U_r^H$  (resp.  $\overline{V}_r$ ) the corresponding coefficient matrix, formed by the first rows (resp. columns) of  $U^H$  (resp.  $\overline{V}$ ). We denote by  $S_r$  the diagonal matrix of the first  $r$  rows and columns of  $S$ , formed by the first  $r$  singular values.

We denote by  $H_0^r$ ,  $H_{1,i_1}^r$  and  $\widehat{H}_0^r$  the matrices obtained by the truncated singular value decomposition of  $H_0$ ,  $H_{i_1}$  and  $\widehat{H}_0$  respectively.

We have the following property

$$H_{i_1}^r = (M_{x_{1,i_1}}^{\mathcal{U}_r^H})^\top H_0^r = H_0^r M_{x_{1,i_1} * T}^{\overline{\mathcal{V}}_r}$$

where  $M_{x_{1,i_1}}^{\mathcal{U}_r^H}$  (resp.  $M_{x_{1,i_1}}^{\overline{\mathcal{V}}_r}$ ) is the multiplication matrix by  $x_{1,i_1}$  in the basis  $\mathcal{U}_r^H$  (resp.  $\overline{\mathcal{V}}_r$ ) and  $M_{x_{1,i_1} * T}^{\overline{\mathcal{V}}_r}$  is the multiplication matrix by  $x_{1,i_1} * T$  in the basis  $\overline{\mathcal{V}}_r$ . Then by linearity, we obtain  $\widehat{H}_0^r = \sum_{i_1=0}^{n_1} \lambda_{i_1} H_{1,i_1}^r = H_0^r \sum_{i_1=0}^{n_1} \lambda_{i_1} M_{x_{1,i_1} * T}^{\overline{\mathcal{V}}_r} = H_0^r M_{\lambda(\mathbf{x}_1) * T}^{\overline{\mathcal{V}}_r}$ .

Then  $(\widehat{H}_0^r)^{-1} = (M_{\lambda(\mathbf{x}_1) * T}^{\overline{\mathcal{V}}_r})^{-1} (H_0^r)^{-1}$  so multiplying by the first equation we get

$$(\widehat{H}_0^r)^{-1} H_{1,i_1}^r = (M_{\lambda(\mathbf{x}_1) * T}^{\overline{\mathcal{V}}_r})^{-1} M_{x_{1,i_1} * T}^{\overline{\mathcal{V}}_r} = M_{(x_{1,i_1}/\lambda(\mathbf{x}_1)) * T}^{\overline{\mathcal{V}}_r}$$

We compute the eigenvalues and the eigenvectors of the multiplication matrices  $M_{(x_{1,i_1}/\lambda(\mathbf{x}_1)) * T}^{\overline{\mathcal{V}}_r}$  in order to obtain the weights and the points of the decomposition.

## Algorithm

We describe now the algorithm to recover the sum  $\underline{T}^*(\underline{\mathbf{x}}, \underline{\mathbf{y}}, \underline{\mathbf{z}}) = \sum_{p=1}^r \omega_p \mathbf{e}_{\mathbf{u}_p}(\underline{\mathbf{x}}, \underline{\mathbf{y}}, \underline{\mathbf{z}})$ ,  $\omega_p \in \mathbb{K} \setminus \{0\}$ ,  $\mathbf{u}_p \in \mathbb{K}^{\sum_{i=1}^3 (n_i+1)}$ , from the moments of degree at most one at each bunch of coordinates  $(t_{i,j,k})_{\substack{0 \leq i \leq n_1 \\ 0 \leq j \leq n_2 \\ 0 \leq k \leq n_3}}$  of the formal power series. To simplify, we change notations to better understand the nine dimensional multivariate space seen as three dimensional space. We only use 3 groups of variables and

denote  $x_{1,i_1}$  by  $x_i$  and  $x_{2,i_2}$  by  $y_j$  and  $x_{3,i_3}$  by  $z_k$ .

---

**Algorithm 7 :** Decomposition of non symmetric tensor with constant weights

---

**Input:** the moments  $(t_{i,j,k})_{\substack{0 \leq i \leq n_1 \\ 0 \leq j \leq n_2 \\ 0 \leq k \leq n_3}}$  of  $T$ .

1. Compute the monomial sets  $A_1 = (x_i y_j)_{\substack{0 \leq i \leq n_1 \\ 0 \leq j \leq n_2}}$  and  $A_2 = (z_0, z_1, \dots, z_{n_3})$  and substitute the  $x_0, y_0$  and  $z_0$  by 1 to define  $B_1$  and  $B_2$ .
2. Compute the Hankel matrix  $H_{T^*}^{B_1, B_2} = [t_{i,j,k}]_{\substack{0 \leq i \leq n_1 \\ 0 \leq j \leq n_2 \\ 0 \leq k \leq n_3}}$  for the monomial sets  $B_1$  and  $B_2$ .
3. Compute the singular value decomposition of  $H_{T^*}^{\bar{B}_1, \bar{B}_2} = USV^T$  where  $\bar{B}_1 = \langle 1, x_1, \dots, x_{n_1} \rangle$  and  $\bar{B}_2 = \langle 1, z_1, \dots, z_{n_3} \rangle$  with singular values  $s_1 \geq s_2 \geq \dots \geq s_m \geq 0$ .
4. Determine its numerical rank, that is, the largest integer  $r$  such that  $\frac{s_r}{s_1} \geq \epsilon$ .
5. Form the multiplication matrices by  $y_j$  in the basis  $\bar{V}_r$ ,  

$$M_{y_j}^{\bar{V}_r} = S_r^{-1} U_r^H H_{y_j \star T^*}^{\bar{B}_1, \bar{B}_2} \bar{V}_r$$
where  $H_{y_j \star T^*}^{\bar{B}_1, \bar{B}_2}$  is the Hankel matrix associated to  $y_j \star T^*$  for  $j = 1, \dots, n_2$ .
6. Compute the eigenvectors  $\mathbf{v}_p$  of  $\sum_{j=1}^{n_2} l_j M_{y_j}^{\bar{V}_r}$  such that  $|l_j| \leq 1, j = 1, \dots, n_2$  and for each  $p = 1, \dots, r$  do the following:
  - The  $y$ 's coordinates of the  $\mathbf{u}_p$  are the eigenvalues of the multiplication matrices by  $y_j$ .  
Use the formula  $M_{y_j}^{\bar{V}_r} \mathbf{v}_p = u_{p,2,j} \mathbf{v}_p$  for  $p = 1, \dots, r$  and  $j = 1, \dots, n_2$  and deduce the  $u_{p,2,j}$ .
  - Write the matrix  $H_{T^*}^{\bar{B}_1, \bar{B}_2}$  in the basis of interpolation polynomials and use the corresponding matrix  $\mathcal{T}$  to compute the  $z$ 's coordinates.  
Divide the  $k^{\text{th}}$  row on the first row of the matrix  $\mathcal{T}$  to obtain the values of  $u_{p,3,k}$  for  $p = 1, \dots, r$  and  $k = 1, \dots, n_3$ .
  - The  $x$ 's coordinates of  $\mathbf{u}_p$  are computed using the eigenvectors of

The cost of the SVD computation is in  $\mathcal{O}(s^3)$  where  $s \geq r$  is the maximal size of the Hankel matrix  $H_0$  and  $r$  the rank of the decomposition. The computation of each multiplication matrix is in  $\mathcal{O}(r^2)$  and the eigencomputation is in  $\mathcal{O}(r^3)$ . This yields a complexity bound in  $\mathcal{O}(s^3 + nr^2)$  for the complete algorithm, where  $n = \max(n_1, n_2, n_3)$  is a bound on the dimension of the spaces. This complexity bound extends to the decomposition of general multi-symmetric tensors, provided  $r = \text{rank } H_0$ .

### 3.6 Example

In this section, we illustrate the decomposition algorithm on a non symmetric tensor of degree one at each bunch of 3 variables and of rank 3 by an example already detailed in [HMK17].

If  $\delta_l = 1$  for all  $l = 1, \dots, k$ ,  $k > 1$  and  $n_l = n$ , let  $k = 3$ ,  $n_l = n = 2$ ,  $r = 3$  and  $\delta_l = 1$  then we have  $\mathbf{x} = (x_0, x_1, x_2)$ ,  $\mathbf{y} = (y_0, y_1, y_2)$  and  $\mathbf{z} = (z_0, z_1, z_2)$ . For

$$\bar{\alpha} \in \mathbb{N}^3, |\bar{\alpha}| = 1 \Rightarrow \bar{\alpha} = (1), (0, 1), (0, 1) \Rightarrow \mathbf{x}^{\bar{\alpha}} = x_i, i = 0, \dots, 2$$

$$\bar{\beta} \in \mathbb{N}^3, |\bar{\beta}| = 1 \Rightarrow \bar{\beta} = (1), (0, 1), (0, 1) \Rightarrow \mathbf{y}^{\bar{\beta}} = y_j, j = 0, \dots, 2$$

$$\bar{\gamma} \in \mathbb{N}^3, |\bar{\gamma}| = 1 \Rightarrow \bar{\gamma} = (1), (0, 1), (0, 1) \Rightarrow \mathbf{z}^{\bar{\gamma}} = z_k, k = 0, \dots, 2$$

The multi symmetric tensor is defined by a multi symmetric array of coefficients such that  $t_{\alpha, \beta, \gamma} := t_{\bar{\alpha}, \bar{\beta}, \bar{\gamma}} = t_{i, j, k}$  then  $T(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{\substack{0 \leq i \leq 2 \\ 0 \leq j \leq 2 \\ 0 \leq k \leq 2}} t_{i, j, k} x_i y_j z_k =$   
 $0.4461x_0y_0z_0 - 0.2262x_0y_0z_1 + 0.4427x_0y_0z_2 - 0.2756x_0y_1z_0 + 0.1612x_0y_1z_1 - 0.3100x_0y_1z_2 -$   
 $0.1209x_0y_2z_0 + 0.1465x_0y_2z_1 - 0.1169x_0y_2z_2 - 0.0123x_1y_0z_0 - 0.0518x_1y_0z_1 + 0.0180x_1y_0z_2 -$   
 $0.0133x_1y_1z_0 + 0.0263x_1y_1z_1 - 0.0259x_1y_1z_2 - 0.3195x_1y_2z_0 + 0.0931x_1y_2z_1 - 0.1116x_1y_2z_2 -$   
 $0.1460x_2y_0z_0 + 0.0655x_2y_0z_1 - 0.1734x_2y_0z_2 + 0.1010x_2y_1z_0 - 0.0574x_2y_1z_1 + 0.1238x_2y_1z_2 -$   
 $0.1485x_2y_2z_0 + 0.0323x_2y_2z_1 - 0.0037x_2y_2z_2.$

Let  $x_0 = y_0 = z_0 = 1$  then  $\underline{T}(\underline{\mathbf{x}}, \underline{\mathbf{y}}, \underline{\mathbf{z}}) = \sum_{\substack{1 \leq i \leq 2 \\ 1 \leq j \leq 2 \\ 1 \leq k \leq 2}} t_{i, j, k} \underline{x}_i \underline{y}_j \underline{z}_k.$

Then the tensor decomposition problem consists in finding the coefficient vectors of linear forms and their corresponding weights such that  $\underline{T}(\underline{\mathbf{x}}, \underline{\mathbf{y}}, \underline{\mathbf{z}}) = \sum_{p=1}^r \omega_p \mathbf{u}_{p,1}(\underline{\mathbf{x}}) \mathbf{u}_{p,2}(\underline{\mathbf{y}}) \mathbf{u}_{p,3}(\underline{\mathbf{z}})$ . Given all the moments of degree at most one at each group of coordinates  $(t_{i, j, k})_{\substack{0 \leq i \leq 2 \\ 0 \leq j \leq 2 \\ 0 \leq k \leq 2}}$ ,

We create two sets  $A_1 = (\mathbf{x}^{\bar{\alpha}}\mathbf{y}^{\bar{\beta}})_{\substack{|\bar{\alpha}|=1 \\ |\bar{\beta}|=1}} = (x_i y_j)_{\substack{0 \leq i \leq 2 \\ 0 \leq j \leq 2}}$  and  $A_2 = (\mathbf{z}^{\bar{\gamma}})_{|\bar{\gamma}|=1} =$

$(z_k)_{0 \leq k \leq 2}$  so that

$A_1 = (x_0 y_0, x_0 y_1, x_0 y_2, x_1 y_0, x_1 y_1, x_1 y_2, x_2 y_0, x_2 y_1, x_2 y_2)$  and  $A_2 = (z_0, z_1, z_2)$ .

For  $x_0 = y_0 = z_0 = 1$  then  $B_1 = (1, y_1, y_2, x_1, x_1 y_1, x_2, x_2 y_1, x_2 y_2)$  and  $B_2 = (1, z_1, z_2)$ , the Hankel matrix associated to the tensor in the monomial basis  $B_1$  and  $B_2$  is

$$H_{T^*}^{B_1, B_2} = [t_{\bar{\alpha} + \bar{\beta} + \bar{\gamma}}]_{\substack{|\bar{\alpha}|=1 \\ |\bar{\beta}|=1 \\ |\bar{\gamma}|=1}} = \begin{bmatrix} 1 & y_1 & y_2 & x_1 & x_1 y_1 & x_1 y_2 & x_2 & x_2 y_1 & x_2 y_2 \\ t_0 & t_1 & t_2 & t_{0,1} & t_{0,2} & t_{1,1} & t_{1,2} & t_{2,1} & t_{2,2} \\ t_{0,1} & t_{1,1} & t_{2,1} & t_{0,1,1} & t_{0,2,1} & t_{1,1,1} & t_{1,2,1} & t_{2,1,1} & t_{2,2,1} \\ t_{0,2} & t_{1,2} & t_{2,2} & t_{1,2} & t_{0,2,2} & t_{1,1,2} & t_{1,2,2} & t_{2,1,2} & t_{2,2,2} \end{bmatrix} \begin{matrix} 1 \\ z_1 \\ z_2 \end{matrix}$$

$$= \begin{bmatrix} 1 & y_1 & y_2 & x_1 & x_1 y_1 & x_1 y_2 & x_2 & x_2 y_1 & x_2 y_2 \\ 0.4461 & 0.2756 & 0.1209 & 0.0123 & 0.0133 & 0.3195 & 0.1460 & 0.1010 & 0.1485 \\ -0.2262 & 0.1612 & 0.1465 & 0.0518 & 0.0263 & 0.0931 & 0.0655 & 0.0574 & 0.0323 \\ 0.4427 & 0.3100 & 0.1169 & .0180 & 0.0259 & 0.1116 & 0.1734 & 0.1238 & 0.0037 \end{bmatrix}$$

All the entries of this matrix are known, we choose  $\bar{B}_1 = \langle 1, x_1, x_2 \rangle$  and  $\bar{B}_2 = \langle 1, z_1, z_2 \rangle$  to be able to multiply by  $y_1$  and to compute the multiplication matrix. Computing the singular value decomposition of  $H_{T^*}^{\bar{B}_1, \bar{B}_2}$ , we obtain

$$H_{T^*}^{\bar{B}_1, \bar{B}_2} = USV^T = \begin{bmatrix} 1 & x_1 & x_2 \\ t_0 & t_1 & t_2 \\ t_{0,1} & t_{1,1} & t_{2,1} \\ t_{0,2} & t_{1,2} & t_{2,2} \end{bmatrix} \begin{matrix} 1 \\ z_1 \\ z_2 \end{matrix} = \begin{bmatrix} 1 & x_1 & x_2 \\ 0.4461 & -0.0123 & -0.1460 \\ -0.2262 & -0.0518 & 0.0655 \\ 0.4427 & 0.0180 & -0.1734 \end{bmatrix}$$

where  $S$  is the diagonal matrix of all singular values of  $H_{T^*}^{\bar{B}_1, \bar{B}_2}$  arranged in a decreasing order,  $U$  is an unitary matrix whose columns are the left singular vectors of  $H_{T^*}^{\bar{B}_1, \bar{B}_2}$ ,  $V$  is an unitary matrix whose columns are the right singular vectors of  $H_{T^*}^{\bar{B}_1, \bar{B}_2}$ . We denote by  $U^H$  the Hermitian transpose of  $U$  and  $\bar{V}$  the conjugate of  $V$ .

Let  $v_i = [v_{\alpha, i}]_{\alpha \in \bar{B}_1}$  and  $w_j = [w_{\beta, j}]_{\beta \in \bar{B}_2}$  be respectively the  $i^{\text{th}}$  and  $j^{\text{th}}$  columns of  $U^H$  and  $\bar{V}$ . We denote by  $v_i(\mathbf{x}) = v_i^T U_r^H$  and  $w_j(\mathbf{z}) = w_j^T \bar{V}_r$  the corresponding polynomials. The bases formed by these first  $r$  polynomials are



denoted  $\mathcal{U}_r^H := (v_i(\mathbf{x}))_{i=1,\dots,r}$  and  $\bar{\mathcal{V}}_r := (w_j(\mathbf{z}))_{j=1,\dots,r}$ . We will also denote by  $U_r^H$  (resp.  $\bar{V}_r$ ) the corresponding coefficient matrix, formed by the first rows (resp. columns) of  $U^H$  (resp.  $\bar{V}$ ). We denote by  $S_r$  the diagonal matrix of the first  $r$  rows and columns of  $S$ , formed by the first  $r$  singular values. To compute the multiplication matrices  $M_{y_1}^{\bar{\mathcal{V}}_r}$  and  $M_{y_2}^{\bar{\mathcal{V}}_r}$  we need to compute the following matrices

$$H_{y_1^* T^*}^{\bar{B}_1, \bar{B}_2} = \begin{array}{c} \begin{array}{ccc} y_1 & y_1 x_1 & x_2 y_1 \\ \begin{bmatrix} t_{0,1} & t_{1,1} & t_{2,1} \\ t_{0,1,1} & t_{1,1,1} & t_{2,1,1} \\ t_{0,1,2} & t_{1,1,2} & t_{2,1,2} \end{bmatrix} & \begin{matrix} 1 \\ z_1 \\ z_2 \end{matrix} \end{array} \\ = \begin{array}{ccc} y_1 & y_1 x_1 & x_2 y_1 \\ \begin{bmatrix} -0.2756 & -0.0133 & 0.1010 \\ 0.1612 & 0.0263 & -0.0574 \\ -0.3100 & -0.0259 & 0.1238 \end{bmatrix} \end{array} \end{array}$$

$$H_{y_2^* T^*}^{\bar{B}_1, \bar{B}_2} = \begin{array}{c} \begin{array}{ccc} y_2 & y_2 x_1 & x_2 y_2 \\ \begin{bmatrix} t_{0,2} & t_{1,2} & t_{2,2} \\ t_{0,2,1} & t_{1,2,1} & t_{2,2,1} \\ t_{0,2,2} & t_{1,2,2} & t_{2,2,2} \end{bmatrix} & \begin{matrix} 1 \\ z_1 \\ z_2 \end{matrix} \end{array} \\ = \begin{array}{ccc} y_2 & y_2 x_1 & x_2 y_2 \\ \begin{bmatrix} -0.1209 & -0.3195 & -0.1485 \\ 0.1465 & 0.09311 & 0.0323 \\ -0.116 & -0.111 & -0.0037 \end{bmatrix} \end{array} \end{array}$$

Then we compute  $M_{y_1}^{\bar{\mathcal{V}}_r} = S_r^{-1} U_r^H H_{y_1^* T^*}^{\bar{B}_1, \bar{B}_2} \bar{V}_r$  and  $M_{y_2}^{\bar{\mathcal{V}}_r} = S_r^{-1} U_r^H H_{y_2^* T^*}^{\bar{B}_1, \bar{B}_2} \bar{V}_r$ , and the eigenvectors  $\mathbf{v}_p$  of  $\sum_{j=1}^2 l_j M_{y_j}^{\bar{\mathcal{V}}_r}$  such that  $|l_j| \leq 1, j = 1, \dots, 2$ . To recover the points  $\mathbf{u}_p \in \mathbb{K}^{n \times k}$  for  $p = 1, \dots, r$  of the form

$$\mathbf{u}_p = \begin{bmatrix} u_{p,1,1} & x_1 \\ u_{p,1,2} & x_2 \\ \\ u_{p,2,1} & y_1 \\ u_{p,2,2} & y_2 \\ \\ u_{p,3,1} & z_1 \\ u_{p,3,2} & z_2 \end{bmatrix}$$

We do the following:

In general we have  $M_{x_j, i_j}^{\bar{\mathcal{V}}_r} \mathbf{v}_i = u_{i,j,i_j} \mathbf{v}_i$ , for  $i = 1, \dots, r, j = 1, \dots, k, i_j = 1, \dots, n_j$ , so in this case we get  $M_{y_1}^{\bar{\mathcal{V}}_r} \mathbf{v}_p = u_{p,2,1} \mathbf{v}_p$  and  $M_{y_2}^{\bar{\mathcal{V}}_r} \mathbf{v}_p = u_{p,2,2} \mathbf{v}_p$  for  $p = 1, \dots, 3$ .

We compute  $u_{p,2,1}$  and  $u_{p,2,2}$  for  $p = 1, \dots, 3$ , so that we get

$$\begin{aligned} u_{p,2,1} &= [-0.7463 \quad -0.2937 \quad -0.3048] \\ u_{p,2,2} &= [1.4032 \quad -0.3363 \quad -3.5903]. \end{aligned}$$

The eigenvectors  $\mathbf{v}_p \in \langle 1, x_1, x_2 \rangle$  for  $p = 1, \dots, 3$  are up to a scalar the interpolation polynomials at the roots so that if the dual of the tensor has an affine decomposition  $\underline{T}^*(\underline{\mathbf{x}}, \underline{\mathbf{y}}, \underline{\mathbf{z}}) = \sum_{p=1}^r \omega_p \mathbf{e}_{\mathbf{u}_p}(\underline{\mathbf{x}}, \underline{\mathbf{y}}, \underline{\mathbf{z}})$  then  $\underline{T}^*(\mathbf{v}_p) = \sum_{p=1}^r \omega_p \mathbf{e}_{\mathbf{u}_p}(\mathbf{v}_p) = \lambda_p \omega_p$ ,  $\underline{T}^*(z_1 \mathbf{v}_p) = \sum_{p=1}^r \omega_p \mathbf{e}_{\mathbf{u}_p}(z_1 \mathbf{v}_p) = \lambda_p \omega_p u_{p,3,1}$  and  $\underline{T}^*(z_2 \mathbf{v}_p) = \sum_{p=1}^r \omega_p \mathbf{e}_{\mathbf{u}_p}(z_2 \mathbf{v}_p) = \lambda_p \omega_p u_{p,3,2}$ , for  $p = 1, \dots, 3$ . Then the values of  $u_{p,3,1}$  and  $u_{p,3,2}$  for  $p = 1, \dots, 3$  come from the computation of the matrix:

$$\mathcal{T} = \begin{array}{c} \begin{array}{ccc} v_1 & v_2 & v_3 \end{array} \\ \begin{bmatrix} T^*(v_1) & T^*(v_2) & T^*(v_3) \\ T^*(z_1 v_1) & T^*(z_1 v_2) & T^*(z_1 v_3) \\ T^*(z_2 v_1) & T^*(z_2 v_2) & T^*(z_2 v_3) \end{bmatrix} \begin{array}{l} 1 \\ z_1 \\ z_2 \end{array} \end{array} = \begin{array}{c} \begin{array}{ccc} v_1 & v_2 & v_3 \end{array} \\ \begin{bmatrix} \lambda_1 \omega_1 & \lambda_2 \omega_2 & \lambda_3 \omega_3 \\ \lambda_1 \omega_1 u_{1,3,1} & \lambda_2 \omega_2 u_{2,3,1} & \lambda_3 \omega_3 u_{3,3,1} \\ \lambda_1 \omega_1 u_{1,3,2} & \lambda_2 \omega_2 u_{2,3,2} & \lambda_3 \omega_3 u_{3,3,2} \end{bmatrix} \begin{array}{l} 1 \\ z_1 \\ z_2 \end{array} \end{array}$$

Therefore the value of  $u_{p,3,1}$  (resp.  $u_{p,3,2}$ ) comes from the ratio of the second row (resp. the third row) and the first row of the matrix for  $p = 1, \dots, 3$ . So that we get

$$\begin{aligned} u_{p,3,1} &= [-0.6558 \quad 0.0321 \quad -0.5209] \\ u_{p,3,2} &= [1.2474 \quad 0.4035 \quad 0.2427]. \end{aligned}$$

The common eigenvectors of all  $(M_{y_j}^{\bar{V}_r})^\top$  -are up to scalar- the evaluations. They are represented by vectors of the form  $\mathbf{v}_p^* = \mu_p [1, u_{p,1,1}, u_{p,1,2}]$  in the dual basis of  $\bar{B}_1 = \langle 1, x_1, x_2 \rangle$  then the computation of the coordinates of  $u_{p,1,1}$  and  $u_{p,1,2}$  come from the eigenvectors of the transpose of the multiplication operators which are obtained by transposing the inverse of the matrix  $V$  of vectors of  $M_{y_j}^{\bar{V}_r}$  for  $j = 1, \dots, 2$ , therefore the value of  $u_{p,1,1}$  (resp.  $u_{p,1,2}$ ) comes from the ratio of the second element of  $\mathbf{v}_p^*$  (resp. the third element) and the first element of it, so that

$$\begin{aligned} u_{p,1,1} &= [0.1142 \quad -1.0860 \quad 1.2381] \\ u_{p,1,2} &= [-0.4057 \quad -0.5676 \quad 0.8734]. \end{aligned}$$

Notice that the computation of  $\omega_p, p = 1, \dots, 3$  can be done using the following formula  $\omega_p = \frac{\langle T^* | \mathbf{v}_p \rangle}{\mathbf{v}_p(\mathbf{u}_p)}$  since if  $\mathbf{v}_p \in \langle 1, x_1, x_2 \rangle$  then  $\mathbf{v}_p = a_p + x_1 b_p + x_2 c_p$  and  $\mathbf{v}_p^* = \mu_p[1, u_{p,1,1}, u_{p,1,2}] \in (\langle 1, x_1, x_2 \rangle)^*$ , so that  $\mathbf{v}_p(\mathbf{u}_p) = a_p + u_{p,1,1} b_p + u_{p,1,2} c_p = \langle \mathbf{v}_p | \mathbf{v}_p^* \rangle$ , the computation gives

$$\omega = (\omega_p)_{1 \leq p \leq r} = [0.3185 \quad 0.0889 \quad 0.0386]$$

# Chapter 4

## Tensors in Learning Latent Variable Models

In this chapter, we describe an application of a statistical model associated to symmetric tensor decomposition problem. Given a corpus of  $r$  distinct topics, such that each one of them consists of  $l$  words with associated vocabulary of  $n$  words which is modeled by the canonical basis of  $\mathbb{R}^n$ . We show how to recover the vector probabilities of a fixed collection of three words given a topic with the probability of this topic using given number of moments. We recover a number of topics in the corpus from the numerical rank of the invertible truncated Hankel matrix associated to the tensor.

We recall some basic probability definitions which are used after to model the statistical problem.

### 4.1 Discret Distributions

**Definition 4.1.1.** *A random discrete variable denoted by a capital letter  $X$  is a variable whose value is a numerical outcome of a phenomenon which is obtained by counting. A random continuous variable  $X$  is a variable whose value is obtained by measuring and lie in an interval.*

Let  $X = (x_1, x_2, \dots, x_r)$  a random discrete variable with the probability distribution  $P(X = x_i) = p_i$  for  $i = 1, \dots, r$ .

**Definition 4.1.2.** *The mean of  $X$  denoted by  $\mu_X$ , is the weighted sum of the*

values of  $X$  by their associated probabilities such that

$$\mu_X = \sum_{i=1}^r x_i p_i.$$

It is also called the expected value of  $X$ .

**Definition 4.1.3.** The variance of  $X$  denoted by  $\sigma_X^2$ , is defined as follows:

$$\sigma_X^2 = \sum_{i=1}^r (x_i - \mu_X)^2 p_i.$$

The square root of the variance is called standard deviation of  $X$  and it is denoted by  $\sigma_X$ .

**Definition 4.1.4.** The joint probability distribution of a  $l$  random discrete variables

$X_1, X_2, \dots, X_l$  is defined by  $P(X_1 = x_1, X_2 = x_2, \dots, X_l = x_l) = P(X_2 = x_2/X_1 = x_1) \times P(X_3 = x_3/X_2 = x_2, X_1 = x_1) \times \dots \times P(X_l = x_l/X_1 = x_1, X_2 = x_2, \dots, X_{l-1} = x_{l-1})$ .

**Definition 4.1.5.** The random discrete variables  $X_1, X_2, \dots, X_l$  are exchangeable for any permutation of indices  $1, 2, 3, \dots$  if the joint probability distribution is the same for any permutation of the original sequence of indices i.e.,  $P(X_{\sigma(1)} = x_1, X_{\sigma(2)} = x_2, \dots, X_{\sigma(l)} = x_l) = P(X_1 = x_1, X_2 = x_2, \dots, X_l = x_l)$

We deduce later from this definition, that the future observations behave like earlier ones in LDA statistical model.

**Definition 4.1.6.** The random discrete variables  $X_1, X_2, \dots, X_l$  are called identically independent which are denoted by *i.i.d.* if they have the same probability distribution and they are mutually independent.

**Theorem 4.1.7.** [Aus08] Given a latent distribution form  $H$ , a sequence of exchangeable random variables  $X_1, X_2, \dots, X_l$  can be seen as a mixture of conditionally identically independent variables.

## 4.2 Continuous Distributions

**Definition 4.2.1.** A discrete probability distribution of a random discrete variable  $X$  assigns a probability to each value that the variable can take. It is impossible to write down the discrete probability distribution of a continuous random variable  $X$ . However, we define the cumulative distribution function  $F(X)$  which gives the probability of  $X$  taking a value which is equal or less than  $x$  such that  $F(x) = P(X \leq x)$ .

**Proposition 4.2.2.** The cumulative distribution function  $F(x)$  satisfies the following properties:

- $F(-\infty) = 0$ ;
- $F(\infty) = 1$ ;
- $F(a) \leq F(b)$  if  $a \leq b$ .

We deduce that  $P(a < x \leq b) = F(b) - F(a)$ .

**Definition 4.2.3.** A random continuous variable  $X$  has a probability density function  $p(x)$  defined on an interval  $[a, b]$  as follows:

$$p(x) = \frac{d}{dx} F(x) = F'(x).$$

so that if  $F(x)$  is the cumulative distribution function of  $X$  then

$$F(x) = \int_{-\infty}^x p(t) dt.$$

and the continuous probability distribution that the variable lies in the interval  $[a, b]$  is computed using the probability density function such that

$$P(a < x \leq b) = \int_a^b p(x) dx.$$

Let  $X$  be a random continuous variable with range  $[a, b]$  and probability density function  $p(x)$ .

**Definition 4.2.4.** The mean or average of  $X$  denoted by  $\mu_X$  is defined by

$$\mu_X = \int_a^b xp(x) dx.$$

It is also called the expected value of  $X$ .

**Definition 4.2.5.** The variance of  $X$  denoted by  $\sigma_X^2$  is defined by

$$\sigma_X^2 = \int_a^b x^2 p(x) dx - \left( \int_a^b x p(x) dx \right)^2.$$

The standard deviation of  $X$  is the square root of  $X$  denoted by  $\sigma_X$ .

The properties of mean and variance of  $X$  for random continuous variables are the same as the discrete ones. If  $X, Y$  are two random variables on a sample  $\Omega$ ,  $c, d$  are two constants then:

- $\mu_{X+Y} = \mu_X + \mu_Y$ .
- $\mu_{cX+d} = c\mu_X + d$ .
- If  $X$  and  $Y$  are independent then  $\sigma_{X+Y}^2 = \sigma_X^2 + \sigma_Y^2$ .
- $\sigma_{cX+d}^2 = c\sigma_X^2 + d$ .

If  $g(x)$  is a function defined on  $\mathbb{R}$  then  $Y = g(X)$  is a random variable and  $\mu_Y = \mu_{g(X)} = \int_{-\infty}^{\infty} g(x)p(x)dx$ .

**Definition 4.2.6.** The joint probability distribution of  $r$  random continuous variables

$X_1, X_2, \dots, X_r$  is defined by its probability density function  $p_{X_1, X_2, \dots, X_r}(x_1, x_2, \dots, x_r) = p_{X_2|X_1}(x_2 | x_1) \times p_{X_3|X_2, X_1}(x_3 | x_2, x_1) \times p_{X_r|X_2, \dots, X_{r-1}}(x_r | x_1, x_2, \dots, x_{r-1})$  where  $p_{X_i|X_2, \dots, X_{i-1}}(x_i | x_1, x_2, \dots, x_{i-1})$  is the conditional probability distribution of  $X_i$  given  $(X_1 = x_1, X_2 = x_2, \dots, X_{i-1} = x_{i-1})$  if  $p(x_1, x_2, \dots, x_{i-1}) > 0$  for  $i = 1, \dots, r$ .

**Definition 4.2.7.** The covariance between two real-valued random continuous variables  $X_i$  and  $X_j$  for  $i, j = 1, \dots, r$  denoted by  $\sigma_{X_i, X_j}$  or  $cov(X_i, X_j)$  is the expected value of the product of deviations of the variables from their means such that

$$\sigma_{X_i, X_j} = cov(X_i, X_j) := \mu_{(X_i - \mu_{X_i})(X_j - \mu_{X_j})}$$

is the mean of the product.

**Definition 4.2.8.** The correlation coefficient between two random variables denoted by  $\rho_{X_i, X_j}$  for  $i, j = 1, \dots, r$  is equal to

$$\rho_{X_i, X_j} = \frac{\sigma_{X_i, X_j}}{\sqrt{\sigma_{X_i, X_i} \sigma_{X_j, X_j}}} \text{ for } i, j = 1, \dots, r.$$

**Definition 4.2.9.** *The covariance matrix denoted by  $\Sigma$  measures the deviation of the random variable vector  $\mathbf{X} = (X_1, X_2, \dots, X_r)^\top$  from the mean vector  $\mu_{\mathbf{X}} = (\mu_{X_1}, \mu_{X_2}, \dots, \mu_{X_r})^\top$  such that*

$$\Sigma = \mu_{(\mathbf{X}-\mu)(\mathbf{X}-\mu)^\top} = (\sigma_{X_i, X_j})_{i,j=1, \dots, r}.$$

The leading diagonal of the covariance matrix contains the variances and the off-diagonal elements describe the correlations between the variables.

**Proposition 4.2.10.** *The covariance matrix of a random vector  $\mathbf{X}$  is a symmetric positive semi definite matrix.*

**Definition 4.2.11.** *The covariance matrix is said to be a spherical if all the variances are equal and the covariances are null such that*

$$\Sigma = \sigma^2 I.$$

#### 4.2.1 Multivariate Gaussian Distribution

Let  $X$  be a random Gaussian continuous variable described by two parameters  $\mu$  and  $\sigma^2$ .

The parameters  $\mu$  and  $\sigma^2$  in machine learning models can be estimated such that they maximize the likelihood of the statistical model generating the training data, defined as follows:

$$\mathfrak{M} = \{\mathcal{X}, \mathbb{P}_\theta, \Theta\}$$

where  $\mathcal{X} = \mathbb{R}$  is the state space of realizations,  $\mathbb{P}_\theta = \mathcal{N}(\mu, \sigma)$  is the Gaussian distribution described hereafter and  $\Theta = \mathbb{R} \times \mathbb{R}^+$  is the parameter set.

So that

$$\hat{\mu} = \frac{1}{r} \sum_{i=1}^N x_i,$$

$$\hat{\sigma}^2 = \frac{1}{k} \sum_{i=1}^N (x_i - \hat{\mu})^2.$$

where  $x_i$  is the observation of the  $i^{th}$  sample and  $N$  is the number of samples.



**Definition 4.2.12.** *The univariate dimensional Gaussian Distribution of a random variable  $X$  is a continuous probability distribution defined by its probability density function as follows:*

$$p(x | \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-(x - \mu)^2}{2\sigma^2}\right).$$

We can extend the definition of univariate Gaussian distribution to the multivariate Gaussian distribution as follows:

**Definition 4.2.13.** *The multivariate dimensional Gaussian Distribution of a random variable vector  $\mathbf{X} = (X_1, X_2, \dots, X_r)^\top$  with the mean vector  $\mu_{\mathbf{X}} = (\mu_{X_1}, \mu_{X_2}, \dots, \mu_{X_r})^\top$  and the associated covariance matrix  $\Sigma$  is a continuous probability distribution defined by its probability density function as follows:*

$$p(\mathbf{x} | \mu_{\mathbf{X}}, \Sigma) = \frac{1}{(2\pi)^{\frac{r}{2}} \Sigma^{\frac{r}{2}}} \exp\left(\frac{-1}{2} (\mathbf{x} - \mu_{\mathbf{X}})^\top \Sigma^{-1} (\mathbf{x} - \mu_{\mathbf{X}})\right).$$

## 4.2.2 Dirichlet Distribution

**Definition 4.2.14.** *The univariate Beta distribution of a random continuous variable  $X$  with support  $[0, 1]$  and shape positive parameters  $\alpha, \beta$  is defined by its probability density function as the power function of  $X$  and its reflection  $1 - X$  such that:*

$$p(x|\alpha, \beta) = \frac{1}{\mathbf{B}(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1} = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1} (1-x)^{\beta-1}.$$

where  $\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx$  is the Gamma function.

We can extend the definition of univariate Beta distribution to the Dirichlet distribution as follows:

**Definition 4.2.15.** *The multivariate Dirichlet distribution of a random continuous variable*

$$\mathbf{X} = (X_1, X_2, \dots, X_r)^\top$$

with support the simplex

$$S = \{x_i \in \mathbb{R} \mid \sum_{i=1}^r x_i = 1, x_i \geq 0\}$$

and shape positive parameters vector  $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_r)$  is defined by its probability density function as follows:

$$p(\mathbf{x}|\alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^r x_i^{\alpha_i-1} = \frac{\Gamma(\sum_{i=1}^r \alpha_i)}{\prod_{i=1}^r \Gamma(\alpha_i)} \prod_{i=1}^r x_i^{\alpha_i-1}.$$

where  $B(\alpha)$  is said to be the Beta function.

## 4.3 Learning Latent Variable Models

### 4.3.1 Exchangeable Single Topic Model

We denote by  $H$  the only topic of each document, it is modeled by a random latent discrete variable  $H$  which takes only a finite number  $r$  of distinct positive integer values.

The probability of each topic in the corpus is modeled by the weight of the corresponding probability vector in the decomposed tensor such that  $P(H = p) = \omega_p$  for  $p = 1, \dots, r$ .

This topic consists of  $l$  words  $x_1, x_2, \dots, x_l$  which are values of random discrete i.d.d. variables  $X_1, X_2, \dots, X_l$ .

The discrete conditional probability distribution of each one of the variables  $X_1, X_2, \dots, X_l$  is given by the probability vector defined by  $P(x_j = i/H = p) = \xi_{p,i}$  for  $j = 1, \dots, l$ ,  $p = 1, \dots, r$  and  $i = 1, \dots, n$ . The probability vectors correspond to the points of the decomposed compressed tensor of the statistical model which describes the corpus content represented by three slices of a tensor.

The probability to draw the  $l$  words from a fixed document  $t$  in the increasing order from  $x_1$  to  $x_l$  depends on the conditional probability of each one of them as they are conditionally independent. Each word can be drawn from the canonical basis  $e_1, e_2, \dots, e_n$  of  $\mathbb{R}^n$ .

The discrete conditional probability of  $x_j$  given a topic  $H = p$  is simply  $\xi_p$ , i.e.  $P(x_j/H = p) = \sum_{i=1}^n P(j^{th}word = i/H = p)e_i = \sum_{i=1}^n \xi_{p,i}e_i = \xi_p$  for  $p = 1, \dots, r$ .

The cross conditional probability of  $n$ -dimensional random vectors  $x_1, x_2, \dots, x_l$  given a topic  $H = p$  is then  $P(x_1 \otimes x_2 \dots \otimes x_l/H = p) = \xi_p \otimes \xi_p \dots \otimes \xi_p$  for  $p = 1, \dots, r$ .

For instance, the cross conditional probability of  $x_1$  and  $x_2$  given a topic  $H = p$  is then  $P(x_1 \otimes x_2 / H = p) = \xi_p \otimes \xi_p$  for  $p = 1, \dots, r$ . This leads to the following theorem developed by (Anandkumar and al.) in [AHK12]:

**Theorem 4.3.1.** *Let  $x_1$  and  $x_2$  are the realizations of two random variables  $X_1$  and  $X_2$  which describe the first two words in a document,*

$$M_2 := P(x_1 \otimes x_2)$$

and

$$M_3 := P(x_1 \otimes x_2 \otimes x_3)$$

then

$$M_2 := \sum_{p=1}^r \omega_p \xi_p \otimes \xi_p$$

and

$$M_3 := \sum_{p=1}^r \omega_p \xi_p \otimes \xi_p \otimes \xi_p$$

where  $\xi_p = (\xi_{p,1}, \dots, \xi_{p,n}) \in \mathbb{R}^n$ .

So that, the probability vectors of topics can be computed using the minimal symmetric tensor decomposition using the cross probabilities of any two and three words in a document.

### 4.3.2 Spherical Gaussian Mixtures

#### Common Covariance

We either denote by  $H$  the random discrete variable which describes the weight of each topic in the corpus with discrete probability distribution defined as follows:

$$P(H = p) = \omega_p$$

for  $p = 1, \dots, r$ .

Each topic is described by the Gaussian random vector  $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$  with its probability density function defined as follows

$$f(\mathbf{x} | \xi_p, \Sigma_p) = \frac{1}{(2\pi)^{\frac{n}{2}} \Sigma_p^{\frac{n}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \xi_p)^\top \Sigma_p^{-1}(\mathbf{x} - \xi_p)\right) \text{ for } p = 1, \dots, r.$$

where the mean vector is equal to

$$\xi_p = (\xi_{p,1}, \xi_{p,2}, \dots, \xi_{p,n})^\top \in \mathbb{R}^n \text{ for } p = 1, \dots, r.$$

and  $\Sigma_p$  is a spherical semi-definite positive covariance matrix

$$\Sigma_p = \sigma_p^2 I \text{ for } p = 1, \dots, r..$$

We suppose that all topics has the same common covariance matrix such that

$$\Sigma = \Sigma_p = \sigma_p^2 I = \sigma^2 I \text{ for } p = 1, \dots, r.$$

The mixed model is observed as follows

$$\mathbf{x} = \xi_h + \mathbf{z}$$

where  $\mathbf{z} \sim \mathcal{N}(0, \sigma^2 I)$  is an independent Gaussian random vector with zero mean and the same spherical covariance  $\sigma^2 I$  of  $\mathbf{x}$ .

The following theorem developed in [HK13] shows how to find out the symmetric structure of associated tensor using the manipulated cross moments  $M_2$  and  $M_3$ :

**Theorem 4.3.2.** *Let  $\mathbf{x} \in \mathbb{R}^n$  be a Gaussian random vector which describes each topic in the corpus such that all of them have the same variance  $\sigma^2$ , if  $n \geq r$  and the variance  $\sigma^2$  is the smallest eigenvalue of the covariance matrix  $P(\mathbf{x} \otimes \mathbf{x}) - P(\mathbf{x}) \otimes P(\mathbf{x})$ . Furthermore, if*

$$M_2 := P(\mathbf{x} \otimes \mathbf{x}) - \sigma^2 I$$

and

$$M_3 := P(\mathbf{x} \otimes \mathbf{x} \otimes \mathbf{x}) - \sigma^2 \sum_{i=1}^n P(\mathbf{x}) \otimes e_i \otimes e_i + e_i \otimes P(\mathbf{x}) \otimes e_i + e_i \otimes e_i \otimes P(\mathbf{x})$$

then

$$M_2 := \sum_{p=1}^r \omega_p \xi_p \otimes \xi_p$$

and

$$M_3 := \sum_{p=1}^r \omega_p \xi_p \otimes \xi_p \otimes \xi_p$$

where  $\xi_p = (\xi_{p,1}, \dots, \xi_{p,n}) \in \mathbb{R}^n$ .

For more details, see [AGH<sup>+</sup>15].

## Different Covariances

In this case, each Gaussian random vector  $\mathbf{x}$  has different variance  $\sigma_p^2$  and  $\mathbf{z} \sim \mathcal{N}(0, \sigma^2 I)$  is a Gaussian random vector with zero mean and different spherical covariance  $\sigma_q^2 I$  which depends on the choice of the topic  $H = q$ . The model is again

$$\mathbf{x} = \xi_h + \mathbf{z}.$$

But the tensor structure is extracted using the following theorem:

**Theorem 4.3.3.** *Let  $\mathbf{x} \in \mathbb{R}^n$  be a Gaussian random vector which describes each topic in the corpus such that each one of them has different variance  $\sigma_p^2$  for  $p = 1, \dots, r$ , if  $n \geq r$  and the average variance  $\bar{\sigma}^2 = \sum_{p=1}^r \omega_p \sigma_p^2$  is the smallest eigenvalue of the covariance matrix  $P(\mathbf{x} \otimes \mathbf{x}) - P(\mathbf{x}) \otimes P(\mathbf{x})$ . Let  $v$  be any normalized eigenvector corresponding to the eigenvalue  $\bar{\sigma}^2$ . Furthermore, if*

$$M_1 := P(\mathbf{x}(v^\top(\mathbf{x} - P(\mathbf{x})))^2),$$

$$M_2 := P(\mathbf{x} \otimes \mathbf{x}) - \bar{\sigma}^2 I$$

and

$$M_3 := P(\mathbf{x} \otimes \mathbf{x} \otimes \mathbf{x}) - \sigma^2 \sum_{i=1}^n M_1 \otimes e_i \otimes e_i + e_i \otimes M_1 \otimes e_i + e_i \otimes e_i \otimes M_1$$

then

$$M_2 := \sum_{p=1}^r \omega_p \xi_p \otimes \xi_p$$

and

$$M_3 := \sum_{p=1}^r \omega_p \xi_p \otimes \xi_p \otimes \xi_p$$

where  $\xi_p = (\xi_{p,1}, \dots, \xi_{p,n}) \in \mathbb{R}^n$ .

For more details, see [HK13].

## 4.4 Latent Dirichlet Allocation

In Latent Dirichlet Allocation model denoted by LDA model, each document corresponds to a mixture of topics. Furthermore each word may belong to several topics.

The random continuous variable  $H = (H_1, H_2, \dots, H_r)^\top$  which describes the topic mixture in the corpus follows a Dirichlet distribution  $Dir(\alpha)$  with shape positive parameters vector  $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_r)$  defined by its probability density function as follows

$$p(H|\alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^r H_i^{\alpha_i-1} = \frac{\Gamma(\sum_{i=1}^r \alpha_i)}{\prod_{i=1}^r \Gamma(\alpha_i)} \prod_{i=1}^r H_i^{\alpha_i-1}.$$

with support the simplex  $S = \{H_i \in \mathbb{R} \mid \sum_{i=1}^r H_i = 1, H_i \geq 0\}$  and  $B(\alpha)$  the Beta function. For each word  $x_1, x_2, \dots, x_l$  we independently draw a single topic  $j$  with random probability vector  $\xi_j$  and then draw the word according to the probability vector. We encode a word  $x_t$  by setting  $x_t = e_i$  iff the  $t^{\text{th}}$  word in the document is  $i$ . For example, the word *government* may belongs to the topic *politics* and topic *organization* at the same time, so we first select the topic *politics* and then we compute the probability that this word belongs to the topic *politics*.

We denote by  $\alpha_0 = \sum_{i=1}^r \alpha_i$ . We are interested in the case of  $\alpha_0$  is small which corresponds to  $r$  small. The following theorem proposed by [AFH<sup>+</sup>12] helps us to compute the symmetric tensor decomposition using the manipulated cross moments.

**Theorem 4.4.1.** *Let  $x_1, x_2, x_3$  be the first three words of a corpus independently drawn from a discrete distribution specified by the  $\sum_{i=1}^r H_i \xi_i$ . If*

$$M_1 := P(x_1) \tag{4.1}$$

$$M_2 := P(x_1 \otimes x_2) - \frac{\alpha_0}{\alpha_0 + 1} M_1 \otimes M_1 \tag{4.2}$$

and

$$\begin{aligned} M_3 = & P(x_1 \otimes x_2 \otimes x_3) \\ & - \frac{\alpha_0}{\alpha_0 + 2} (P(M_1 \otimes x_1 \otimes x_2) + P(x_1 \otimes M_1 \otimes x_2) + P(x_1 \otimes x_2 \otimes M_1)) \\ & + \frac{2\alpha_0^2}{(\alpha_0 + 2)(\alpha_0 + 1)} M_1 \otimes M_1 \otimes M_1 \end{aligned} \tag{4.3}$$

then

$$M_2 := \sum_{p=1}^r \frac{\alpha_p}{(\alpha_0 + 1)(\alpha_0)} \xi_p \otimes \xi_p$$

and

$$M_3 := \sum_{p=1}^r \frac{2\alpha_p}{(\alpha_0 + 2)(\alpha_0 + 1)\alpha_0} \xi_p \otimes \xi_p \otimes \xi_p$$

where  $\xi_p = (\xi_{p,1}, \dots, \xi_{p,n}) \in \mathbb{R}^n$ .

## 4.5 Numerical Results

In this section, we apply the method of moments on a corpus of 10000 documents followed by a whitening step as detailed hereafter and tensor decomposition process to learn mixed membership topic model(LDA model). Each document consists of a different number of words where each word is drawn from a vocabulary of 100 words.

A word may belongs to any subset of 3 topics, so that we define the latent topic probability vector  $H = (H_1, H_2, \dots, H_r)^\top$  which describes the distribution of topics over the corpus and follows a Dirichlet distribution  $Dir(\alpha) = Dir(\alpha_1, \alpha_2, \dots, \alpha_r)$ . We also define the parameter  $\alpha_0 = \sum_{i=1}^r \alpha_i$  which controls to what extend the membership are mixed.

For each document, we draw the topic according to the distribution specified by  $H$  and then we draw the word according to the conditional distribution associated to the chosen topic. Each word of the vocabulary is represented by the vector  $e_i \in \mathbb{R}^n$  of the canonical basis. We use the formulas of  $M_1, M_2$  and  $M_3$  proposed in Theorem 4.4.1 to compute the first, second and third order moments of the model. The third order moment  $M_3 \in \mathbb{R}^{100 \times 100 \times 100}$  is high dimensional symmetric tensor hard to compute and memorize. We approximate a whitening matrix  $W \in \mathbb{R}^{100 \times 3}$  from the second order moment matrix  $M_2$  by computing the singular value decomposition of  $M_2 = USU^\top$  and we suppose that  $W = -US^{-\frac{1}{2}}$ . Using the whitening matrix  $W$ , the reduced third order moment  $T(W, W, W) \in \mathbb{R}^{3 \times 3 \times 3}$  (easy to store in memory) is computed implicitly with the expression of  $M_3$  in Theorem 4.4.1. The whitening step leads to the following symmetric tensor represented by its 3 slices:

$$T[:, :, 1] = \begin{bmatrix} 1.0153 & -0.0094 & -0.0071 \\ -0.0094 & 1.0078 & -0.0054 \\ -0.0071 & -0.0054 & 1.0209 \end{bmatrix}$$

$$T[:, :, 2] = \begin{bmatrix} -0.0094 & 1.0078 & -0.0054 \\ 1.0078 & 0.1942 & 0.6905 \\ -0.0054 & 0.6905 & -0.1454 \end{bmatrix}$$

$$T[:, :, 3] = \begin{bmatrix} -0.0071 & -0.0054 & 1.0209 \\ -0.0054 & 0.6905 & -0.1454 \\ 1.0209 & -0.1454 & -0.7283 \end{bmatrix}$$

We apply the low rank symmetric tensor decomposition method described in 7 to compute the frequencies which represent the probability vectors of all topics and the weights in the corpus.

This leads to the following decomposition  $T' = \sum_{p=1}^r \omega_p (\xi_p)^3$  where

$$\xi = (\xi_1, \xi_2, \xi_3) = \begin{bmatrix} 1.0 & 1.0 & 1.0 \\ -0.0936 & -1.1394 & 1.3000 \\ -1.4190 & 0.7793 & 0.6183 \\ 1.0 & 1.0 & 1.0 \\ -0.0936 & -1.1394 & 1.3000 \\ -1.4199 & 0.7790 & 0.6182 \\ 1.0 & 1.0 & 1.0 \\ -0.0936 & -1.1394 & 1.3000 \\ -1.4190 & 0.7793 & 0.6183 \end{bmatrix}$$

$$\omega = (\omega_1, \omega_2, \omega_3) = \begin{bmatrix} 0.3394 \\ 0.3510 \\ 0.3248 \end{bmatrix}$$

With the frequencies and weights, we recover the tensor  $T'$  using the same formula  $T' = \sum_{p=1}^r \omega_p (\xi_p)^3$  and we compare it to the reduced tensor  $T$  by computing the norm  $\|T - T'\|_2$  which is equal to 0.0008.

We are now able to compute the weight of each of 3 topics in all documents using a projection step. If  $X \in \mathbb{R}^{100}$  is the vector representing the recurrence of 100 words in a single document, then  $X' = W^\top X \in \mathbb{R}^3$  represents the recurrence vector of words in the basis  $\xi$  of frequencies.

Using the formula

$$X'_\xi = (\xi[1 : 3, 1 : 3])^{-1} X'$$



we are allowed to compute the weights  $\delta_p$  of each topic in a fixed document such that  $X' = \sum_{p=1}^r \delta_p (\xi_p)^3$  for  $p = 1, \dots, 3$ .

## Chapter 5

# Tensor in Fiber Crossing Detection Model

The white matter is found in the deeper tissues of brain whose role is to protect tissues' cells called neurons from injuries.

Each neuron is extended on its extremities by intersected fibers called axons. This white matter helps also to transmit the electrical signals in the brain. Fibers connect nerves among each other, process and store informations which are responsible of daily jobs body such as remembering, thinking or eating. An anomaly which may affect the structure of these fibers, prevents this factory of running well. This is named Alzheimer disease. In this section, we are interested of studying the structure of intersected fibers in human brain.

Basser and al. introduced in [BML94] the first model of fibers' structure called DTI model, in order to reconstruct the shape of the collection of fibers running in parallel close together. The algorithms based on this model supposed that we have only one fiber in a voxel of 3-dimensional space. So we could not model the net of fibers when more than one fiber intersect.

Other methods have been proposed after, such as SHOT to estimate the orientations of underlying fibers. They are characterized by a high angular resolution. They allow to estimate the Diffusion Orientation Distribution Function **ODF** whose maxima are aligned with orientation of fibers.

In [JGJJ11], they proposed the rank one high order tensor decomposition technique to reconstruct the orientation of one fiber which is the only one maxima of the **ODF** function.

In [Bro97], they proposed the low rank approximation technique so called CP method described in Chapter 3 in order to reconstruct the orientation of underlying fibers. In this case, the number of intersected fibers need to be known a priori. This method uses the ALS algorithm described in Section 3.3.2 whose convergence is not guaranteed and relies on initial choices.

## 5.1 Extracting Coefficients of Fiber Crossing Tensor

In this section, we describe a technique which extracts the coefficients of high order low rank symmetric tensor. This technique, described in more details in [MKBD14] relies on the computation of signals which come from the convolution of the so called Watson function [TCC07, WBA12] and **CT-FOD** function.

**CT-FOD** function  $F$  is modeled by a positive semi-definite symmetric high order tensor of dimension 3.

**Definition 5.1.1.** *Watson function is defined as follows:*

$$\Psi(\mathbf{y}_0, \lambda_0, \mathbf{x}) = e^{-\lambda_0 D(\mathbf{y}_0^T \mathbf{x})^2}.$$

where

- $\mathbf{x} = (x_1, x_2, x_3)$  is the vector normalized on the unit sphere.
- $\mathbf{y}_0 = (y_{0,1}, y_{0,2}, y_{0,3})$  is the gradient of magnetic field in three dimensional space.
- $\lambda_0$  is the weight.
- $D$  is the diffusivity coefficient.

**Definition 5.1.2.** *The signal  $S(\mathbf{y}, \lambda) = (S(\mathbf{y}_i, \lambda_i))_{1 \leq i \leq r}$  where  $r$  is the number of intersected fibers in a voxel in three dimensional space is given by the convolution of Watson function and **CT-FOD** function:*

$$S(\mathbf{y}, \lambda) = \Psi(\mathbf{y}, \lambda) \otimes F(\mathbf{y}, \lambda).$$

which is equal to

$$S(\mathbf{y}, \lambda) = \int_{i=1}^r \Psi(\mathbf{y}_i, \lambda_i, \mathbf{x}) F(\mathbf{x}) dx.$$

The following algorithm described in [MKBD14] shows us how to extract the coefficients  $t_{\alpha,\beta,\gamma}$  of positive definite symmetric high order tensor  $T$  which describes the structure of intersected fibers in human brain. Diffusion signal:

$$S(\xi_i, \omega_i) = \int_{S_2} \Psi(\xi_i, \omega_i, \mathbf{x}) T(\mathbf{x}) d\mathbf{x}. \quad (5.1)$$

where  $\Psi(\xi, \omega, \mathbf{x}') = S_0 e^{-wD(\xi^T \mathbf{x}')^2}$  is a Watson Function such that  $D$  is the Diffusion Coefficient and  $T$  is the 4-order Cartesian Positive Definite Symmetric Tensor.

---

**Algorithm 8** : Extraction of tensor coefficients using **ODF** function.

---

**Input** : The Diffusion Signal  $S(\mathbf{y}_i, \lambda_i)$  and the spherical coordinates  $\mathbf{x} = (x_1, x_2, x_3)$ .

**Output** : The coefficients  $t_{\alpha,\beta,\gamma}$  of the tensor  $T$

1. Modeling of the Fibers Orientation Distribution Function **ODF** by a tensor  $T$  of degree  $d$  and dimension 3.

$$T(\mathbf{y}) = \sum_{\alpha+\beta+\gamma=d} t_{\alpha,\beta,\gamma} y_1^\alpha y_2^\beta y_3^\gamma. \quad (5.2)$$

where

$$t_{\alpha,\beta,\gamma} = \sum_{i=1}^r \omega_i \xi_{i,1}^\alpha \xi_{i,2}^\beta \xi_{i,3}^\gamma$$

and  $y_j$  are the components of the gradient vector for  $j = 1, \dots, 3$ .

2. Parametrize  $T$  by a sum of squares of polynomials of degree  $\frac{d}{2}$  in order to obtain

a positive symmetric tensor using the Ternary quartics theorem [GDM09]

$$T(\mathbf{x}) = \sum_{i=1}^r \omega_i \mathbf{u}(\mathbf{x}, \xi_i)^2. \quad (5.3)$$

where  $\omega_i \in \mathbb{R}^+$  are weights and  $\xi_i = (\xi_{i,1}, \xi_{i,2}, \xi_{i,3}) \in \mathbb{R}^3$  are the normalized coefficients of polynomials representing rank one tensors.

3. Substitute 5.3 in 5.1 in order to estimate the Diffusion signal and to compute each  $\xi_i$

which represents each orientation.

$$\tilde{S}(\mathbf{y}_i, \lambda_i) = \sum_{i=1}^r \omega_i \int_{S_2} \mathbf{u}(\mathbf{x}, \xi_i)^2 \Psi(\mathbf{y}_i, \lambda_i, \mathbf{x}) d\mathbf{x}. \quad (5.4)$$

- a. Minimize  $E$  to compute  $\omega_i \in \mathbb{R}^+$  using **NNLS** algorithm

$$E = \sum_{l=1}^L \left( \frac{S_l}{S_0} - \sum_{i=1}^r \omega_i \int_{S_2} \mathbf{u}(\mathbf{x}, \xi_i)^2 \Psi(\xi_l, \omega_l, \mathbf{x}) d\mathbf{x} \right)^2. \quad (5.5)$$

- b. Multiply the matrix of  $[\xi_i^{(\alpha,\beta,\gamma)}]_{|\alpha,\beta,\gamma|=d, 1 \leq i \leq r}$  by the vector  $[\omega_i]_{1 \leq i \leq r}$  to obtain the coefficients  $t_{\alpha,\beta,\gamma} = \sum_{i=1}^r \omega_i \xi_i^{(\alpha,\beta,\gamma)}$ .
-

## 5.2 Algorithm

In this section, we show how to decompose a  $d$ -order Symmetric Tensor of dimension  $n$  as a weighted sum of product of linear forms such that

$$T(\mathbf{x}) = \sum_{i=1}^r \omega_i (\mathbf{u}_i(\mathbf{x}))^d. \quad (5.6)$$

where  $\mathbf{u}_i(\mathbf{x}) = (\xi_{i,0}x_0, \xi_{i,1}x_1, \dots, \xi_{i,n}x_n)$ . We suppose  $\xi_{i,0} \neq 0$ , we scale  $\mathbf{u}_i(\mathbf{x})$  and we multiply  $\omega_i$  by the scaling factor so  $\xi_{i,0} = 1$ . We dehomogenize  $T(\mathbf{x})$  by supposing  $x_0 = 1$  then by **Newton multinomial Formula** we get

$$t_\alpha = \sum_{i=1}^r \omega_i \binom{d}{\alpha} \xi_i^\alpha = \sigma_\alpha \binom{d}{\alpha} \quad (5.7)$$

We propose an algorithm to recover the  $t_\alpha$  coefficients of a **ODF** Symmetric tensor of low rank in 3 dimensional space. We compute the coefficients  $t_\alpha$  using the moments of formal power series associated to the Tensor  $T$ . We adapt the algorithm proposed in [HMK17] which describes the decomposition of formal power series using the associated multivariate Hankel Matrices. Only one constraint appears on the rank  $r$  of the tensor  $T$ , which is resumed by  $r < 3$ . We use properties of the associated Artinian Gorenstein Algebra. The eigenvectors and the eigenvalues of the multiplication matrices are mainly

used to compute the components of directions and the ponderations.

---

**Algorithm 9 :** Crossing fibers detection via **ODF** symmetric tensor decomposition.

---

*Input :* the moments  $t_\alpha = \frac{\sigma_\alpha}{\binom{d}{\alpha}}$  of  $\sigma$  for  $|\alpha| \leq d$ .

Let  $d_1$  and  $d_2$  be positive integers such that  $d_1 + d_2 + 1 = d$ , for example  $d_1 := \lceil \frac{d-1}{2} \rceil$  and  $d_2 := \lfloor \frac{d-1}{2} \rfloor$ .

1. Compute the Hankel matrix  $H_{T^*}^{d_1, d_2} = [t_{\alpha+\beta}]_{\substack{|\alpha| \leq d_1 \\ |\beta| \leq d_2}}$  of  $T^*$  in for the monomial sets

$$A_1 = (\mathbf{x}^\alpha)_{|\alpha| \leq d_1} \text{ and } A_2 = (\mathbf{x}^\beta)_{|\beta| \leq d_2}.$$

2. Compute the singular value decomposition of  $H_{T^*}^{d_1, d_2} = USV^\top$  with singular values  $s_1 \geq s_2 \geq \dots \geq s_m \geq 0$ .

3. Determine its numerical rank, that is, the largest integer  $r$  such that  $\frac{s_r}{s_1} \geq \epsilon$ .

4. Form the matrices  $M_{x_j}^{\bar{V}_r} = S_r^{-1} U_r^H H_{x_j \star T^*}^{d_1, d_2} \bar{V}_r$ ,  $j = 1, \dots, n$ , where  $H_{x_j \star T^*}^{d_1, d_2}$  is the Hankel matrix associated to  $x_j \star T^*$ .

5. Compute the eigenvectors  $\mathbf{v}_j$  of  $\sum_j l_j M_{x_j}$  for a random choice of  $l_j$  in  $[-1, 1]$ ,  $j = 1, \dots, n$

and for each  $i = 1, \dots, r$  do the following:

a. Compute  $\xi_{i,j}$  such that  $M_j \mathbf{v}_i = \xi_{i,j} \mathbf{v}_i$  for  $j = 1, \dots, n$  and deduce the point  $\xi_i := (\xi_{i,1}, \dots, \xi_{i,n})$ .

b. Compute  $\omega_i = \frac{\langle T^* | \mathbf{v}_i(\mathbf{x}) \rangle}{\mathbf{v}_i(\xi_i)} = \frac{[1]^\top H_{T^*}^{d_1, d_2} \bar{V}_r \mathbf{v}_i}{[\xi_i^\alpha]_{\alpha \in A_2}^\top \bar{V}_r \mathbf{v}_i}$

where  $[1]$  is the coefficient vector of 1 in the basis  $A_1$ .

*Output :*  $r \in \mathbb{N}$ ,  $\omega_i \in \mathbb{C} \setminus (0)$ ,  $\xi_i \in \mathbb{C}^n$ ,  $i = 1, \dots, r$  such that

$$T^*(\mathbf{y}) = \sum_{i=1}^r \omega_i \mathbf{e}_{\xi_i}(\mathbf{y}) \text{ up to degree } d.$$


---

Notice that if  $n = 3$  and  $d = 4$  then  $d_1 = d_2 = 1$  so that  $A_1 = (1, x_2, x_3)$  and  $A_2 = (1, x_2, x_3)$ , we obtain  $\text{rank}(H_{T^*}) \leq 3$ . If  $r > 4$  we call the Numerical Completion Methods using **SDP** and **SVT** algorithms.

### 5.2.1 Numerical Results

We import 16 tensors computed in [MKBD14] using Algorithm 8, each one of them of order 4 is represented in three dimensional space by 15 coefficients with 2 or 3 directions extracted using Algorithm 8. We associate to each tensor the monomials in the following order:

$$[x_2^4, x_1x_2^3, x_1^2x_2^2, x_1^3x_2, x_1^4, x_0x_2^3, x_0x_1x_2^2, x_0x_1^2x_2, x_0x_1^3, x_0^2x_2^2, x_0^2x_1x_2, x_0^2x_1^2, x_0^3x_2, x_0^3x_1, x_0^4].$$

We apply Algorithm 9 to obtain the weights and the directions of fibers which model the tensor. We compute the angular error between the input angle between two fibers which represent directions of the tensor and the output angle between directions of output decomposed tensor. Figure 5.1 shows that the error is lower than 5 degrees for all angles greater than 30 degrees. For two orthogonal fibers of a tensor, the angular error is almost zero. In Figure 5.2, we represent the directions of 2 fibers of ODF tensor in 3-dimensional space computed using Algorithm 9 which are spaced with an angle in the range  $90^\circ : -6^\circ : 0^\circ$ . We recover the input angle for all tensors with input angle bigger than  $30^\circ$ . We notice that for each tensor, there is one preponderate direction to another unless in the case of orthogonal directions.



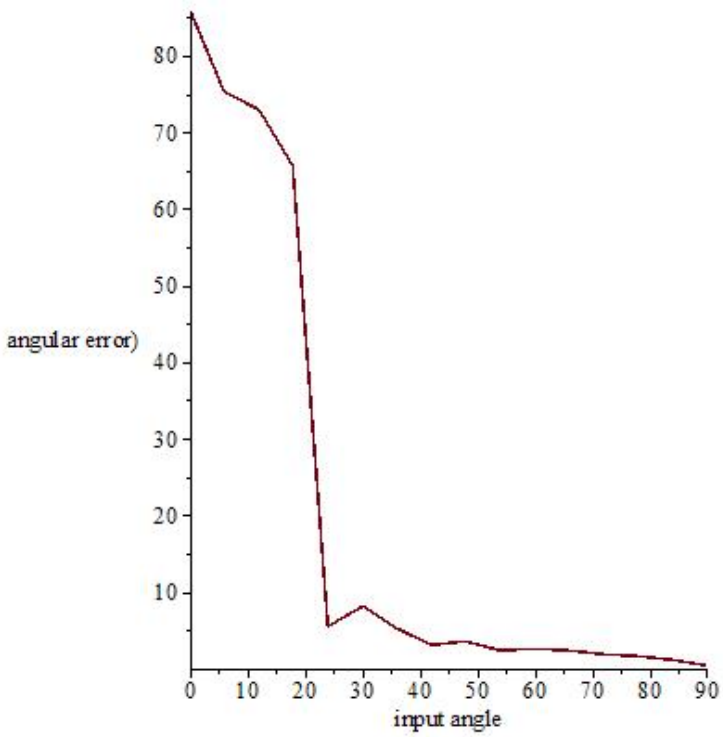


Figure 5.1: Angular error between input and output directions

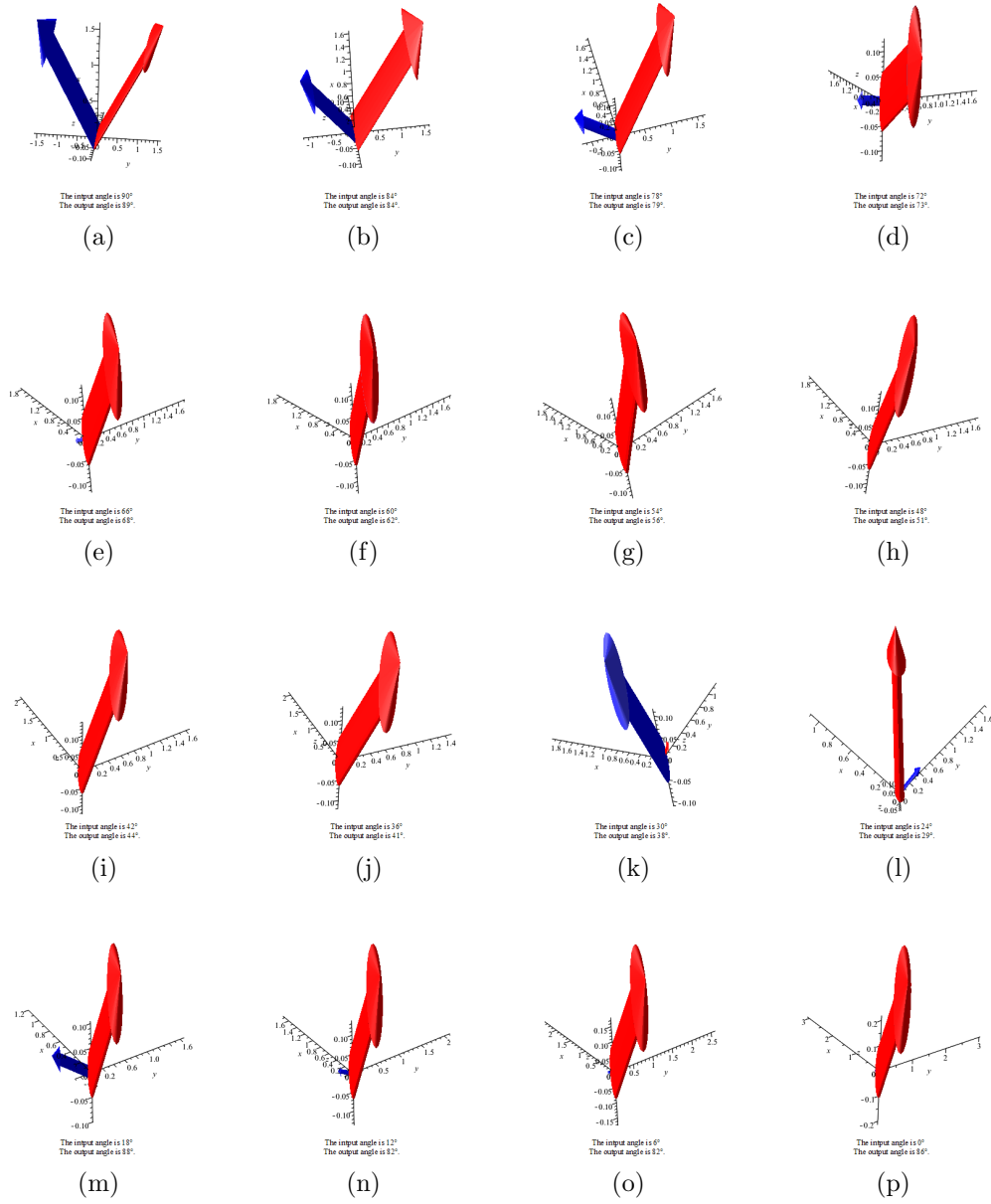


Figure 5.2: Weighted directions of 2 fibers representing 4-order ODF tensors in 3 dimensional space.

## Chapter 6

# Completion of Hankel Matrix of Low Rank

In this chapter, we study the problem of completion of Hankel matrix of low rank knowing some entries of this matrix. The problem is seen as NP-hard non-convex optimization problem as follows:

$$\begin{aligned} & \underset{X}{\text{minimize}} && \text{Rank}(X) \\ & \text{subject to} && X[i, j] = M[i, j], \quad (i, j) \in \Omega \\ & && X \text{ is Hankel.} \end{aligned} \tag{6.1}$$

where  $X \in \mathbb{R}^{m \times n}$  is the optimization variable,  $M_{i,j}$  are known given values and  $\Omega$  is a subset of  $\{1, 2, \dots, m\} \times \{1, 2, \dots, n\}$  of cardinal  $N$ . The completion of missed values of a matrix of low rank is a difficult non-convex optimization problem. In the last two decades, the matrix and formal series completion problems has been developed [CT10] [UC15]. Several questions related to semidefinite programs have been raised, solved and applied to several applications such that identification systems, graph theory, information theory, image processing and estimation problems. It also appears in the calculus of low rank decomposition of symmetric tensors. The resolution of completion problem of Hankel matrix is based on semidefinite positive programming SDP. In this chapter, we give the definition of semidefinite programming SDP in addition to Rank Minimisation Problem RMP. We present some results of RMP for a general matrix and we apply them to the completion problem of Hankel matrix of Low rank. We propose two heuristics which are based on convex optimization and resolve approximatively the RMP. The first heuris-

tic is based on the minimization of the trace where the problem (6.1) has been relaxed into an SDP problem. The second heuristic is based on the minimization of nuclear norm of the matrix. Contrarily to existing methods, these heuristics can be applied to a general matrix, they are very effective numerically and gives an a global sub born over the RMP if the feasible set is born. First, we study the trace minimization in two cases: first case where  $X$  is semi-definite positive and second case where  $X$  is symmetric non semi-definite positive. This heuristic relies on the fact that the minimization of norm  $l_1$  of a vector (here the vector of eigenvalues) gives a sparse vector. We generalize the problem to the nuclear norm minimization problem which gives an optimal solution to the problem (6.1) by minimizing the convex hull of the objective function *Rank*.

## 6.1 Semidefinite Programming

A semidefinite program SDP with variable  $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$  is the following optimization problem:

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimize}} && c^\top \mathbf{x} \\ & \text{subject to} && A_0 + A_1 x_1 + \dots + A_n x_n \geq B \end{aligned} \tag{6.2}$$

where  $A_i, B \in \mathbb{R}^{m \times m}$  are square symmetric matrices,  $c \in \mathbb{R}^n$  and  $\geq$  is a matrix inequality defined in Section 6.2.1. In other terms, a semidefinite program minimize a linear function with linear matrix inequalities.

The SDP may be solved globally using interior points methods which are effective [BV04]. If dimensions  $m$  and  $n$  are large, some special methods can be proposed to do the computation [VB96]. Some other methods have been developed recently to exploit the special structure of some problems such that the method of Gauss-Newton [KMR<sup>+</sup>01], interior points methods with double echelon [Tod97].

## 6.2 Minimization Rank Problem-RMP

The general form of low rank minimization problem RMP is

$$\begin{aligned} & \underset{X}{\text{minimize}} && \text{Rank}(X) \\ & \text{subject to} && X \in \mathcal{C} \end{aligned} \tag{6.3}$$

where  $X \in \mathbb{R}^{m \times n}$  is the optimization variable, and  $\mathcal{C}$  is the set defined by the constraints. This problem appears in several domains such that identification systems, statistics, signal processing and combinatory optimization. However, RMP is known to be a NP-hard difficult problem [CR09]. Then, the completion of Hankel matrix is a RMP problem where the matrix  $X$  is Hankel.

### 6.2.1 RMP Semidefinite Positive

**Definition 6.2.1.** *We define par  $S_+^n = \{X \in \mathbb{R}^{n \times n} \mid X \text{ is semidefinite positive}\}$ . It is a convex set named semidefinite positive cone.*

The cone  $S_+^n$  has some properties which help to find a low rank matrix. This matrix can be found on the boundary of the cone. A matrix  $X$  is called semi-definite positive and denoted by  $X \geq 0$  if it is symmetric and all eigenvalues are positive or zero. Another characterization frequently used is:  $X \in S_+^n \iff u^\top X u \geq 0, \forall u \in \mathbb{R}^n$ .

**Definition 6.2.2.** *The RMP is called semidefinite positive if the matrix  $X$  is semidefinite positive, which means that the set  $\mathcal{C}$  is included in the cone  $S_+^n$ .*

There is a lot of applications where  $X$  is not necessarily semidefinite positive or even not square. We could verify using the following lemma that every general RMP could be integrated in a bigger semidefinite positive RMP.

**Lemma 6.2.3.** *Let  $X \in \mathbb{R}^{m \times n}$  a given matrix. So that  $\text{Rank}(X) \leq r$  if and only if there exist two matrices  $Y = Y^\top \in \mathbb{C}^{m \times m}$  and  $Z = Z^\top \in \mathbb{R}^{n \times n}$  such that*

$$\text{Rank}(Y) + \text{Rank}(Z) \leq 2r$$

and

$$\begin{bmatrix} Y & X \\ X^\top & Z \end{bmatrix} \geq 0.$$

The proof is given in [Faz02] and uses the generalization of the well known condition of the complement of Schur of semidefinite positive matrix. This lemma shows that we could associate to any non-square matrix, a semidefinite positive matrix such that the rank is two times bigger than the rank of  $X$ . It means that for a non-square general matrix, the RMP (6.3) is equivalent to the rank minimization problem of the diagonal semidefinite positive matrix  $\text{diag}(Y, Z)$  [Faz02].

$$\begin{aligned} & \text{minimize} && \frac{1}{2} \mathbf{Rank} \text{diag}(Y, Z) \\ & \text{subject to} && \begin{bmatrix} Y & X \\ X^\top & Z \end{bmatrix} \succeq 0, \\ & && X \in \mathcal{C}. \end{aligned} \tag{6.4}$$

with the variables  $X, Y$  and  $Z$ . This equivalence is based on the fact that the triplet  $(X^*, Y^*, Z^*)$  is optimal for (6.4) if  $(X^*)$  is optimal for (6.3) and the objective values are the same for two problems (That is why we have the factor  $\frac{1}{2}$ ).

### 6.3 Relaxations: Trace, Nuclear Norm

In this section, we describe heuristics based on convex optimization which resolve approximatively the RMP. Contrarily to some existing methods, these heuristics may be applied to any general matrix non-positive or non-square necessarily and they are numerically effective and do not need an initial point specified by the user. We will see that RMP could be used for the completion of Hankel matrix of low rank.

#### 6.3.1 Matrix Norms

For a rectangular matrix  $X \in \mathbb{R}^{m \times n}$ ,  $\sigma_i$  is the  $i^{\text{th}}$  singular value of  $X$  and it is equal to the  $i^{\text{th}}$  eigenvalue of  $XX^\top$ . The numerical rank of  $X$  is designed by  $r$  and it is equal to the number of top non-zero singular values of  $X$  ranged in a decreasing order.

**Definition 6.3.1.** Let  $X, Y$  two matrices of the same dimension, we define the inner product in  $\mathbb{R}^{m \times n}$  by

$$\langle X | Y \rangle := \mathbf{Tr}(X^T Y) = \sum_{i=1}^m \sum_{j=1}^n X[i, j] Y[i, j]$$

**Definition 6.3.2.** The norm associated to the inner product is named Frobenius Norm, it is equal to the euclidean norm of the vector of singular values:

$$\|X\|_F := \sqrt{\mathbf{Tr}(X^T X)} = \left( \sum_{i=1}^m \sum_{j=1}^n X[i, j]^2 \right)^{\frac{1}{2}} = \left( \sum_{i=1}^{\min(m, n)} \sigma_i^2 \right)^{\frac{1}{2}}$$

**Definition 6.3.3.** The Spectral Norm is the norm  $l_\infty$  of the vector of singular values which is equal to maximum singular value and it is denoted by:

$$\|X\| := \sigma_1(X).$$

**Definition 6.3.4.** The Nuclear Norm is the norm  $l_1$  of the vector of singular values which is equal to the sum of singular values and it is denoted by:

$$\|X\|_* := \sum_{i=1}^{\min(m, n)} \sigma_i.$$

It is also called Ky-Fan or  $n$ -norm of  $X$ .

The three norms are related using the following inequalities which hold for any matrix of rank low or equal to  $r$ :

$$\|X\| \leq \|X\|_F \leq \|X\|_* \leq \sqrt{r} \|X\|_F \leq r \|X\|. \quad (6.5)$$

**Definition 6.3.5.** For every norm in the space of inner product, there is a norm which is called Dual Norm  $\|\cdot\|$  defined by:

$$\|X\|_d := \max \langle X | Y \rangle : \|Y\| \leq 1.$$

Then, the dual norm of the norm  $\|\cdot\|_d$  is the norm  $\|\cdot\|$ .

**Proposition 6.3.6.** The dual norm of the spectral norm  $\|\cdot\|$  in  $\mathbb{R}^{m \times n}$  is the nuclear norm  $\|\cdot\|_*$ .

The proof of this proposition is developed in [RFP10].

We recall the definition of Cauchy-Schwartz inequality as follows:

**Definition 6.3.7.** For every  $X, Y \in \mathbb{R}^{m \times n}$ ,  $\langle X|Y \rangle \leq \|X\|_F \|Y\|_F$ .

It is either proved in [CR09] that  $\langle X|Y \rangle \leq \|X\|_* \|Y\|$ .

**Theorem 6.3.8.** The nuclear norm  $\|\cdot\|_*$  of a matrix is convex.

For a proof, see [Hos16].

### 6.3.2 Trace Minimization

Consider the problem (6.3), we minimize the trace in the case of semidefinite positive matrix and then in the case of symmetric non semidefinite positive matrix.

#### Case 1: X semi-definite positive

We suppose that the optimization variable  $X$  is a semi-definite positive matrix,  $\lambda_i$  is the  $i^{th}$  eigenvalue of  $X$ ,  $\lambda = \{\lambda_i \mid i = 1, \dots, n\}$  and  $\|\cdot\|_1$  is the norm  $l_1$  of the vector. A good heuristic relies on replacing the objective **Rank** by the trace **Tr** and to solve the problem:

$$\begin{aligned} & \underset{X}{\text{minimize}} && \mathbf{Tr}(X) \\ & \text{subject to} && X \in \mathcal{C} \\ & && X \geq 0 \end{aligned} \tag{6.6}$$

We denote by  $\mathbf{Tr}(X) = \sum_{i=1}^n \lambda_i$ . Because of  $X$  is semidefinite positive, which means that all eigenvalues are positive or zero then  $\mathbf{Tr}(X) = \|\lambda\|_1 = \sum_{i=1}^n |\lambda_i|$ . To obtain a sparse vector, we minimize the norm  $l_1$  of the vector. Therefore, we reduce the number of non-zero eigenvalues while minimizing the norm  $l_1$  of the vector  $\lambda$  which gives a matrix of low rank. Therefore, the semidefinite positive Hankel matrix completion problem is the following:

$$\begin{aligned} & \underset{X}{\text{minimize}} && \mathbf{Tr}(X) \\ & \text{subject to} && X \geq 0 \\ & && X[i, j] = M[i, j], (i, j) \in \Omega \\ & && X \text{ is Hankel} \end{aligned} \tag{6.7}$$



The problem (6.7) is a convex relaxation SDP of problem (6.1) in the case of semidefinite positive matrix.

### 6.3.3 X is Symmetric Non Semidefinite Positive

We could extend the heuristic of minimization of the trace to the case of  $X$  is symmetric but not necessarily semidefinite positive. In this case, the sum of eigenvalues of  $X$  is not definitely equal to the norm  $l_1$  of  $\lambda$  where  $\lambda$  is the vector of eigenvalues of  $X$ . Therefore, we minimize the sum of absolute values of eigenvalues

$$\begin{aligned} & \underset{X}{\text{minimize}} && \sum_{i=1}^r |\lambda_i| \\ & \text{subject to} && X \in \mathcal{C} \end{aligned} \tag{6.8}$$

We prove then that this problem can be written as an SDP and could be easily solved.

**Proposition 6.3.9.** *The problem 6.8 is equivalent to the following SDP*

$$\begin{aligned} & \underset{X}{\text{minimize}} && \mathbf{Tr}(X_+) + \mathbf{Tr}(X_-) \\ & \text{subject to} && X = X_+ - X_- \\ & && X_+ \geq 0, X_- \geq 0 \\ & && X \in \mathcal{C} \end{aligned} \tag{6.9}$$

*Proof.* The function  $\sum_{i=1}^r |\lambda_i|$  is convex as it is a matrix norm. If we suppose that  $X = X_+ - X_-$ , then because of convexity we have

$$\begin{aligned} \sum_{i=1}^r |\lambda_i| & \leq \frac{1}{2} \left( \sum_{i=1}^r |(\lambda_i)_+| + \sum_{i=1}^r |(\lambda_i)_-| \right) \\ & = \frac{1}{2} \left( \mathbf{Tr}(X_+) + \mathbf{Tr}(X_-) \right) \end{aligned} \tag{6.10}$$

We prove that there exists two matrices  $X_+$ ,  $X_-$  such that the inequality (6.10) holds. Let  $X = Q\lambda Q^T$  is the eigenvalue decomposition of  $X$ , we collect the non-negative eigenvalues and the negative eigenvalues as the diagonal entries of  $\lambda_+$  and  $\lambda_-$  respectively. For example if  $X$  has  $k$  eigenvalues where  $d$  are

non-negative,  $\lambda_+ = \mathbf{diag}(\lambda_1, \lambda_2, \dots, \lambda_d)$  and  $\lambda_- = \mathbf{diag}(\lambda_{d+1}, \lambda_{d+2}, \dots, \lambda_k)$ . As well, we collect the corresponding eigenvalues as  $Q_+$  and  $Q_-$ . We obtain,

$$X = Q\lambda Q^\top = \begin{bmatrix} Q_+ & Q_- \end{bmatrix} \begin{bmatrix} \lambda_+ & 0 \\ 0 & \lambda_- \end{bmatrix} \begin{bmatrix} Q_+^\top \\ Q_-^\top \end{bmatrix}$$

Then, we suppose that  $X_+ = Q_+\lambda_+Q_+^\top$  and  $X_- = Q_-\lambda_-Q_-^\top$ . They are compatible with the inequality in (6.10). The two problems (6.7) and (6.8) are equivalent.  $\square$

It is the case of a general RMP (6.3), where the optimization variable  $X$  is symmetric not necessarily semidefinite positive. In particular, this propriety is valid for the problem (6.1). Furthermore, if  $m = n$  we deduce that the following problem:

$$\begin{aligned} & \underset{X}{\text{minimize}} && \mathbf{Tr}(X_+) + \mathbf{Tr}(X_-) \\ & \text{subject to} && X = X_+ - X_- \\ & && X_+ \geq 0, X_- \geq 0 \\ & && X[i, j] = M[i, j], (i, j) \in \Omega \\ & && X \text{ is Hankel} \end{aligned} \tag{6.11}$$

is a relaxation of the difficult problem 6.1. This is a convex optimization problem SDP which takes two semidefinite positive matrices  $X_+$  and  $X_-$  as optimization variables and minimize the sum of their trace under the condition that their difference  $X = X = X_+ - X_-$  is a Hankel matrix.

## 6.4 General Case: Minimization of Nuclear Norm

The minimization of trace is a heuristic which could not be applied to problems of minimization of rank RMP of a non semidefinite positive matrix. The extension to the problems where  $X$  is not semidefinite positive or more generally to non-square matrices is not evident because of the trace is not defined for non square matrices. However, there exists different important applications of RMP where the optimization variable  $X$  is not square matrix. We show in the following that this heuristic is simple and effective to be

extended to the general RMP. Using Lemma 6.2.3, we could integrate every general RMP (6.3) in a semidefinite positive problem

$$\begin{aligned} & \text{minimize} \quad \mathbf{Rank} \, \mathbf{diag}(Y, Z) \\ & \text{subject to} \quad \begin{bmatrix} Y & X \\ X^\top & Z \end{bmatrix} \geq 0 \\ & \quad \quad \quad X \in \mathcal{C} \end{aligned} \tag{6.12}$$

where  $Y \in \mathbb{R}^{m \times m}$  and  $Z \in \mathbb{R}^{n \times n}$  are additional variables. Because of the arguments of the function **Rank** in the problem (6.12) are known to be semidefinite positive, the direct application of the trace as in (6.6) gives

$$\begin{aligned} & \text{minimize} \quad \mathbf{Tr} \, \mathbf{diag}(Y, Z) \\ & \text{subject to} \quad \begin{bmatrix} Y & X \\ X^\top & Z \end{bmatrix} \geq 0 \\ & \quad \quad \quad X \in \mathcal{C} \end{aligned} \tag{6.13}$$

which is a convex optimization problem in  $X, Y$  and  $Z$  and could be solved efficiently. Therefore, we obtain an equivalent form of (6.13) which gives an intuition of this heuristic and its relationship to the general RMP. In the following, we prove that (6.13) is equivalent to the problem

$$\begin{aligned} & \underset{X}{\text{minimize}} \quad \|X\|_* \\ & \text{subject to} \quad X \in \mathcal{C} \end{aligned} \tag{6.14}$$

where  $\|X\|_* = \sum_{i=1}^{\min(p,q)} \sigma_i$  is the nuclear norm of the matrix  $X$ . We see that this norm is the dual norm of the spectral norm  $\|\cdot\|$ . The equivalence between (6.13) and (6.14) is given by the following lemma:

**Lemma 6.4.1.** *For  $X \in \mathbb{R}^{m \times n}$  and  $t \in \mathbb{R}$ , we have  $\|X\|_* \leq t$  if and only if there exists two matrices  $Y \in \mathbb{R}^{m \times m}$  and  $Z \in \mathbb{R}^{n \times n}$  such that*

$$\text{Tr}(Y) + \text{Tr}(Z) \leq 2t, \begin{bmatrix} Y & X \\ X^\top & Z \end{bmatrix} \geq 0 \tag{6.15}$$

*Proof.* ( $\Leftarrow$ ) Let  $Y$  and  $Z$  two matrices which satisfy the relations (6.15) and  $X = USV^\top$  the singular value decomposition of  $X$ . Here,  $\Sigma$  is of dimension  $r \times r$

where  $r$  is the rank of  $X$ . As the trace of product of two semidefinite positive matrices is non-negative, then  $\mathbf{Tr} \begin{bmatrix} UU^\top & -UV^\top \\ -VU^\top & VV^\top \end{bmatrix} \begin{bmatrix} Y & X \\ X^\top & Z \end{bmatrix} \geq 0$

which gives

$$\mathbf{Tr}(UU^\top Y) - \mathbf{Tr}(UV^\top X^\top) - \mathbf{Tr}(VU^\top Y) + \mathbf{Tr}(VV^\top Z) \geq 0 \quad (6.16)$$

As the columns of  $U$  are orthonormal, we could add other columns to fill it, which means that there exists  $\tilde{U}$  such that  $[U\tilde{U}][U\tilde{U}]^\top = I$ , or  $UU^\top + \tilde{U}\tilde{U}^\top = I$ . As  $Y \geq 0$ ,  $\mathbf{Tr}(\tilde{U}\tilde{U}^\top Y) \geq 0$  and we have

$$\mathbf{Tr}(\tilde{U}\tilde{U}^\top Y) \leq \mathbf{Tr}(UU^\top + \tilde{U}\tilde{U}^\top Y) = \mathbf{Tr}(Y).$$

As well, for  $V$  we have  $\mathbf{Tr}(\tilde{V}\tilde{V}^\top Z) \leq \mathbf{Tr}(Z)$ . Furthermore, we have  $\mathbf{Tr}(V\tilde{U}^\top X) = \mathbf{Tr}(V\Sigma\tilde{V}^\top) = \mathbf{Tr}(\Sigma)$  and  $\mathbf{Tr}(U\tilde{V}^\top X^\top) = \mathbf{Tr}(U\Sigma\tilde{U}^\top) = \mathbf{Tr}(\Sigma)$ . We replace these equalities in (6.16) and use the two previous inequalities to obtain,

$$\mathbf{Tr}(Y) + \mathbf{Tr}(Z) - 2\mathbf{Tr}(\Sigma) \geq 0,$$

$$\frac{1}{2}(\mathbf{Tr}(Y) + \mathbf{Tr}(Z)) \geq \mathbf{Tr}(\Sigma) \geq 0,$$

$$\mathbf{Tr}(\Sigma) = \|X\|_* \leq t.$$

( $\Rightarrow$ ) We suppose that  $\|X\|_* \leq t$ . The two matrices  $Y$  and  $Z$  which we are looking for may be chosen to satisfy the relations (6.15): If  $Y = U\Sigma U^\top + \gamma I$  and  $Z = V\Sigma V^\top + \gamma I$  so that we have

$$\mathbf{Tr}(Y) + \mathbf{Tr}(Z) = 2\mathbf{Tr}(\Xi) + \gamma(p+q) = 2\|Tr\|_*(\Xi) + \gamma(p+q).$$

We choose  $\gamma = \frac{2(t-\|X\|_*)}{p+q}$ , we obtain  $\mathbf{Tr}(Y) + \mathbf{Tr}(Z) = 2t$ . We notice that,

$$\begin{bmatrix} Y & X \\ X^\top & Z \end{bmatrix} = \begin{bmatrix} U\Sigma U^\top & U\Sigma V^\top \\ V\Sigma U^\top & V\Sigma V^\top \end{bmatrix} + \gamma \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} = \begin{bmatrix} U \\ V \end{bmatrix} \Sigma \begin{bmatrix} U^\top & V^\top \end{bmatrix} + \gamma I$$

is a semidefinite positive matrix. In other terms, the condition  $\|X\|_* \leq t$  could be seen as a linear equality matrix. Now, it is simple to demonstrate that the generalized heuristic of minimization of trace (6.13) and the minimization of

nuclear norm (6.14) are equivalent. Simply, we could write the problem (6.13) as follows:

$$\begin{aligned}
& \text{minimize} && t \\
& \text{subject to} && \mathbf{Tr}(Y) + \mathbf{Tr}(Z) \leq 2t \\
& && \begin{bmatrix} Y & X \\ X^\top & Z \end{bmatrix} \succeq 0 \\
& && X \in \mathcal{C}
\end{aligned} \tag{6.17}$$

with the variables  $X, Y, Z$  and  $t$ . Then, if we apply lemma (6.4.1), we obtain

$$\begin{aligned}
& \text{minimize} && t \\
& \text{subject to} && \|X\|_* \leq t \\
& && X \in \mathcal{C}
\end{aligned} \tag{6.18}$$

with the variables  $X$  and  $t$ , which is equivalent to the problem (6.14)  $\square$

We now recall the relationship between this problem and the original RMP which has been developed in (6.3).

## 6.5 Convex Hull of the Rank function

We explain in more details the relationship between the RMP (6.3) and the heuristic of nuclear norm (or generalized trace)(6.14).

**Definition 6.5.1.** *Let  $\mathcal{C}$  be a convex set and  $f : \mathcal{C} \rightarrow \mathbb{R}$  not necessarily convex. The convex hull is the largest convex function  $g$  such that  $g(x) \leq f(x)$  for  $x \in \mathcal{C}$ . This means that among all convex functions,  $g$  is the best punctual approximation of  $f$ .*

In particular, if the optimal  $g$  can be described than  $f$  could be minimized efficiently. In the problem (6.3) where the objective function is not convex, its convex hull could be used as an approximation which could minimize the rank efficiently.

**Theorem 6.5.2.** *The convex hull of the function  $\phi(X) = \mathbf{Rank}(X)$  over the set  $\mathcal{C} = \{X \in \mathbb{R}^{m \times n}; \|X\| \leq 1\}$  is  $\phi_{env}(X) = \|X\|_* = \sum_{i=1}^{\min(m,n)} \sigma_i$ .*

*Proof.* Because of (6.5), we have  $\|X\|_* \leq r\|X\|$ . This gives that  $\mathbf{Rank}(X) \geq \frac{\|X\|_*}{\|X\|}$  for every  $X$ . Then, for every matrix  $X$  where  $\|X\| \leq 1$ , we have  $\mathbf{Rank}(X) \geq \|X\|_*$ . Then, the nuclear norm is the convex inferior limit of the function  $\mathbf{Rank}$  over the set  $\mathcal{C}$ . In fact, this is the closer convex inferior limit which is more developed in [Faz02].  $\square$

This theorem gives an important interpretation of the heuristic of the nuclear norm: in fact, this heuristic minimize the convex hull of the  $\mathbf{Rank}$  function on a bounded set. In addition, we have the following implications for the general RMP (6.3) and the following heuristic: Suppose that the set  $\mathcal{C}$  is bounded by  $M$ , which means that for every  $X \in \mathcal{C}$ ,  $\|X\| \leq M$ . The convex hull of  $\mathbf{Rank}(X)$  over the set  $\{X \in \mathbb{R}^{m \times n}; \|X\| \leq M\}$  is given by  $\frac{1}{M}\|X\|_*$  which means that  $\mathbf{Rank}(X) \geq \frac{1}{M}\|X\|_*$  for every  $X \in \mathcal{C}$ . We denote by  $p_{rmp}$  the optimal value of the rank minimization problem (6.3) and  $p_{tr}$  the optimal value of nuclear norm minimization problem, then

$$p_{rmp} \geq p_{tr}.$$

In other words, solving (6.14), we could obtain a lower bound of the optimal value of Problem (6.3). We notice that the nuclear norm is the closest convex approximation which gives the inferior limit among all the convex approximations of the function  $\mathbf{Rank}$  over the set  $\mathcal{C}$ . The property "Convex Hull" of the nuclear norm given by the previous theorem plays an important role in the efficiency of the heuristic 6.14: the simplest way to connect the rank of the non-square general matrix to the semidefinite positive matrix is to see that  $\mathbf{Rank}(X) = \mathbf{Rank}(XX^\top)$ . As  $\mathbf{Rank}(XX^\top)$  is a semidefinite positive matrix, then we could apply directly the heuristic trace and obtain,

$$\begin{aligned} & \underset{X}{\text{minimize}} && \mathbf{Tr}(XX^\top) \\ & \text{subject to} && X \in \mathcal{C} \end{aligned} \tag{6.19}$$

We have  $\mathbf{Tr}(XX^\top) = \sum_{i=1}^r \sigma_i^2$  the Frobenius norm of  $X$  is a convex optimization of the  $\mathbf{Rank}$ . Notice that this is the norm  $l_2$  of the vector of singular values. However, we know that the minimization of the norm  $l_2$  of a vector contrarily to the norm  $l_1$  does not always give a sparse vector so that we do not expect to obtain a sparse set of singular values or a low rank matrix using the norm

$l_2$ . The minimization problem 6.19 could be written as the following SDP:

$$\begin{aligned} & \text{minimize} && \text{Tr}(Y) \\ & \text{subject to} && \begin{bmatrix} Y & X \\ X^\top & I \end{bmatrix} \geq 0 \end{aligned} \tag{6.20}$$

We notice that this problem and Problem (6.13) are the same, unless that the variable  $Z$  is replaced by the identity which reduces the number of degrees of freedom in the optimization. The completion problem of a Hankel matrix (6.1) is a rank minimization problem RMP of a Hankel matrix. In this problem  $\mathcal{C} = \{X \in \mathbb{R}^{m \times n}, X[i, j] = M[i, j], (i, j) \in \Omega, X \text{ is Hankel.}\}$ . We suppose that  $X$  is not semidefinite positive or non-square matrix so that because of the previous heuristic which represents the nearest convex approximation which gives the closest limit inferior among all the convex approximations of the **Rank** over a bounded set is given by the minimization of the nuclear norm of the matrix  $X$  as follows:

$$\begin{aligned} & \underset{X}{\text{minimize}} && \|X\|_* \\ & \text{subject to} && X[i, j] = M[i, j], (i, j) \in \Omega \\ & && X \text{ is Hankel.} \end{aligned} \tag{6.21}$$

Therefore, this problem is in relationship with Problem (6.1): solving (6.21) we obtain an inferior limit of the optimal value of (6.1). In other words, the resolution of the problem allows us to obtain a Hankel matrix of the lowest possible nuclear norm. In addition, it is the first step towards the completion of Hankel matrix  $X$  which consists of minimizing the nuclear norm which refers to solve (6.21).

## 6.6 Numerical Example SDP

In this section, we present a direct application of the completion of Hankel matrix of low rank. Given the first moments  $\sigma_\alpha, |\alpha| \leq d$  of the series  $\sigma(\mathbf{y}) = \sum_{i=1}^r \omega_i e_i(\mathbf{y})$  with  $\xi_i \in \mathbb{R}^n$  and  $\omega_i \neq 0$ . The goal is to complete the Hankel associated matrix to the series in addition to its moments  $\sigma_\alpha$ . Notice that the nuclear norm is the closest convex approximation which gives the tightest inferior limit among all convex approximations of **Rank** over the set

$\mathcal{C}$ . We solve this problem in an efficient way, we use the relaxation of the minimization of nuclear norm of  $X$ . We use the packages JuMP, MultivariatePolynomials, CSDP, DataStructures, TensorDec of the software Julia.

- JuMP makes it easy to solve optimization problems.
- MultivariatePolynomials provides an interface for manipulating multivariate polynomials such as accessing the coefficients, the monomials and the terms of them.
- CSDP is an interface to some Semi Definite Programming Solvers.
- DataStructures implements a variety of Data structures such as Dictionaries and Linked Lists.
- TensorDec manipulates the decomposition of series of moments and tensors.

**Example 6.6.1.** *Given the weights and the frequencies as follows:*

$$\omega_0 = [2 \ 3] \text{ and } \xi_0 = \begin{bmatrix} -1 & 1 \\ 0 & 4 \end{bmatrix}$$

*We have the series of moments truncated in degree  $d = 3$ .*

*The polynomial of degree 3 computed with 2 frequencies  $\xi_0$  and 2 weights  $\omega_0$  is  $t_0 = -0.001x_1^3 + 0.0402x_1^2x_2 + 0.138x_1x_2^2 + 0.194x_2^3 + 0.15x_1^2 - 0.6x_1x_2 + 1.5x_2^2 - 0.3x_1 + 4.2x_2 + 5$ .*

*We use the set of monomials of degree lower than or equal to  $d_1 = 3$  associated to the variables  $x_1$  and  $x_2$ , the Hankel matrix  $H_0$  associated to  $t_0$  before the minimization of the nuclear norm is:*

$$H_0 = \begin{bmatrix} 5 & -0.1 & 1.4 & 0.05 & -0.1 & 0.5 \\ 1.4 & -0.1 & 0.5 & 0.014 & -0.046 & 0.194 \\ 0.05 & -0.01 & 0.014 & 0 & 0 & 0 \\ -0.1 & 0.014 & -0.046 & 0 & 0 & 0 \\ 0.5 & -0.046 & 0.194 & 0 & 0 & 0 \end{bmatrix}$$

*After the minimization of the nuclear norm, the complete Hankel matrix is the following:*



$$H = \begin{bmatrix} 5 & -0.1 & 1.4 & 0.05 & -0.1 & 0.5 \\ 1.4 & -0.1 & 0.5 & 0.014 & -0.046 & 0.194 \\ 0.05 & -0.01 & 0.014 & 0.0005 & -0.001 & 0.0005 \\ -0.1 & 0.014 & -0.046 & -0.001 & 0.0005 & -0.019 \\ 0.5 & -0.046 & 0.194 & 0.0005 & -0.019 & 0.077 \end{bmatrix}$$

The rank of  $H_0$  is 5 but the numerical rank of  $H$  is 2 for  $\epsilon = 0.01$ .

We apply the low decomposition algorithm on the series associated to  $H$  and we obtain the following weights and frequencies which are near of the initial weights and frequencies:

$$\omega = [2 \ 3] \text{ and } \xi = \begin{bmatrix} -1 & 1 \\ 0 & 4 \end{bmatrix}$$

Furthermore, the norm  $l_2$  of the difference between the initial polynomial and the complete polynomial associated to  $H$  is  $10^{-15}$ .

## 6.7 Singular Value Thresholding Algorithm

In this section, we propose another algorithm which gives an optimal solution to the nuclear norm minimization problem with linear and convex constraints. This algorithm is iterative and applies a singular value thresholding step to a sparse matrix which requires a small storage space and a low computational cost at each step. The choice of enough big threshold at each step allows us to expect a low rank completed matrix. We prove from a theoretical of view point that the sequence of iterates converges.

Candés and Recht proved in [CR09] that the matrix completion problem is not as ill-posed as people think. They showed that the matrix of low rank can be reconstructed from a small set of sampled entries by solving a simple convex optimization problem.

**Theorem 6.7.1.** *A matrix  $M \in \mathbb{R}^{n \times m}$  of rank  $r$  can be covered by solving the optimization problem:*

$$\begin{aligned} & \underset{X}{\text{minimize}} \quad \|X\|_* \\ & \text{subject to} \quad X[i, j] = M[i, j], (i, j) \in \Omega \end{aligned} \tag{6.22}$$

where the number of sampled entries obeys  $p \geq Cn^{\frac{6}{5}}r \log n$  for some positive constant  $C$ .

For the proof, see [CR09]. Candés and Recht also proved in [CR09] that under some conditions, the rank minimization problem and the nuclear norm minimization problem have the same solution.

### 6.7.1 Algorithm Outline

Because it has been proved in [CR09] that there exists a solution for nuclear norm minimization problem under some conditions, it is important to develop an algorithm for solving it. The semidefinite programming solvers proposed in the previous section are problematic when the size of the matrix is big because we solve big linear equation systems and don't guarantee the matrix to be of low rank. We propose a singular value thresholding algorithm which fills a matrix such that it requires the condition of lowest possible nuclear norm. We set the nuclear norm minimization problem as:

$$\begin{aligned} & \underset{X}{\text{minimize}} && \|X\|_* \\ & \text{subject to} && \mathcal{A}(X) = b \end{aligned} \tag{6.23}$$

where  $\mathcal{A}$  is a linear operator on the space  $\mathbb{R}^{n \times m}$  into  $\mathbb{R}^l$ , and  $b \in \mathbb{R}^l$ .

The algorithm works well when the desired matrix is of big sizes and of low rank.

**Definition 6.7.2.** *Let  $\Omega$  to be a set of indices, the orthogonal projector  $\mathcal{P}_\Omega$  associated to  $\Omega$  is defined as follows:*

$$\begin{aligned} \mathcal{P}_\Omega : \mathbb{R}^{n \times m} &\longrightarrow \mathbb{R} \\ X &\longmapsto \mathcal{P}_\Omega(X) = \begin{cases} X[i, j], & \text{if } (i, j) \in \Omega \\ 0 & \text{Otherwise} \end{cases} \end{aligned} \tag{6.24}$$

We sketch Problem (6.23) in the case of completion of low nuclear norm matrix, so that the associated optimization problem is:

$$\begin{aligned} & \underset{X}{\text{minimize}} && \|X\|_* \\ & \text{subject to} && \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M) \end{aligned} \tag{6.25}$$

We propose an algorithm which is based on the computation of the top singular values of the input matrix at each step. If only a few number of the

singular values is chosen at each step then the sequence of matrices converges rapidly to a matrix of lowest possible rank. Another expectation of choosing a big threshold is to make the computation of each matrix easy and to reduce the storage space in memory.

It is convenient to generalize the constraints of problem 6.23 to be convex, as follows:

$$\begin{aligned} & \underset{X}{\text{minimize}} && \|X\|_* \\ & \text{subject to} && f_i(X) \leq 0, 1 \leq i \leq l. \end{aligned} \tag{6.26}$$

where  $f_i$  are convex and Lipschitz. The benefit of this generalization is to make possible to complete a low nuclear norm matrix with noisy sampled entries by redefining the constraints of the completion problem through the conditions of generalized problem as detailed later in 6.8.

We recall the singular value thresholding algorithm SVT for the matrix completion problem in terms of a well-known Lagrange multiplier algorithm and we extend SVT to be able to find a numerical solution to the general problem 6.23. We then adapt it to the completion problem of low rank Hankel matrix.

We recall the definition of a singular value thresholding operator as follows:

**Definition 6.7.3.** *Let  $X \in \mathbb{R}^{n \times m}$  of rank  $r$  with the truncated singular value decomposition  $X = U_r S_r V_r^\top$ . For  $\tau \geq 0$ , we introduce the singular value thresholding operator*

$$\mathcal{D}_\tau(X) = U_r \mathcal{D}_\tau(S_r) V_r^\top$$

where  $\mathcal{D}_\tau(S_r) = \text{diag}((s - \tau)_+)$ .

In other words, this operator moves the singular values which are bigger than  $\tau$  towards zero. Even though the singular value decomposition is not unique, it is easy to prove that the thresholding operator is well defined.

We recall the following theorem which is used to recast the SVT algorithm as a Lagrange multiplier algorithm:

**Theorem 6.7.4.** *Let  $\tau \geq 0$  and  $Y \in \mathbb{R}^{n \times m}$ , the singular value shrinkage operator defined in 6.7.3 obeys:*

$$\mathcal{D}_\tau(Y) = \arg \min_X \left( \frac{1}{2} \|X - Y\|_F^2 + \tau \|X\|_* \right) \tag{6.27}$$

For the proof, see [CCS10].

## Interpretation as Lagrange Multiplier Method

In this section, we recast the SVT algorithm as a Lagrange multiplier algorithm. This allows us to extend the SVT algorithm known as Uzawa's algorithm to nuclear norm optimization problem with linear or convex constraints. For some fixed  $\tau \geq 0$ , we set  $f_\tau(X) = \frac{1}{2}\|X\|_F^2 + \tau\|X\|_*$  and we define the optimization problem as follows:

$$\begin{aligned} & \underset{X}{\text{minimize}} && f_\tau(X) \\ & \text{subject to} && \mathcal{P}_\Omega(X) = \mathcal{P}_\Omega(M) \end{aligned} \tag{6.28}$$

The Lagrangian for this problem is given by:

$$\mathcal{L}(X, Y) = f_\tau(X) + \langle Y | \mathcal{P}_\Omega(M - X) \rangle$$

where  $Y \in \mathbb{R}^{n \times m}$ . Strong duality holds, and  $X^*$  is primal-optimal and  $Y^*$  is dual-optimal if we have:

$$\sup_Y \inf_X \mathcal{L}(X, Y) = \mathcal{L}(X^*, Y^*) = \inf_X \sup_Y \mathcal{L}(X, Y)$$

The function  $g_0(Y) = \inf_X \mathcal{L}(X, Y)$  is called the dual function.  $g_0$  is continuously differentiable and Lipschitz with constant Lipschitz at most one, as this is a consequence of well-known results concerning conjugate functions. Uzawa's algorithm approaches the problem of finding the point  $(X^*, Y^*)$  by solving the iterative procedure starting from  $Y^0 = 0$  as follows:

$$\begin{cases} \mathcal{L}(X^k, Y^{k-1}) = \min_X \mathcal{L}(X, Y^{k-1}) \\ Y^k = Y^{k-1} + \delta_k \mathcal{P}_\Omega(M - X^k) \end{cases} \tag{6.29}$$

where  $\delta_k$  is a sequence of positive step sizes.

We observe that the subgradient of  $g_0(Y)$  is given by [CCS10] as follows:

$$\nabla_Y g_0(Y) = \nabla_Y \mathcal{L}(\tilde{X}, Y) = \mathcal{P}_\Omega(M - \tilde{X})$$

where  $\tilde{X}$  is the minimizer of the Lagrangian for that value of  $Y$  so that the gradient descent update is of the form:

$$Y^k = Y^{k-1} + \delta_k \nabla_Y g_0(Y^{k-1}) = Y^{k-1} + \delta_k \mathcal{P}_\Omega(M - X^k).$$

It remains to compute the minimizer of the Lagrangian (6.29).

Because of (6.27) we have:

$$\arg \min_X f_\tau(X) + \langle Y | \mathcal{P}_\Omega(M - X) \rangle = \arg \min_X \left( \frac{1}{2} \|X - \mathcal{P}_\Omega(Y)\|_F^2 + \tau \|X\|_* \right) = \mathcal{D}_\tau(\mathcal{P}_\Omega(Y)).$$

So that, and because of  $Y^k = (\mathcal{P}_\Omega(Y^k))$  for  $k \geq 0$  Uzawa's algorithm takes the form:

$$\begin{cases} X^k = \mathcal{D}_\tau(Y^{k-1}) \\ Y^k = Y^{k-1} + \delta_k \mathcal{P}_\Omega(M - X^k) \end{cases} \quad (6.30)$$

### Convex Inequality Constraints

In this section, we present a general formulation of SVT algorithm for minimizing the nuclear norm under linear equality constraints.

For some fixed  $\tau \geq 0$ , we set  $f_\tau(X) = \frac{1}{2} \|X\|_F^2 + \tau \|X\|_*$  and we define the optimization problem as follows:

$$\begin{aligned} & \underset{X}{\text{minimize}} && f_\tau(X) \\ & \text{subject to} && \mathcal{A}(X) = b \end{aligned} \quad (6.31)$$

where  $\mathcal{A}$  is a linear operator on the space  $\mathbb{R}^{n \times m}$  into  $\mathbb{R}^l$ , and  $b \in \mathbb{R}^l$ .

The Lagrangian for this problem is given by:

$$\mathcal{L}(X, y) = f_\tau(X) + \langle y | b - \mathcal{A}(X) \rangle$$

where  $y \in \mathbb{R}^l$ .

The iterative procedure of Uzawa's starting from  $y^0 = 0$  is defined as follows:

$$\begin{cases} X^k = \mathcal{D}_\tau(\mathcal{A}^\top(y^{k-1})) \\ y^k = y^{k-1} + \delta_k (b - \mathcal{A}(X^k)) \end{cases} \quad (6.32)$$

For more details, see [CCS10].

### Convex Inequality Constraints

In this paragraph, we present a general formulation of SVT algorithm for minimizing the nuclear norm under affine constraints.

We set  $\mathcal{F}(X) = (f_1(X), f_2(X), \dots, f_l(X))$  where  $f_i$  are convex functionals for  $i = 1, \dots, l$  and we define the optimization problem as follows:

$$\begin{aligned} & \underset{X}{\text{minimize}} && f_\tau(X) \\ & \text{subject to} && f_i(X) \leq 0, \text{ for } i = 1, \dots, l \end{aligned} \tag{6.33}$$

The Lagrangian for this problem is given by:

$$\mathcal{L}(X, y) = f_\tau(X) + \langle y | \mathcal{F}(X) \rangle$$

where  $X \in \mathbb{R}^{n \times m}, y \geq 0 \in \mathbb{R}^l$ .

We apply a subgradient method with projection to maximize the dual function and we obtain the following Uzawa's iteration starting from  $y^0 = 0$ :

$$\begin{cases} X^k = \arg \min_X \{f_\tau(X) + \langle y^{k-1} | \mathcal{F}(X) \rangle\} \\ y^k = [y^{k-1} + \delta_k \mathcal{F}(X^k)]_+ \end{cases} \tag{6.34}$$

For more details, see [CCS10].

**When the proximal problem become close**

In the following theorem we show that minimizing the objective function  $f_\tau(X)$  is the same as minimizing the nuclear norm when the threshold  $\tau$  is big.

**Theorem 6.7.5.** *Let  $X_\tau^*$  be the solution of (6.33) and  $X_\infty$  be the minimum Frobenius norm solution of (6.26) defined as:*

$$X_\infty := \arg \min_X \{\|X\|_F^2, X \text{ solution of (6.26)}\}$$

*Assume that  $f_i(X)$  for  $i = 1, \dots, l$  are convex and lower continuous. Then*

$$\lim_{\tau \rightarrow \infty} \|X_\tau^* - X_\infty\|_F = 0.$$

For the proof, see [CCS10].

## Convergence Analysis

In this section, we recall two theorems which establish the convergence of SVT iterations associated to problems (6.28), (6.31) and (6.33).

**Theorem 6.7.6.** *Suppose that the step sizes obey  $0 < \inf \delta_k \leq \sup \delta_k < \frac{2}{\|\mathcal{A}\|^2}$ . Then the sequence  $\{X^k\}$  obtained via (6.32) converges to the unique solution of (6.31). In particular, the sequence  $\{X^k\}$  obtained via (6.30) converges to the unique solution of (6.28) provided that  $0 < \inf \delta_k \leq \sup \delta_k < 2$ .*

For the proof, see [CCS10].

**Theorem 6.7.7.** *Suppose that the step sizes obey  $0 < \inf \delta_k \leq \sup \delta_k < \frac{2}{\|L(\mathcal{F})\|^2}$  where  $L(\mathcal{F})$  is the Lipschitz constant for the the function  $(\mathcal{F}(X))$ . Then assuming the strong duality, the sequence  $\{X^k\}$  obtained via (6.34) converges to the unique solution of (6.33).*

For the proof, see [CCS10].

## 6.8 Completion of Hankel matrix using SVT algorithm

In this section, we fill a Hankel matrix with some first predefined elements such that its associated nuclear norm is minimum as much as possible. For some fixed  $\tau \geq 0$ , we minimize the function  $f_\tau(X) = \frac{1}{2}\|X\|_F^2 + \tau\|X\|_*$  with the linear constraints  $\mathcal{A}(X) = b$  of Problem (6.31). We define the linear operator  $\mathcal{A}$  in the case of  $X$  is Hankel and some elements  $X[i, j] = M[i, j]$ ,  $(i, j) \in \Omega$  are already predefined. We then apply the Uzawa's algorithm iteration (6.32) on the linear operator defined in Hankel case to obtain the Hankel matrix.

We set the predefined elements of the Hankel matrix as the first elements of a sparse vector  $b$ . We define the integer  $k_0$  which obeys:

$$\frac{\tau}{\delta\|\mathcal{A}^\top(b)\|}(k_0 - 1, k_0]$$

where  $\delta$  is a fixed positive moving step and  $\tau$  is chosen to be as large as possible to guarantee the convergence of  $f_\tau(X)$  to the nuclear norm  $\|X\|_*$ . We start the algorithm with the initial vector  $y^0 = b \star k_0 \star \delta$ . The operator  $\mathcal{A}$  and its transpose  $\mathcal{A}^\top$  depend on list of matrices  $L$  describing the constraints

of Problem (6.21). We define  $N$  as the number of matrices of  $L$ . They are defined as follows:

$$\mathcal{A}(X, L) = [\langle L[i]|X \rangle]_{i=1}^N \quad (6.35)$$

$$\mathcal{A}^\top(y, L) = \sum_{i=1}^N y[i]L[i] \quad (6.36)$$

These operators satisfy the relations

$$\mathcal{A}^\top \mathcal{A} = \mathcal{P}_\Omega$$

and

$$\mathcal{A}(M) = b$$

where  $M$  is the matrix from which the first elements of the Hankel matrix have been selected.

Given the Hankel matrix  $H_0$  in which only three first real moments are extracted from the matrix  $M$ :

$$H_0 = \begin{bmatrix} 0.1541 & 0.0268 & 0.7740 & 0.0 & 0.0 & 0.0 \\ 0.0268 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.7740 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

We complete the Hankel matrix  $H_0$  with SVT algorithm and we associate to the output Hankel matrix a series  $\sigma_0(x_1, x_2)$  of degree  $d = 4$  in two variables  $x_1, x_2$  defined as follows:

$\sigma_{initial}(x_1, x_2) = 6.0805e^{-6}x_1^2x_2 + 1.0532e^{-5}x_1x_2^3 + 0.0015x_1x_2 + 0.0002x_2^4 + 2.6041e^{-7}x_1^3 + 8.7406e^{-10}x_1^4 + 0.0035x_2^3 + 0.7740x_2 + 0.0307x_2^2 + 4.6233e^{-7}x_1^2x_2^2 + 0.0001x_1 * x_2^2 + 0.0268x_1 + 7.2915e^{-5}x_1^2 + 2.0137e^{-8}x_1^3x_2 + 0.1541$  we decompose this series into weighted sum of 2 frequencies using Algorithm 1 and we obtain the following points and weights:

$$\xi = \begin{bmatrix} 0.0033 & 0.0783 \\ -0.0006 & -0.0392 \end{bmatrix},$$

$$\omega = \begin{bmatrix} 6.63673 \\ -6.48259 \end{bmatrix}$$



Each column of  $\xi$  corresponds to a frequency and each element of  $\omega$  corresponds to a weight. We reconstruct the series associated to these points and weights and we obtain the following series

$$\begin{aligned} \sigma_{final}(x_1, x_2) = & 6.0805e^{-6}x_1^2x_2 + 1.0532e^{-5}x_1x_2^3 + 0.0015x_1x_2 + 0.0002x_2^4 + 2.5997e^{-7}x_1^3 + \\ & 8.7238e^{-10}x_1^4 + 0.0035x_2^3 + 0.7740x_2 + 0.0307x_2^2 + 4.6244e^{-7}x_1^2x_2^2 + 0.0001x_1 * x_2^2 + \\ & 0.0268x_1 + 7.3168e^{-5}x_1^2 + 2.0121e^{-8}x_1^3x_2 + 0.1541 \end{aligned}$$

We compute the norm of the difference between  $\sigma_{initial}(x_1, x_2)$  and  $\sigma_{final}(x_1, x_2)$  and we obtain:

$$\|\sigma_{initial}(x_1, x_2) - \sigma_{final}(x_1, x_2)\|_2 = 2.5245e^{-7}$$

The SVT converges to a Hankel matrix which respects the constraints of Problem (6.21) for  $\delta = 0.5$  in 3398 iterations and the convergence error:

$$\frac{\|\delta(b - \mathcal{A}(H_{initial}, L))\|_F}{\|\mathcal{A}(H_{initial}, L)\|_F} = 9.9929e^{-8}$$

where  $H_{initial}$  is the Hankel operator obtained at the convergence step of SVT algorithm and corresponding to  $\sigma_{initial}$ .

# Chapter 7

## Newton Iteration

In this section we first give an intuition of Newton method, we review the Newton-Raphson iteration to solve univariate and multivariate system. We then present a new Newton iteration, which converges locally to the multivariate Hankel matrix of a given rank which is the closest to a given input Hankel matrix. Numerical experimentations show that the Newton iteration combined the decomposition method allows to compute accurately and efficiently the polynomial-exponential decomposition of the symbol, even for noisy input moments. We finally give a numerical example which shows the efficiency of this new iteration to recover the weights and points of a series after completion step with uzawa's algorithm.

### Intuition for Newton method

The problem is to find a close approximation to the value of a function  $f$  at a point  $x \in \mathbb{R}$ ,  $f(x) \in \mathbb{R}$ . An opposite problem is to solve an equation

$$F(x) = 0 \tag{7.1}$$

where  $x \in \mathbb{R}$ . Both of problems is based on the same idea.

Suppose at a point  $x = a$  near to the solution, we know the derivate at the point defined as

$$\frac{df}{dx}(a) = \lim_{x \rightarrow a} \frac{f(x) - f(a)}{x - a}.$$

An approximation of the derivate  $\frac{df}{dx}(a)$  at the point  $a$  is given by

$$\frac{df}{dx}(a) \approx \frac{f(x) - f(a)}{x - a}.$$

So that the approximation to  $f(x)$  is given by the formula

$$f(x) \approx f(a) + (x - a) \frac{df}{dx}(a). \quad (7.2)$$

This formula tells us that if we want to find a value of  $f$  at a nearby point, we can use a linear approximation because the graph of this formula is close to a linear line. So we are following a straight line instead of following a curved graph which starts out at the correct point  $x = a$  which has a correct slope  $\frac{df}{dx}(a)$  and is close to the curve as much as possible. In the opposite way, let's create the formula which approximately solve (7.1). The difference is that  $F(x)$  is what I know to be 0 and  $x$  is what I'm looking for. So the so called Raphson equation is defined as

$$x - a \approx \frac{F(a)}{\frac{dF}{dx}(a)} \quad (7.3)$$

### Newton Iteration to solve univariate system

The Taylor series of  $F(x)$  at a point  $x = x_0 + \Delta x$  which is close to an initial guess  $x_0$  is given by

$$F(x_0 + \Delta x) = F(x_0) + \frac{dF}{dx}(x_0)\Delta x + \frac{1}{2} \frac{d^2F}{dx^2}(x_0)(\Delta x)^2 + \dots \quad (7.4)$$

The derivative corresponds to keeping only the first order terms of Taylor series as follows

$$F(x_0 + \Delta x) \approx F(x_0) + \frac{dF}{dx}(x_0)\Delta x. \quad (7.5)$$

Setting  $F(x_0 + \Delta x) = 0$  and solving (7.5) for  $\Delta x = \Delta x_0$  gives us

$$\Delta x_0 \approx \frac{F(x_0)}{\frac{dF}{dx}(x_0)}. \quad (7.6)$$

which is the first-order adjustment to the root position. By letting  $x_1 = x_0 + \Delta x_0$ , computing the new  $\Delta x_1$  and so on the process can be repeated and converges to a fixed point which is the root. The moving step at  $n$  iteration is given by

$$\Delta x_n \approx \frac{F(x_n)}{\frac{dF}{dx}(x_n)}. \quad (7.7)$$

Then, with good initial choice of the root position the algorithm can be applied iteratively to obtain the so called Newton iteration

$$x_{n+1} = x_n - \frac{F(x_n)}{\frac{dF}{dx}(x_n)} \quad (7.8)$$

This algorithm may be unstable near an horizontal asymptote or local extrema.

### Newton Iteration to solve multivariate system

To solve a multivariate system with Newton method we need to review the Taylor expansion of an  $N$  dimensional function defined from  $\mathbb{R}^N$  to  $\mathbb{R}^N$ .

**Definition 7.0.1.** *The  $N$  dimensional multivariate equation system is given by*

$$\begin{aligned} F_1(x_1, x_2, \dots, x_N) &= F_1(\mathbf{x}) = 0 \\ F_2(x_1, x_2, \dots, x_N) &= F_2(\mathbf{x}) = 0 \\ &\vdots \\ F_N(x_1, x_2, \dots, x_N) &= F_N(\mathbf{x}) = 0 \end{aligned}$$

where  $\mathbf{x} = (x_1, x_2, \dots, x_N)$ . We define a vector function  $F(\mathbf{x}) = (F_1(\mathbf{x}), F_2(\mathbf{x}), \dots, F_N(\mathbf{x}))$ , so that the equation system can be written as

$$F(\mathbf{x}) = 0. \quad (7.9)$$

The Newton-Raphson iteration can be generalized to solve  $N$  dimensional system 7.9. We first consider the first-order terms of Taylor expansion series of the  $N$  dimensional function

$$F_i(\mathbf{x} + \Delta\mathbf{x}) = F_i(\mathbf{x}) + \sum_{j=1}^N \frac{\partial F_i}{\partial x_j}(\mathbf{x}) \Delta x_j + O(\Delta\mathbf{x}^2) \quad (7.10)$$

for  $i = 1, \dots, N$  where  $\Delta\mathbf{x} = (\Delta x_1, \Delta x_2, \dots, \Delta x_N)$ .

The  $N$  equations can be written in vector form as

$$F(\mathbf{x} + \Delta\mathbf{x}) = \begin{bmatrix} F_1(\mathbf{x} + \Delta\mathbf{x}) \\ F_2(\mathbf{x} + \Delta\mathbf{x}) \\ \vdots \\ F_N(\mathbf{x} + \Delta\mathbf{x}) \end{bmatrix} \approx \begin{bmatrix} F_1(\mathbf{x}) \\ F_2(\mathbf{x}) \\ \vdots \\ F_N(\mathbf{x}) \end{bmatrix} + \begin{bmatrix} \frac{\partial F_1}{\partial x_1}(\mathbf{x}) & \frac{\partial F_1}{\partial x_2}(\mathbf{x}) & \dots & \frac{\partial F_1}{\partial x_N}(\mathbf{x}) \\ \frac{\partial F_2}{\partial x_1}(\mathbf{x}) & \frac{\partial F_2}{\partial x_2}(\mathbf{x}) & \dots & \frac{\partial F_2}{\partial x_N}(\mathbf{x}) \\ \frac{\partial F_N}{\partial x_1}(\mathbf{x}) & \frac{\partial F_N}{\partial x_2}(\mathbf{x}) & \dots & \frac{\partial F_N}{\partial x_N}(\mathbf{x}) \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_1 \\ \Delta\mathbf{x}_2 \\ \vdots \\ \Delta\mathbf{x}_N \end{bmatrix} = F(\mathbf{x}) + J_F \quad (7.11)$$

where  $J_F(\mathbf{x})$  is  $N \times N$  Jacobian matrix defined over the function vector  $F(\mathbf{x})$ . By assuming that  $F(\mathbf{x} + \Delta\mathbf{x}) = 0$ , we can define the roots  $\mathbf{x} + \Delta\mathbf{x}$ , where  $\delta\mathbf{x}$  can be obtained by solving the following equation

$$\Delta\mathbf{x} = J_F(\mathbf{x})^{-1}[F_i(\mathbf{x} + \Delta\mathbf{x}) - F_i(\mathbf{x})] = J_F(\mathbf{x})^{-1}F(\mathbf{x}). \quad (7.12)$$

And the roots can be found from shifting point  $\mathbf{x}$  as

$$\mathbf{x} + \Delta\mathbf{x} = \mathbf{x} - J_F(\mathbf{x})^{-1}F(\mathbf{x}). \quad (7.13)$$

If the equations are non linear, the result is only an approximation of the real root, which can be improved iteratively as follows

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta\mathbf{x}_n = \mathbf{x}_n - J_F(\mathbf{x}_n)^{-1}F(\mathbf{x}_n). \quad (7.14)$$

Newton iteration can be further generated to solve over-constrained non-linear equation systems with  $N$  unknowns but  $M > N$  equations. In the case the inverse matrix of the  $M \times N$  Jacobian matrix  $J_F$  does not exist, but the  $N \times M$  pseudo-inverse  $J_F^{-1} = (J_F^T J_F)^{-1} J_F^T$  can be used in the Newton iteration.

### Newton iteration to remove perturbation on series

Given a perturbation  $\tilde{\sigma} = \sum_{\alpha} \tilde{\sigma}_{\alpha} \frac{\mathbf{y}^{\alpha}}{\alpha!}$  of a polynomial-exponential series  $\sigma = \sum_{i=1}^r \omega_i \mathbf{e}_{\xi_i}(\mathbf{y})$ , we want to remove the perturbation on  $\tilde{\sigma}$  by computing the polynomial-exponential series of rank  $r$ , which is the closest to the perturbed series  $\tilde{\sigma}$ . Starting from an approximate decomposition, using the previous method on the perturbed data, we apply a Newton-type method to minimize the distance between the input series and a weighted sum of  $r$  exponential terms.

To evaluate the distance between the series, we use the first moments  $\tilde{\sigma}_{\alpha}$  for  $\alpha \in A$ , where  $A$  is a finite subset of  $\mathbb{N}^n$ . For  $\alpha \in A$ , let  $F_{\alpha}(\Xi) = \sum_{i=1}^r \omega_i \xi_i^{\alpha} - \tilde{\sigma}_{\alpha}$  be the error function for the moment  $\tilde{\sigma}_{\alpha}$ , where  $\omega_i, \xi_{i,j}$  are variables. We denote by  $\Xi = (\xi_{i,j})_{1 \leq i \leq r, 0 \leq j \leq n}$  this set of variables, with the convention that  $\xi_{i,0} = \omega_i$  for  $i = 1, \dots, r$ . Let  $I = [1, r] \times [0, n] = \{(i, j) \mid 1 \leq i \leq r, 0 \leq j \leq n\}$  be the indices of the variables and  $N = (n+1)r = |I|$ . We denote by  $F(\Xi) = (F_{\alpha}(\Xi))_{\alpha \in A}$  the vector of these error functions.

We want to minimize the distance

$$E(\Xi) = \frac{1}{2} \sum_{\alpha \in A} |F_{\alpha}(\Xi)|^2 = \frac{1}{2} \|F(\Xi)\|^2.$$

Let  $M(\Xi_i) = [w_i \xi_i^\alpha]_{\alpha \in A}$ . We denote by  $V(\Xi) = (\partial_{(i,j)} M(\Xi_i))_{(i,j) \in I}$  the  $|A| \times N$  Vandermonde-like matrix, which columns are the vectors  $\partial_{(i,j)} M(\Xi_i)$ . The gradient of  $E(\Xi)$  is

$$\nabla E(\Xi) = (\langle \partial_{(i,j)} M(\Xi_i), F(\Xi) \rangle)_{(i,j) \in I} = V(\Xi)^T F(\Xi)$$

where  $\partial_{(i,j)}$  is the derivation with respect to  $\Xi_{i,j}$  for  $(i,j) \in I$ . We denote by  $V(\Xi) = (\partial_{(i,j)} M(\Xi_i))_{(i,j) \in I}$  the  $|A| \times N$  Vandermonde-like matrix, which columns are the vectors  $\partial_{(i,j)} M(\Xi_i)$ ,  $(i,j) \in I$ .

To find a local minimizer of  $E(\Xi)$ , we compute a solution of the system  $\nabla E(\Xi) = 0$ , by Newton method. The Jacobian of  $\nabla E(\Xi)$  with respect to the variables  $\Xi$  is

$$\begin{aligned} J_{\Xi}(\nabla E) &= \langle \partial_{(i,j)} M(\Xi_j), \partial_{(i',j')} M(\Xi_{j'}) \rangle + (\langle \partial_{(i,j)} \partial_{(i',j')} M(\Xi_i), F(\Xi) \rangle)_{(i,j) \in I, (i',j') \in I} \\ &= V(\Xi)^T V(\Xi) + (\langle \partial_{(i,j)} \partial_{(i',j')} M(\Xi_i), F(\Xi) \rangle)_{(i,j) \in I, (i',j') \in I}. \end{aligned}$$

Notice that  $\partial_{(i,j)} \partial_{(i',j')} M(\Xi_i) = 0$  if  $i \neq i'$  so that the second matrix is a block diagonal matrix. Then, Newton iteration takes the form:

$$\Xi_{n+1} = \Xi_n - J_{\Xi}(\nabla E)^{-1} \nabla E(\Xi_n).$$

To study the numerical influence of Newton method, we compute the maximum absolute error between the input frequencies  $\tilde{\xi}_i$  and the output frequencies  $\tilde{\xi}_i$ , and the maximum error between the input weights  $\omega_i$  and the output weights  $\tilde{\omega}_i$  as in (2.8).

Figures 7.1a and 7.1b show that Newton iterations improve the error. The error decreases by a factor of  $\approx 10^2$  compared to the computation without Newton iterations. In Figure 7.1b for the amplitude of frequencies  $M = 100$ , the error is smaller than without Newton iterations by a similar order of magnitude (see in Figure 2.1b).

## Numerical Example

In this section, we give an example where the direct decomposition does not restore the initial values of weights and points. Starting from an approximate decomposition, we apply newton iteration described in 7 to recover the initial

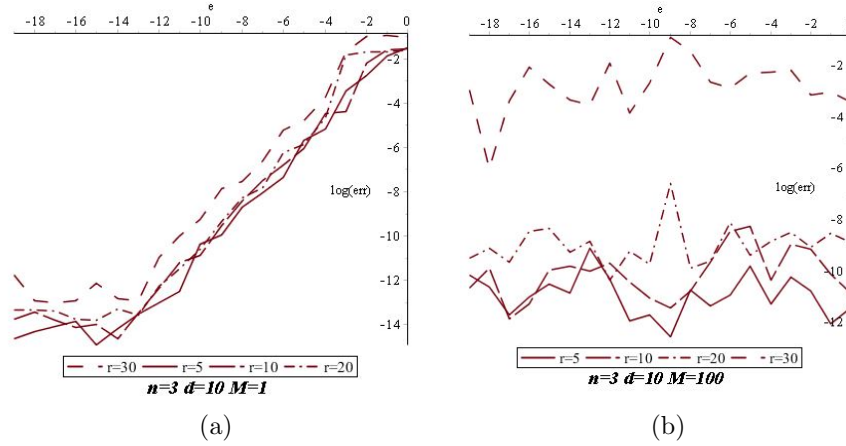


Figure 7.1: Newton influence with 5 iterations

weights and points. Suppose that  $r = 4$  and all corresponding weights are equal to 100.0, we generate the initial frequencies as follows:

$$\xi_0 = \begin{bmatrix} 0.3952 & 0.4310 & 0.0723 & 0.1359 \\ 0.0094 & 0.5427 & 0.0579 & 0.5034 \end{bmatrix}$$

Each column of  $\xi_0$  corresponds to a point in  $\mathbb{R}^2$ .

We compute the polynomial-exponential series of order  $d = 4$  in two variables  $x_1$  and  $x_2$  associated to  $\xi_0$  and  $\omega_0$  which is equal to

$$\sigma_0(x_1, x_2) = 14.4756x_1^3 + 16.1714x_1x_2^2 + 5.9452x_1^2x_2^2 + 36.5794x_1^4 + 4.5347x_1^3x_2 + 31.0352x_1x_2 + 11.1944x_1^2x_2 + 28.7643x_2^3 + 55.1440x_2^2 + 8.6275x_1x_2^3 + 103.4691x_1 + 15.1002x_2^4 + 111.3600x_2 + 5.9311x_1^4 + 400.0.$$

If we want to apply the decomposition algorithm 1, we have  $d_1 = d_2 = 1$  and  $A_1 = (1, x_1, x_2)$  and  $A_2 = (1, x_1, x_2)$  so that  $\text{rank}(H_{\sigma_0}^{d_1, d_2}) \leq 3$ . Therefore, for  $r = 4$  we need to complete the series in order to be able to compute all multiplication matrices.

We complete this series using the Uzawa's iteration described in (6.32) in degree 4 and we decompose the full series as a weighted sum of 4 frequencies using the decomposition algorithm 1. We obtain the following points and weights:

$$\xi_1 = \begin{bmatrix} 0.4404 & 0.1324 & 0.6729 & 0.1437 \\ 0.5359 & 0.4976 & -0.0390 & 0.0410 \end{bmatrix}$$

$$\omega_1 = \begin{bmatrix} 112.805 \\ 87.1916 \\ 174.075 \\ 25.928 \end{bmatrix}$$

Each column of  $\xi_1$  corresponds to a point and each element of  $\omega_1$  corresponds to its weight. We notice that the direct decomposition does not give the initial points  $\xi_0$  and weights  $\omega_0$ .

We apply newton iteration described in 7 and we restore the initial points and weights in 25 iterations. We fix the convergence step to  $s = e^{-15}$ . Each  $\delta$



is the absolute value of moving step of each iteration.

|                       |                          |
|-----------------------|--------------------------|
| $ s  = 119.4744$      | $\delta = 188.5795$      |
| $ s  = 15.7697$       | $\delta = 13.9976$       |
| $ s  = 1.2427$        | $\delta = 46.4228$       |
| $ s  = 2.5142$        | $\delta = 45.1491$       |
| $ s  = 0.6116$        | $\delta = 5.7567$        |
| $ s  = 0.0479$        | $\delta = 0.3559$        |
| $ s  = 2.8825e^{-5}$  | $\delta = 7.4204e^{-5}$  |
| $ s  = 3.6293e^{-11}$ | $\delta = 1.6331e^{-10}$ |
| $ s  = 5.9215e^{-14}$ | $\delta = 9.8644e^{-13}$ |
| $ s  = 1.7153e^{-14}$ | $\delta = 1.0727e^{-12}$ |
| $ s  = 9.8102e^{-15}$ | $\delta = 1.1731e^{-12}$ |
| $ s  = 9.4414e^{-15}$ | $\delta = 2.8450e^{-13}$ |
| $ s  = 6.2204e^{-14}$ | $\delta = 2.0730e^{-13}$ |
| $ s  = 3.3445e^{-14}$ | $\delta = 2.0649e^{-13}$ |
| $ s  = 1.9275e^{-14}$ | $\delta = 1.4634e^{-13}$ |
| $ s  = 9.3152e^{-15}$ | $\delta = 5.6033e^{-13}$ |
| $ s  = 1.7359e^{-14}$ | $\delta = 8.4691e^{-13}$ |
| $ s  = 6.1412e^{-14}$ | $\delta = 5.3685e^{-13}$ |
| $ s  = 1.8093e^{-14}$ | $\delta = 4.2956e^{-13}$ |
| $ s  = 2.3515e^{-14}$ | $\delta = 3.5522e^{-13}$ |
| $ s  = 1.6687e^{-14}$ | $\delta = 1.7530e^{-13}$ |
| $ s  = 2.2938e^{-14}$ | $\delta = 1.8702e^{-13}$ |
| $ s  = 1.8841e^{-14}$ | $\delta = 2.2191e^{-13}$ |
| $ s  = 1.8841e^{-14}$ | $\delta = 3.3689e^{-13}$ |
| $ s  = 2.0905e^{-14}$ | $\delta = 2.0093e^{-13}$ |

The following points and weights are the result of Newton iteration and they are almost equal to the initial ones.

$$\xi_2 = \begin{bmatrix} 0.4310 & 0.1359 & 0.3952 & 0.0723 \\ 0.5427 & 0.5034 & 0.0094 & 0.0579 \end{bmatrix}$$

Each column of  $\xi_2$  corresponds to a point of 4 points.

$$\omega_2 = \begin{bmatrix} 100.0 \\ 100.0 \\ 100.0 \\ 100.0 \end{bmatrix}$$

Each element of  $\omega_2$  corresponds to a weight of 4 weights.

The error on the series before and after completion and Newton step is  $1.6377e^{-14}$ .

# Chapter 8

## Conclusions and Perspectives

We study the decomposition of a Hankel matrix  $H_\sigma$  as a sum of indecomposable Hankel matrices of low rank using a numerically efficient direct method which uses the Singular Value Decomposition of a truncated Hankel matrix to compute orthogonal bases of the quotient algebra  $\mathcal{A}_\sigma$  associated to the decomposition. We associate to the decomposition of a Hankel matrix  $H_\sigma$ , the decomposition of its symbol  $\sigma$  as a weighted sum of evaluations. We compute the multiplication matrices  $M_{x_i}$  by variables  $x_i$  in the orthogonal basis of the dual quotient algebra  $\mathcal{A}_\sigma^*$  from sub-matrices of the Hankel matrix. We use the eigenvectors of multiplication matrices to recover the weights and points. We analyze its numerical behavior in the presence of noise on the coefficients of multivariate series  $\sigma(x_1, x_2, \dots, x_n) \in \mathbb{K}[x_1, x_2, \dots, x_n]$  while changing the dimension of ring of polynomials, the degree of series and the minimal number of terms in the decomposition. We also analyze its numerical consistency when the points are of high amplitude and propose a rescaling technique to solve it.

We study the decomposition of symmetric, multi-linear and multi-symmetric tensors as a sum of tensors of low rank of the same type. We adapt the technique of decomposition of Hankel matrix to the tensor case which referred to the fact that the coefficients of a tensor can be computed using the coefficients of a formal power series and therefore to decompose a series is equivalent to decompose the dual of the tensor associated to it. In the multi-symmetric case, we use three groups of variables  $\mathbf{x}, \mathbf{y}$  and  $\mathbf{z}$  instead of only one and we use the eigenvectors of multiplication matrices by  $y_j$  to compute the  $y$ 's coordinates and we deduce then the  $x$ 's and  $z$ 's coordinates.

We extract the reduced tensor structure from a statistical LDA model which analyzes the content of web pages. We describe the amount of 3 different topics in a corpus of documents and the probability that each word of the associated vocabulary referred to this topic. We adapt the symmetric tensor decomposition problem to the model to deduce the description of the content of the corpus.

We use the moments of ODF tensors which are computed using the physical model described in 8 as input data of symmetric tensor decomposition algorithm. We compute the two and three weighted directions of fibers of 16 neurons in three dimensional space. We recover the initial angle between directions when it is equal or larger than  $30^\circ$ .

We study the completion problem of a Hankel matrix of low rank as a minimization problem RMP. We propose two heuristics which rely on the minimization of the trace and the norm of a Hankel matrix. We show a numerical example which completes a matrix of rank 5 with few elements in order to be Hankel such that the associated series in two dimensional space is of order 2. The output matrix is of rank 2, we test its Hankel structure by decomposing it into 2 Hankel matrices of rank one.

We adapt the SVT iterative algorithm to the Hankel case which applies a singular value thresholding step at each iteration such that the threshold is big enough and guarantee theoretically the convergence. It is referred to the computation of the linear operator which describes the constraints of the minimization problem in the Hankel case and its adjoint. We give a numerical example which completes a matrix with few elements in order to be Hankel and to save the values of these elements. The series in two dimensional space associated to the completed matrix is of order 4. The SVT algorithm converges to a Hankel matrix such that the convergence error is very low.

We present a new Newton iteration which minimizes the distance between the completed Hankel matrix and its representation as a weighted sum of Hankel matrices of rank one. We also use this method to remove the perturbation on the input coefficients.

We look for a new technique which solves the numerical instability of the decomposition problem when the multiplicities of points are more than one and the singular value decomposition is not working anymore. We try to

show the utility of the decomposition algorithm in many other applications such that the phylogenetic trees and image processing where the close relationship between local image structure and apparent diffusion of the tensor makes this image modality very interesting for medical image analysis such as characterizing the local structure in tissues such as white matter in brain. We try to analyze the content of other corpus where the rank of the reduced tensor associated to the LDA model is bigger than the dimension. We aim to study the behavior of our technique for ODF tensors with noised entries.

# Bibliography

- [AFH<sup>+</sup>12] Anima Anandkumar, Dean P. Foster, Daniel J. Hsu, Sham M. Kakade, and Yi-Kai Liu. A spectral algorithm for latent dirichlet allocation. In *Advances in Neural Information Processing Systems*, pages 917–925, 2012.
- [AGH<sup>+</sup>15] Anima Anandkumar, Rong Ge, Daniel Hsu, Sham M. Kakade, and Matus Telgarsky. Tensor decompositions for learning latent variable models (A survey for ALT). In *International Conference on Algorithmic Learning Theory*, pages 19–38. Springer, 2015.
- [AHK12] Animashree Anandkumar, Daniel Hsu, and Sham M. Kakade. A method of moments for mixture models and hidden Markov models. In *Conference on Learning Theory*, pages 33–1, 2012.
- [Aus08] Tim Austin. On exchangeable random variables and the statistics of large graphs and hypergraphs. *Probability Surveys*, 5:80–145, 2008.
- [BBCM13] Alessandra Bernardi, Jerome Brachat, Pierre Comon, and Bernard Mourrain. General tensor decomposition, moment matrices and applications. *Journal of Symbolic Computation*, 52:51–71, 2013.
- [BCL17] Matteo Briani, Annie Cuyt, and Wen-shin Lee. A hybrid Fourier-Prony method. *arXiv preprint arXiv:1706.04520*, 2017.
- [BDHM17] Alessandra Bernardi, Noah S. Daleo, Jonathan D. Hauenstein, and Bernard Mourrain. Tensor decomposition and homotopy

- continuation. *Differential Geometry and its Applications*, 55:78–105, 2017.
- [Ber08] Alessandra Bernardi. Ideals of varieties parameterized by certain symmetric tensors. *Journal of Pure and Applied Algebra*, 212(6):1542–1559, 2008.
- [BML94] Peter J. Basser, James Mattiello, and Denis LeBihan. Estimation of the effective self-diffusion tensor from the NMR spin echo. *Journal of Magnetic Resonance, Series B*, 103(3):247–254, 1994.
- [BP12] Dario Bini and Victor Y. Pan. *Polynomial and Matrix Computations: Fundamental Algorithms*. Springer Science & Business Media, 2012.
- [Bro97] Rasmus Bro. PARAFAC. Tutorial and applications. *Chemometrics and intelligent laboratory systems*, 38(2):149–171, 1997.
- [BV04] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge university press, 2004.
- [CCS10] Jian-Feng Cai, Emmanuel J. Candès, and Zuowei Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.
- [CGT90] A. R. Conn, N. I. M. Gould, and Ph L. Toint. An introduction to the structure of large scale nonlinear optimization problems and the LANCELOT project. *Computing Methods in Applied Sciences and Engineering*, pages 42–54, 1990.
- [CL18] Annie Cuyt and Wen-shin Lee. Multivariate exponential analysis from the minimal number of samples. *Advances in Computational Mathematics*, 44(4):987–1002, August 2018.
- [CR09] Emmanuel J. Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717, 2009.

- [CT10] Emmanuel J. Candès and Terence Tao. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.
- [CTVL18] Annie Cuyt, Min-nan Tsai, Marleen Verhoye, and Wen-shin Lee. Faint and clustered components in exponential analysis. *Applied Mathematics and Computation*, 327:93–103, 2018.
- [DL97] Lieven De Lathauwer. *Signal Processing Based on Multilinear Algebra*. Katholieke Universiteit Leuven Leuven, 1997.
- [DLC07] Lieven De Lathauwer and Joséphine Castaing. Tensor-based techniques for the blind separation of DS–CDMA signals. *Signal Processing*, 87(2):322–336, 2007.
- [DLDMV04] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. Computation of the canonical decomposition by means of a simultaneous generalized Schur decomposition. *SIAM journal on Matrix Analysis and Applications*, 26(2):295–327, 2004.
- [EM07] Mohamed Elkadi and Bernard Mourrain. *Introduction à La Résolution Des Systèmes Polynomiaux*, volume 59. Springer Science & Business Media, 2007.
- [Ems78] Jacques Emsalem. Géométrie des points épais. *Bull. Soc. Math. France*, 106(4):399–416, 1978.
- [Faz02] Maryam Fazel. *Matrix Rank Minimization with Applications*. PhD Thesis, PhD thesis, Stanford University, 2002.
- [GDM09] Aurobrata Ghosh, Rachid Deriche, and Maher Moakher. Ternary quartic approach for positive 4th order diffusion tensors revisited. In *Biomedical Imaging: From Nano to Macro, 2009. ISBI'09. IEEE International Symposium On*, pages 618–621. IEEE, 2009.
- [Gol96] Gene H. Golub. CF Van loan, Matrix computations. *The Johns Hopkins*, 1996.



- [H02] Huy Tài Hà. Box-shaped matrices and the defining ideal of certain blowup surfaces. *Journal of Pure and Applied Algebra*, 167(2-3):203–224, 2002.
- [HA15] Furong Huang and Animashree Anandkumar. Convolutional dictionary learning through tensor factorization. In *Feature Extraction: Modern Questions and Challenges*, pages 116–129, 2015.
- [HK13] Daniel Hsu and Sham M. Kakade. Learning mixtures of spherical gaussians: Moment methods and spectral decompositions. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, pages 11–20. ACM, 2013.
- [HKM18] J. Harmouch, H. Khalil, and B. Mourrain. Structured low rank decomposition of multivariate Hankel matrices. *Linear Algebra and its Applications*, 542:162 – 185, 2018.
- [HMK17] Jouhayna Harmouch, Bernard Mourrain, and Houssam Khalil. Decomposition of Low Rank Multi-Symmetric Tensor. In *International Conference on Mathematical Aspects of Computer and Information Sciences*, pages 51–66. Springer, 2017.
- [Hos16] Seyedroohollah Hosseini. On the Convex Geometry of Weighted Nuclear Norm Minimization. *arXiv preprint arXiv:1609.05944*, 2016.
- [HT04] Hakop A. Hakopian and Mariam G. Tonoyan. Partial differential analogs of ordinary differential equations and systems. *New York J. Math*, 10:89–116, 2004.
- [JGJJ11] Fangxiang Jiao, Yaniv Gur, Chris R. Johnson, and Sarang Joshi. Detection of crossing white matter fibers with high-order tensors and rank-k decompositions. In *Biennial International Conference on Information Processing in Medical Imaging*, pages 538–549. Springer, 2011.
- [KB09] T. Kolda and B. Bader. Tensor Decompositions and Applications. *SIAM Review*, 51(3):455–500, August 2009.

- [KMR<sup>+</sup>01] Serge Kruk, Masakazu Muramatsu, Franz Rendl, Robert J. Vanderbei, and Henry Wolkowicz. The Gauss-Newton direction in semidefinite programming. *Optimization Methods and Software*, 15(1):1–28, 2001.
- [KPRv16] Stefan Kunis, Thomas Peter, Tim Römer, and Ulrich von der Ohe. A multivariate generalization of Prony’s method. *Linear Algebra and its Applications*, 490:31–47, 2016.
- [Kro81] Leopold Kronecker. *Zur Theorie Der Elimination Einer Variablen Aus Zwei Algebraischen Gleichungen von L. Kronecker...* G. Vogt, 1881.
- [LCO92] John Little, D. Cox, and D. O’Shea. Ideals, varieties, and algorithms. *Contemporary Mathematics*, (133), 1992.
- [Lim05] Lek-Heng Lim. Singular values and eigenvalues of tensors: A variational approach. In *1st IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, 2005.*, pages 129–132. IEEE, 2005.
- [MKBD14] Thinhinane Megherbi, Mouloud Kachouane, Fatima Oulebsir Boumghar, and Rachid Deriche. Détection des croisements de fibre en IRM de diffusion par décomposition de tenseur: Approche analytique. In *Reconnaissance de Formes et Intelligence Artificielle (RFIA) 2014*, 2014.
- [Mou97] Bernard Mourrain. Isolated points, duality and residues. *Journal of Pure and Applied Algebra*, 117:469–493, 1997.
- [Mou16] Bernard Mourrain. Polynomial–Exponential Decomposition From Moments. *Foundations of Computational Mathematics*, pages 1–58, 2016.
- [NW99] Jorge Nocedal and Stephen J. Wright. *Springer Series in Operations Research. Numerical Optimization*. New York: Springer, 1999.

- [OP01] Ulrich Oberst and Franz Pauer. The constructive solution of linear systems of partial difference and differential equations with constant coefficients. *Multidimensional Systems and Signal Processing*, 12(3-4):253–308, 2001.
- [Pan16] Victor Y. Pan. How bad are Vandermonde matrices? *SIAM Journal on Matrix Analysis and Applications*, 37(2):676–694, 2016.
- [Ped99] Paul S. Pedersen. Basis for power series solutions to systems of linear, constant coefficient partial differential equations. *Advances in Mathematics*, 141(1):155–166, 1999.
- [PS10] Victor Pereyra and Godela Scherer. *Exponential Data Fitting and Its Applications*. Bentham Science Publishers, 2010.
- [PT13] Daniel Potts and Manfred Tasche. Parameter estimation for multivariate exponential sums. *Electronic Transactions on Numerical Analysis*, 40:204–224, 2013.
- [PT14] Gerlind Plonka and Manfred Tasche. Prony methods for recovery of structured functions. *GAMM-Mitteilungen*, 37(2):239–258, 2014.
- [Puc98] Mario Pucci. The Veronese variety and catalecticant matrices. *Journal of Algebra*, 202(1):72–95, 1998.
- [Qi05] Liqun Qi. Eigenvalues of a real supersymmetric tensor. *Journal of Symbolic Computation*, 40(6):1302–1324, 2005.
- [RFP10] Benjamin Recht, Maryam Fazel, and Pablo A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.
- [RK90] Richard H. Roy and Thomas Kailath. ESPRIT—Estimation of signal parameters via rotational invariance techniques. *Optical Engineering*, 29(4):296–314, 1990.
- [Sau17] Tomas Sauer. Prony’s method in several variables. *Numerische Mathematik*, 136(2):411–438, 2017.

- [Sau18] Tomas Sauer. Prony’s method in several variables: Symbolic solutions by universal interpolation. *Journal of Symbolic Computation*, 84:95–112, 2018.
- [SGB00] Nicholas D. Sidiropoulos, Georgios B. Giannakis, and Rasmus Bro. Blind PARAFAC receivers for DS-CDMA systems. *IEEE Transactions on Signal Processing*, 48(3):810–823, 2000.
- [SK90] Eugenio Sanchez and Bruce R. Kowalski. Tensorial resolution: A direct trilinear decomposition. *Journal of Chemometrics*, 4(1):29–45, 1990.
- [SK92] A. Lee Swindlehurst and Thomas Kailath. A performance analysis of subspace-based methods in the presence of model errors, Part I: The MUSIC algorithm. *IEEE Trans. Signal Processing*, 40(7):1758–1774, 1992.
- [Ste10] Alwin Stegeman. On uniqueness of the  $n$  th order tensor decomposition into rank-1 terms with linear independence in one mode. *SIAM Journal on Matrix Analysis and Applications*, 31(5):2498–2516, 2010.
- [Ste11] Alwin Stegeman. On uniqueness of the canonical tensor decomposition with some form of symmetry. *SIAM Journal on Matrix Analysis and Applications*, 32(2):561–583, 2011.
- [TCC07] J.-Donald Tournier, Fernando Calamante, and Alan Connelly. Robust determination of the fibre orientation distribution in diffusion MRI: Non-negativity constrained super-resolved spherical deconvolution. *Neuroimage*, 35(4):1459–1472, 2007.
- [Tod97] Michael J. Todd. A study of search directions in primal-dual interior-point methods for semidefinite programming. Technical report, Cornell University Operations Research and Industrial Engineering, 1997.
- [UC15] Konstantin Usevich and Pierre Comon. Quasi-Hankel low-rank matrix completion: A convex relaxation. *arXiv preprint arXiv:1505.07766*, 2015.

- [VB96] Lieven Vandenberghe and Stephen Boyd. Semidefinite programming. *SIAM review*, 38(1):49–95, 1996.
- [WBA12] Yonas T. Weldeselassie, Angelos Barmoutis, and M. Stella Atkins. Symmetric positive semi-definite cartesian tensor fiber orientation distributions (CT-FOD). *Medical Image Analysis*, 16(6):1121–1129, 2012.
- [XY13] Yangyang Xu and Wotao Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789, 2013.
- [ZLNMA04] Andreas Ziehe, Pavel Laskov, Guido Nolte, and Klaus-Robert MÅžller. A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation. *Journal of Machine Learning Research*, 5(Jul):777–800, 2004.