

KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELECTROTECHNIEK
AFDELING ESAT
Kardinaal Mercierlaan 94 — 3001 Leuven (Heverlee)

SUBSPACE-BASED PARAMETER ESTIMATION OF EXPONENTIALLY DAMPED SINUSOIDS WITH APPLICATION TO NUCLEAR MAGNETIC RESONANCE SPECTROSCOPY DATA

Promotors:
Prof. Dr. Ir. S. Van Huffel
Prof. Dr. Ir. J. Vandewalle
Prof. Dr. P. Van Hecke

Thesis submitted to obtain the degree of doctor in applied sciences

by

Hua CHEN

May 1996



KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELECTROTECHNIEK
AFDELING ESAT
Kardinaal Mercierlaan 94 — 3001 Leuven (Heverlee)

SUBSPACE-BASED PARAMETER ESTIMATION OF EXPONENTIALLY DAMPED SINUSOIDS WITH APPLICATION TO NUCLEAR MAGNETIC RESONANCE SPECTROSCOPY DATA

Jury:
Chairman: Prof. Dr. Ir. W. Dutré, vice-dean
Prof. Dr. Ir. S. Van Huffel, promotor
Prof. Dr. Ir. J. Vandewalle, promotor
Prof. Dr. P. Van Hecke, promotor
Dr. Ir. M. Moonen
Prof. Dr. Ir. D. van Ormondt
Prof. Dr. Ir. P. Van Dooren
Prof. Dr. Ir. P. Suetens

Thesis submitted to obtain the degree of doctor in applied sciences

by

Hua CHEN

UDC: 621.395.38(043) May 1996

© Katholieke Universiteit Leuven - Faculteit Toegepaste Wetenschappen
Arenbergkasteel, B-3001 Heverlee (Belgium)

Alle rechten voorbehouden. Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotocopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaandelijke schriftelijke toestemming van de uitgever.

All rights reserved. No part of this publication may be reproduced in any form by print, photoprint, microfilm or any other means without written permission from the publisher.

D/1996/7515/16

ISBN 90-5682-030-3

Acknowledgments

I would like to express my gratitude to my daily supervisor, Prof. S. Van Huffel, of the ESAT-SISTA group, for her invaluable guidance, insightful suggestions and enlightening discussions. I am heavily indebted to her for her painstaking reading and revising of all my papers, as well as the drafts of this thesis.

I would also like to thank Prof. J. Vandewalle, head of ESAT-SISTA, who provides me with the opportunity to study and work in the nice environment of ESAT-SISTA, K. U. Leuven. His constant encouragement and enthusiasm is very important to the accomplishment of this work.

My thanks also go to Prof. P. Van Hecke from the NMR lab of the Gasthuisberg hospital, for providing NMR data for analysis, suggesting the problem, and careful reading of my papers and thesis.

A special note of thanks goes to Prof. D. Van Ormondt from the Applied Physics laboratory of Delft University of Technology for his novel ideas, cheerful discussions, and valuable advice on fitting NMR data offered.

As well, Dr. M. Moonen is gratefully acknowledged for his critical review and constructive comments which made this thesis more readable.

I am truly grateful to all the staff and my fellow students, Belgian and Chinese, for their help and assistance in various way.

Last but not least, I would like to thank my husband, Jianbin, and our children, for their love, support, encouragement and tolerance over the last few years.

Hua Chen

Abstract

This dissertation treats the issue of subspace-based parameter estimation for data modeled as a sum of exponentially damped complex-valued exponentials in the context of NMR spectroscopy where the models are of the form

$$x_n = \sum_{k=1}^{K} a_k e^{j\varphi_k} e^{(-d_k + j2\pi f_k)n\Delta t} + e_n \qquad n = 0, \ldots, N-1$$

Firstly, aiming at emphasizing the sinusoids of interest and suppressing the unwanted sinusoids and noise, prefiltering methods are presented using FIR or IIR filter matrices prior to an SVD and subspace-based method. A theoretical analysis on special cases of one and two exponentially damped sinusoids is given, which gives the increment of the SNR obtained from prefiltering and reveals the relationship between the singular values/vectors of the prefiltered and original data matrices.

Secondly, an enhancement algorithm, based on the minimum variance estimation method, is presented for cleaning up noisy data before estimating signal parameters using a subspace-based method. This algorithm is superior to Cadzow's and non-enhanced algorithms in both resolution and accuracy performance. The convergence of these enhancement procedures is found to be unnecessary when followed by an SVD and subspace-based estimator such as HTLS, offering drastic reduction in the computational cost. The improvement in resolution and computational efficiency makes the minimum variance enhanced estimation method well suited to the quantitative time-domain analysis of NMR measurement data.

Thirdly, the structure-preserving total least norm technique, called STLN, and the structured total least squares technique for complex-valued Hankel data matrices are presented and discussed. Applying STLN as signal enhancement prior to the estimation approach of HTLS results in a more accurate estimator, HTLN, which removes corruption noise effectively and improves resolution and parameter accuracy, especially, the standard deviations of the parameters.

Finally, prior knowledge of some known signal poles is incorporated into parameter estimators like HTLS and HTLN, etc. The remarkable advantages of the new methods are confirmed through quantification of both simulated and *in vivo* NMR signals, especially when the known signal poles are close to the unknown ones in the spectrum. Furthermore, the incorporation of prior knowledge is interpreted in terms of prefiltering operation, which enables intuitive analysis in the frequency domain.

List of Abreviations

Abreviation	Meaning
SVD	singular value decomposition
LS	least squares
TLS	total least squares
LCTLS	linearly-constrained total least squares
NMR	nuclear magnetic resonance
FID	free induction decay
^{31}P	phosphorus with atomic mass 31
PDE	phospho-diesters
PME	phospho-monoesters
P_i	inorganic phosphate
NTP	nucleotide triphosphate
ATP	adenosine triphosphate
NAD	nicotine adenine dinucleotide
DFT	discrete Fourier transform
FFT	fast Fourier transform
SNR	signal-to-noise ratio
RMSE	root mean-squared error
CR	Cramér-Rao (bound)
HTLS	Hankel TLS (technique)
HSVD	Hankel SVD (technique)
VARPRO	variable projection (technique)
LP	linear prediction (technique)
FLP	forward linear prediction (technique)
BLP	backward linear prediction (technique)
MP	matrix pencil (technique)
FIR	finite impulse response
IIR	infinite impulse response
$HTLS_F$	HTLS prefiltering (technique)
MV	minimum variance (method)
CA	Cadzow's (method)
CTLS	constrained total least squares (technique)
STLS	structured total least squares (technique)
STLN	structured total least norm (technique)
HTLN	STLN-enhanced HTLS (technique)

Abreviation	Meaning
PK	prior knowledge
LTI	linear time-invariant
a.u.	arbitrary units

List of Important Notations

Notation	Meaning
j	$\sqrt{-1}$
x	scalar
$\lfloor x \rfloor$	truncate a scalar x to a closest smaller integer
$\mathbf{x}$	vector
$\mathbf{X}$	matrix
$\|\mathbf{X}\|$	unitarily invariant norm, say, $\| \|_F$ and $\| \|_2$, of matrix $\mathbf{X}$
$\mathbf{X}^*$	conjugate of matrix $\mathbf{X}$
$\mathbf{X}^T$	transpose of matrix $\mathbf{X}$
$\mathbf{X}^H$	Hermitian conjugate of matrix $\mathbf{X}$
$\mathbf{X}^\uparrow$	matrix $\mathbf{X}$ of which the top row is deleted
$\mathbf{X}_\downarrow$	matrix $\mathbf{X}$ of which the bottom row is deleted
$\mathcal{R}(\mathbf{X})$	range of matrix $\mathbf{X}$
$\mathcal{N}(\mathbf{X})$	null space of matrix $\mathbf{X}$
$\mathbb{R}^{L\times M}$	set of all $L \times M$ real-valued matrices
$\mathbb{C}^{L\times M}$	set of all $L \times M$ complex-valued matrices
$\overset{\circ}{x}_n$	the nth noise-free data sample
e_n	the nth noise sample
x_n	the nth noisy data sample such that $x_n = \overset{\circ}{x}_n + e_n$
$\tilde{x}_n$	the nth filtered data sample
K	number of exponentials in the data model
N	number of data samples
L	number of rows in matrix $\mathbf{X}$
M	number of columns in matrix $\mathbf{X}$
f_k	frequency of the kth exponential in Hz
d_k	damping factor of the kth exponential in rad/s
a_k	amplitude of the kth exponential in a.u.
$\varphi_k(\phi_k)$	phase of the kth exponential in radians (degrees, °)
c_k	complex-valued amplitude of the kth exponential
z_k	signal pole of the kth exponential
f_s	sampling frequency
Δt	sampling time interval
t_n	the nth time sample

Notation	Meaning
$\mathbf{U}$	unitary matrix of left singular vectors $\mathbf{u}_i$
$\mathbf{V}$	unitary matrix of right singular vectors $\mathbf{v}_i$
$\mathbf{\Sigma}$	diagonal matrix of singular values σ_i in decreasing order of magnitude
σ_ν	standard deviation of real or imaginary part of white noise
ν	standard deviation of white noise ($\nu = 2\sqrt{\sigma_\nu}$, for complex-valued white noise)
$\mathbf{H}$	filter matrix
$\mathbf{W}, \mathbf{D}$	weighting matrices
$\mathcal{X}(m)$	discrete Fourier transform of data vector $\mathbf{x}$
$\mathrm{X}(\mathrm{z})$	z-transform of data vector $\mathbf{x}$

Contents

Acknowledgments i

Abstract ii

Abreviations iv

Notations vi

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Parameter estimation in NMR spectroscopy . . . 3
- 1.3 Parameter estimation in other applications . . . 8
- 1.4 Thesis outline . . . 9

2 Subspace-based Parameter estimation 13
- 2.1 Parameter estimation techniques . . . 13
  - 2.1.1 Time-domain methods versus frequency-domain methods 14
  - 2.1.2 Noninteractive methods versus interactive methods . . . 15
- 2.2 Subspace-based parameter estimation . . . 16
  - 2.2.1 SVD and signal subspace estimation . . . 16
  - 2.2.2 HTLS: a subspace-based parameter estimation algorithm 20
- 2.3 Monte-Carlo simulation procedure . . . 23

3 Prefiltering techniques 25
- 3.1 Hankel matrix and filtering . . . 26
- 3.2 FIR and IIR filter matrices . . . 29
  - 3.2.1 FIR filter matrix: $\mathbf{H}_{\text{FIR}}$ . . . 29
  - 3.2.2 IIR filter matrix: $\mathbf{H}_{\text{IIR}}$ . . . 32
- 3.3 Prefiltered HTLS . . . 35
- 3.4 SVD relationships between the prefiltered and original data matrices . . . 37
  - 3.4.1 Definitions and implications . . . 37
  - 3.4.2 SVD expressions of $\mathbf{X}$ and $\bar{\mathbf{X}}$ in terms of signal parameters 40
  - 3.4.3 SVD relationships between $\mathbf{X}$ and $\bar{\mathbf{X}}$ . . . 41
- 3.5 SNR relationships between the prefiltered and original data matrices . . . 50
- 3.6 HTLS_F prefiltering . . . 52
- 3.7 Simulations . . . 54
- 3.8 Conclusions . . . 65

4 Minimum variance signal enhancement 67
- 4.1 Introduction . . . 68
- 4.2 Enhancement algorithms . . . 69
  - 4.2.1 Cadzow's algorithm . . . 70
  - 4.2.2 Minimum variance algorithm . . . 71
  - 4.2.3 Composite property mapping . . . 72
- 4.3 Computational complexity . . . 73
- 4.4 Simulation results . . . 76
- 4.5 Application to real-world NMR signals . . . 87
- 4.6 Conclusions . . . 90

5 Structured TLN and structured TLS in signal enhancement 92
- 5.1 Formulation of STLN and STLS . . . 93
- 5.2 STLN and STLS algorithms applied to complex Hankel/Toeplitz data matrices . . . 95
  - 5.2.1 The STLN algorithms applied to complex Hankel data matrices . . . 95
  - 5.2.2 The STLS algorithm applied to complex Hankel data matrices . . . 101
  - 5.2.3 Discussion . . . 104
- 5.3 New parameter estimator HTLN . . . 106
- 5.4 Numerical comparison of different algorithms . . . 109
  - 5.4.1 Optimality of the STLN, STLS solutions . . . 109
  - 5.4.2 Simulation results . . . 113
- 5.5 Conclusion . . . 117

6 Incorporation of prior knowledge of known signal poles 120
- 6.1 Brief review of some parameter estimators . . . 121
- 6.2 Prior-knowledge-incorporating estimators . . . 123

6.3 Simulation results 131
6.4 Conclusion 139

7 More analysis of HTLS(PK) with application to NMR 141
7.1 HTLS(PK) and prefiltering 141
7.2 Quantification of simulated signals 145
7.2.1 Comparison of HTLS(PK) and prefiltered HTLS 145
7.2.2 Enhancement of HTLS(PK) 147
7.2.3 Computational efficiency of HTLS(PK) 147
7.2.4 Sensitivity of HTLS(PK) w.r.t. the accuracy of given poles 149
7.3 Quantification of a real-world NMR signal 154
7.4 Conclusion 159

8 General Conclusions 160
8.1 Summary 160
8.2 Suggestions for further research 164

A Simulated signals 166

B Publications 176

List of tables 179

List of figures 182

Bibliography 187

Chapter 1

Introduction

In this opening chapter, we first motivate our study of subspace-based estimation techniques. The data model and its parameters are described thereafter. Finally we give a chapter-by-chapter outline of the dissertation.

1.1 Motivation

Parameter estimation of signals modeled as a sum of **exponentially damped sinusoids** embedded in noise is a kernel problem in many signal processing applications, including direction-of-arrival (DOA) estimation [1][2] in sensor systems such as radar, sonar, and electronic surveillance; high-resolution spectral analysis of time series in speech processing and biomedical signal processing [3][4]; the study of transient signals [5], etc. The problem is one of taking measurements and estimating the parameters that describe the system under study, where the functional form of the underlying signal is assumed to be known as a sum of exponentially damped sinusoids.

With the development of the singular value decomposition (SVD) [6] and other numerical techniques, tremendous research efforts have been put on developing **subspace-based parameter estimation** methods. Via the SVD of a properly structured noisy data matrix, a subdivision into a "signal" and a "noise" subspace is possible in subspace-based approaches. The class of SVD and subspace-based methods becomes more and more attractive in parameter estimation problems due to its efficiency and ease of operation (with little human involvement).

The study of subspace-based parameter estimation in this thesis is greatly motivated by the problem of **spectral estimation in Nuclear Magnetic**

Resonance (NMR) spectroscopy, where the measurement data are considered as impulse responses often modeled as complex-valued exponentials corrupted by additive white noise. The objective is to estimate from measurements a set of constant parameters upon which the impulse response modeled as a sum of exponentials depends. Although this model may be inadequate for signals containing unknown non-exponential components, the quantification of these signals can be accommodated by the SVD and subspace-based methods which decompose each non-exponential spectral component into a sum of exponentially damped sinusoids, in contrast to nonlinear least squares methods. Furthermore, the automation of subspace-based methods is especially important for the clinical use of NMR spectroscopy.

In real-world applications, the comprised exponentials are typically grouped together, very close to each other and difficult to resolve. Very often, the signal-to-noise ratio (SNR) is not sufficiently high due to the limitations of the experimental set-up. In some cases, the parameters have to be estimated from short data records. Yet high-resolution and accurate (low-bias, low-variance) estimates are desired in order to retrieve relevant information. Thus the aim of this thesis is to investigate parameter estimation techniques for the complex-valued exponential model mentioned above. The features of the different algorithms to be discussed cover theoretical foundations, computational efficiency of the implementations, and in particular, **resolution and accuracy of the estimated parameters**, which is of crucial importance to the information acquisition in many real-world applications.

One way to improve the resolution and accuracy of the SVD and subspace-based parameter estimation is to apply a "cleaning" process, or what we call **signal enhancement**, to remove noise, measurement distortion, or even theoretical mismatch present in the given data set, before extracting parameters. The enhancement is accomplished through the process of slightly modifying the measured signal so that the modified signal possesses certain attributes or properties and better represents the underlying signal than the original measured signal.

The existing SVD and subspace-based estimation methods do not accommodate prior knowledge and yields automatically all the parameters when fitting noisy data sets composed of K exponentially damped sinusoids. However, in some applications such as NMR spectroscopy, some **prior knowledge** might be available, or only some of the exponentials (fewer than K) are of interest. The resolution and accuracy can be further improved if prior knowledge is incorporated or interesting components are emphasized.

In our study, various parameter estimation algorithms are developed and tested using simulated signals consisting of exponentially damped sinusoids in additive white noise, as well as real-world NMR measurement data sets, to demonstrate their performance. It is to be noted that these estimation algorithms, although applied only to NMR spectroscopy in this dissertation, can also be applied, and are in fact of great value, to many other signal processing applications.

1.2 Parameter estimation in NMR spectroscopy

NMR spectroscopy is a routinely used method to monitor metabolic changes in living organisms in a fully non-invasive fashion. With NMR spectroscopy, a lot of biochemical information, such as the structure, chemical environment, mobility, concentration and reaction rates of the atoms and molecules under investigation, is available which is not accessible in any other way.

Basic principles of NMR [9][10][11]

The principle of NMR is based on a quantum mechanical property of an atomic nucleus, called spin, which denotes the spinning of the nucleus around its own axis. In the absence of an external magnetic field, the nuclei will be spinning in random orientations, with zero net magnetization of the whole sample. Once an external magnetic field is applied, the spinning state is different, associated with energy levels. By the uptake of an energy quantum, which is equal to the energy difference between two spinning states, the nucleus is excited specifically from its lower energy level to its higher level. The energy difference ΔE between the two spin states is directly proportional to the magnitude of the applied field [8][9]:

$$\Delta E = \gamma \hbar B_0 \tag{1.1}$$

with B_0 the magnitude of the applied magnetic filed $\vec{\mathbf{B}}_0$, $\hbar$ the Planck constant divided by 2π and γ the so-called gyromagnetic ratio.

However, the transition between the two spin states must be induced by an electromagnetic radiation at Larmor frequency ω_0, which is the precessing angular frequency of nuclei around the applied field:

$$\Delta E = \hbar \omega_0 \tag{1.2}$$

Equating (1.1) and (1.2), we have

$$\omega_0 = \gamma B_0$$

Hence for a given magnetic field strength B_0, nuclei of atoms in different molecular structures, experiencing different chemical environments, can be distinguished by their different Larmor frequencies.

The measurement of an NMR signal requires the perturbation of the system from its equilibrium. Suppose an oscillating magnetic field pulse (short and intense) is applied, with the frequency identical to the Larmor frequency of atoms under study. The nuclei will immediately get excited, absorbing a fraction of the pulse energy. Upon removal of the magnetic field, the system will return to its equilibrium state and the nuclei relax, emitting some energy in a form of a weak electromagnetic radiation which decays with time and is called "*Free Induction Decay*" or *FID*. These electromagnetic waves are detected, picked up, fed to a receiver and stored for further processing and analysis. The obtained measurement data carry highly specific information about the chemical and physical state of the nuclear surroundings in the sample:

- The resonance frequencies of the nuclei enable the distinction of the states of nuclei, since nuclei in different environments give rise to signals at different frequencies.
- The magnitude of the FID at time zero is proportional to the total number of spins, which is a direct measure for the concentration of the molecules under investigation.
- The decay of the measured NMR signal is characterized by the spin-spin relaxation time T_2, which is the time required for the transverse component of the magnetization vector to decrease to a fraction of its original value. This provides more information since different molecules, as well as the same molecules but in different tissues, exhibit different relaxation times due to different macromolecular environments.
- The splitting effect of spectral lines is a result of interaction between the neighboring nuclei. The parameters of the group exhibit special interrelations which reflect the nuclear interaction.

The finally acquired signal, however, will be perturbed by thermal noise from the sample and some noise from the electronic components.

An example of an FID signal, measured from the electromagnetic radiation, is displayed in Fig. 1.1. Clearly, the signal contains contributions from many different metabolites and is perturbed by noise.

Model function [11][12][13]

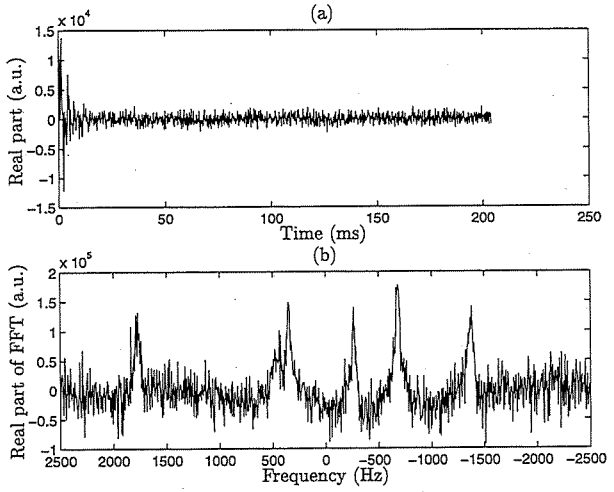


Figure 1.1: An example of an *FID* signal measured from an NMR experiment. (a) Time-domain or measurement-domain signal, (b) Signal spectrum. a.u. means arbitrary unit.

The measured FID signals are collected as complex-valued data since the magnetization is measured in two perpendicular directions. Most of the FID signals are of Lorentzian lineshape (exponential damping) and can be modeled as a sum of exponentially damped complex-valued exponentials as follows

$$x(t_n) = \sum_{k=1}^{K} a_k e^{j\varphi_k} e^{(-d_k + j2\pi f_k)t_n} + e(t_n) \qquad n = 0, \ldots, N-1 \quad (1.3)$$

where $j = \sqrt{-1}$, K is the number of exponentials in the spectrum, also called the model order, and $e(t_n)$ is complex white Gaussian noise. $t_n = t_0 + n\Delta t$ is the time lapse between the effective time origin and sample $x(t_n)$, where Δt is the time interval between observations (or $f_s = 1/\Delta t$ is the sampling frequency) and t_0 the spectrometer dead time. For simplicity, we always assume $t_0 = 0$

in the estimation algorithms, since the estimated parameters can always be extrapolated to the time origin once t_0 is found in practical NMR spectroscopy. The nonlinear parameters f_k and d_k represent the frequency and damping factors while the linear parameters a_k and φ_k are the amplitude and phase of the kth exponential. The physical meaning of these parameters in NMR spectroscopy can be described as follows:

- The frequency of a spectral component is characteristic of the particular metabolite.
- The damping factor of a spectral component corresponds to the inverse of the spin-spin relaxation time T_2 of the metabolite.
- The amplitude is the most important parameter since it is directly proportional to the concentration of the contributing nuclei, or the concentration of the metabolite under investigation.
- $\varphi_k = \varphi^{(0)} + \varphi_k^{(e)}$, in which $\varphi^{(0)}$ is the zero-order phase, whereas $\varphi_k^{(e)}$ represents extra degrees of freedom that may be required under certain experimental conditions (usually all $\varphi_k^{(e)}$ are zero).

To facilitate the analysis using signal processing tools, such as the z-transform, and the formulation and derivation of the estimation algorithms, the above data model (1.3) for $t_n = n\Delta t$ is written as

$$\begin{aligned} x_n &= \sum_{k=1}^{K} a_k e^{j\varphi_k} e^{(-d_k + j2\pi f_k)n\Delta t} + e_n \\ &= \sum_{k=1}^{K} c_k z_k^n + e_n \\ &= \overset{\circ}{x}_n + e_n \qquad\qquad n = 0, \ldots, N-1 \end{aligned} \tag{1.4}$$

where $\overset{\circ}{x}_n, n = 0, \ldots, N-1$ is the signal-only data, $c_k = a_k e^{j\varphi_k}$ is called the kth complex-valued amplitude and $z_k = e^{(-d_k + j2\pi f_k)\Delta t}$ the kth pole of the signal. Throughout this thesis, the signal poles are assumed distinct.

Procedure of parameter estimation

Fig. 1.2 illustrates a typical estimation procedure implemented in this thesis. The given data are time-domain measurements, $x_n, n = 0, 1, \ldots, N-1$, and the results are parameter estimates, $f_k, d_k, a_k, \varphi_k, k = 1, 2, \ldots, K$ defined

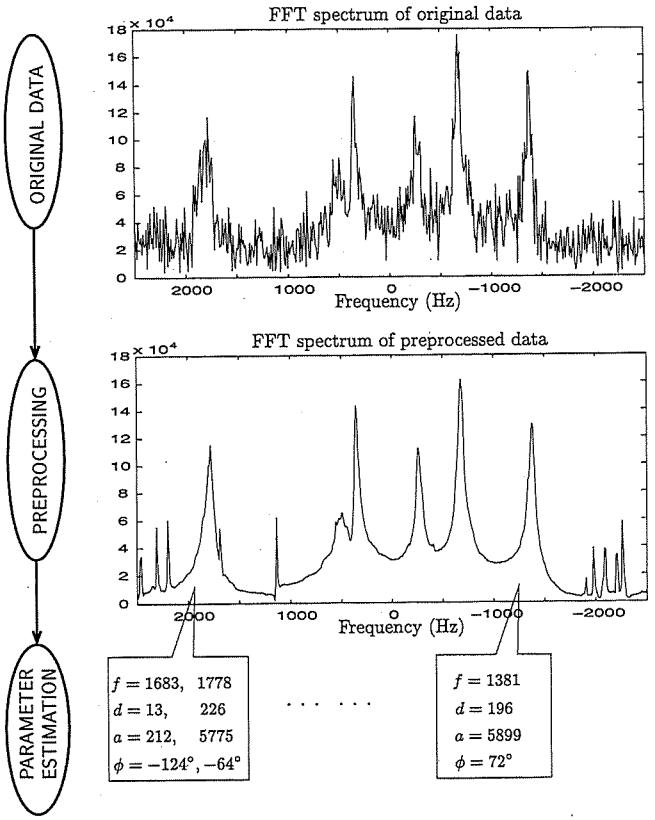


Figure 1.2: Typical parameter estimation procedure.

in (1.4). To visualize signal features, the FFT spectra are plotted, although the data are processed in the time domain in our study. The preprocessed spectrum is the FFT spectrum of the enhanced signal, using the minimum variance enhancement technique discussed in Chapter 4. In fact, the preprocessing step here can be other enhancement techniques or prefiltering techniques developed in this thesis. The estimated parameters are obtained using the HTLS estimation algorithm 1 given in the next Chapter, without incorporating prior knowledge.

1.3 Parameter estimation in other applications

Although our parameter estimation of a sum of exponentials is motivated by and treated in the context of NMR spectroscopy in this thesis, it is generally applicable to a wide variety of problems. For illustrative purposes, we cite the following two applications, where similar parameter estimation problems occur.

Identification of natural modes in transient signals [14][15][5]

Many electromagnetic scatterers and antennas may be modeled as linear time invariant systems represented in terms of their s-plane response. These systems are continuously spatially distributed and thus have an infinite number of modes or resonances. Yet there will be only a finite number of strong modes due to the band-limited nature of physical scatterers and anti-aliasing filters.

In the determination of the natural modes of an electromagnetic scatterer, the measured time series contain desired information, noise, and quite often transient components introduced by the excitation source or measuring equipment. An exponential parametric signal model is often taken for such an observed time series, using a finite and fixed model order:

$$x_n = \sum_{k=1}^{K} a_k e^{s_k n \Delta t} = \sum_{k=1}^{K} a_k z_k^n$$

where x_n, s_k, z_k are complex, and $z_1, z_2, \ldots, z_K$ contain both natural modes of interest and transient components. This is actually the same model function as (1.4), and accurate estimation of the natural modes in this model function is very useful in target identification.

Direction-of-arrival (DOA) estimation [1][2]

In the application of DOA estimation, data from an array of sensors are collected and the objective is to locate point sources assumed to be radiating energy that is detectable by the sensors. Based on a few reasonable assumptions, the radiation impinging on the array is in the form of a sum of plane waves. For simplicity, it is further assumed that the problem is planar and that all sources lie in a half-space, thus reducing the location parameter space to a single-dimensional subset of $\theta_k \in [0, \pi]$, where θ_k is the DOA of the kth source.

For narrowband signals, which are assumed to have the same known center frequency ω_0, the kth signal can be written as the real part of

$$s_k(t) \approx a_k(t) \exp[j(\omega_0 t + \varphi_k(t))] \approx s_k(t) \approx a_k \exp[j(\omega_0 t + \varphi_k)]$$

The approximation here is due to the slow variation of $a_k(t)$ and $\varphi_k(t)$ from narrowband assumption. In this way, the effect of a time delay τ on the received waveforms is simply a phase shift, i.e.,

$$s_k(t - \tau) = s_k(t) \exp(-j\omega_0 \tau)$$

The result is that $x_m(t)$, the complex-valued signal output of the kth sensor at time t, can be written as

$$x_m(t) = \sum_{k=1}^{K} H_m(\theta_k) s_k[t - \tau_m(\theta_k)] = \sum_{k=1}^{K} H_m(\theta_k) \exp[-j\omega_0 \tau_m(\theta_k)] s_k(t)$$

where $\tau_m(\theta_k)$ is the propagation delay between a reference point and the mth wavefront impinging on the array from direction θ_k, the term $H_m(\theta_k)$ is the corresponding sensor complex-valued response (gain and phase) at frequency ω_0, and K point sources are assumed.

Adding measurement noise $e(t)$ and sampling the signals uniformly at $t_n = n\Delta t$, the above model takes basically the same form as (1.3), although the meanings and functions of variables are different. As a matter of fact, the problem of DOA estimation is most frequently referred to in literatures on parameter estimation of a sum of exponentials using subspace-based techniques.

1.4 Thesis outline

In **Chapter 2**, we first give an overview of various parameter estimation techniques. Then, basic principles of SVD and subspace-based parameter estimation techniques, as well as the HTLS estimation algorithm, are reviewed in order to clarify the description of new algorithms based on HTLS in subsequent chapters.

Chapter 3 deals with the problem of frequency-selective estimation. As reviewed in **Chapter 2**, the HTLS estimation method is based on the TLS technique and SVD filtering, which filters out part of the noise by truncating the SVD of the Hankel data matrix to rank K and discarding the non-significant singular values and vectors. When such SVD filtering is not satisfactory at low SNR, or when only some (fewer than K) of the spectral components that reside in a selected frequency region are of interest, we need to further compress the noise and enhance the sinusoids of interest. To emphasize the sinusoids of interest and suppress the unwanted sinusoids and noise, prefiltering methods are presented in **Chapter 3** using FIR and IIR filter matrices prior to an SVD and subspace-based method, such as HTLS. Subsequently, a theoretical analysis of special cases of one and two exponentially damped sinusoids is given, which gives the increment of the SNR obtained from prefiltering and reveals the relationship between the singular values/vectors of the prefiltered and original data matrices. Furthermore, the FIR or IIR prefiltering methods are combined with the HTLS prefiltering technique, which is similar to the HSVD prefiltering technique proposed by Pijnappel et al. [32] for Kung's method. Finally, results of computer simulations are presented to confirm the theoretical predictions.

This chapter is based on [33], [34], [36] and [37].

In **Chapter 4**, we summarize and study a class of SVD-based signal enhancement algorithms, including Cadzow's enhancement [56][57][58] and the minimum variance enhancement [29][61], in the context of NMR spectroscopic signal processing. Based on SVD, these algorithms compute estimates of the signal-only data matrix and then restore the Hankel structure of the computed estimates to clean up noisy data before estimating signal parameters using a subspace-based method. The resolution performance and convergence behavior of these enhancement procedures is studied when followed by an SVD and subspace-based estimator such as HTLS. As demonstrated in this chapter, the minimum variance algorithm is superior to Cadzow's and non-enhanced algorithms in resolution and accuracy performance, as well as computational efficiency.

This chapter is based on [38], [41], [39], [40], and [42].

In **Chapter 5**, we study another class of iterative methods, STLN and STLS, for the problem of approximating Hankel structured rank-deficient data matrices. First, the solution and outline of the STLN and STLS algorithms is given for complex-valued data sets. Then, a new estimator, HTLN, is presented for our problem of exponential data fitting, which starts from the same Hankel data matrix but uses the STLN technique prior to the estimation approach of HTLS. The optimality of the STLN and STLS algorithms, as well as that of the minimum variance and Cadzow's algorithms, are studied in two examples for which the global minimum can be found explicitly. Finally, the enhancement algorithms, called Cadzow's (CA) method, minimum variance (MV), STLS, STLN, and the corresponding enhanced estimators CA-HTLS, MV-HTLS, STLS-HTLS, and HTLN are compared in terms of resolution and accuracy, as well as computational efficiency, based on computer simulations.

This chapter is based on [43], [44] and [45].

The previous chapters deal with different *techniques* of filtering out or cleaning up noise present in the obtained data. In **Chapter 6**, we incorporate *prior knowledge* (PK) to improve our parameter estimates. New methods, such as HTLS(PK), HTLN(PK), etc., that incorporate prior knowledge of some known signal poles into the parameter estimators HTLS and HTLN, etc., are presented. The benefits of the new methods compared to the old methods that allow or do not allow the incorporation of prior knowledge are confirmed through quantification of simulated NMR signals, especially when the known signal poles are close to the unknown ones in the spectrum.

This chapter is based on [46] and [47].

Chapter 7 gives a detailed analysis of the HTLS(PK) method and more quantification results using both simulated and *in vivo* NMR signals. By relating the HTLS(PK) method to the prefiltered HTLS method presented in **Chapter 3**, we provide a means to interpret the incorporation of prior knowledge in terms of a prefiltering operation, which facilitates the intuitive analysis in the frequency domain. The enhancement and efficiency of HTLS(PK), as well as the sensitivity of the HTLS(PK) method with respect to the accuracy of the given poles, are studied. This chapter is based on [48].

Finally, a brief summary of the main features of the new algorithms presented in this thesis, and some suggestions for future research, are given in **Chapter 8**.

To better visualize the structure of this thesis and the application field of the new estimation techniques developed in this thesis, we illustrate in Fig. 1.3 these subspace-based estimation techniques, as well as when and how to use them.

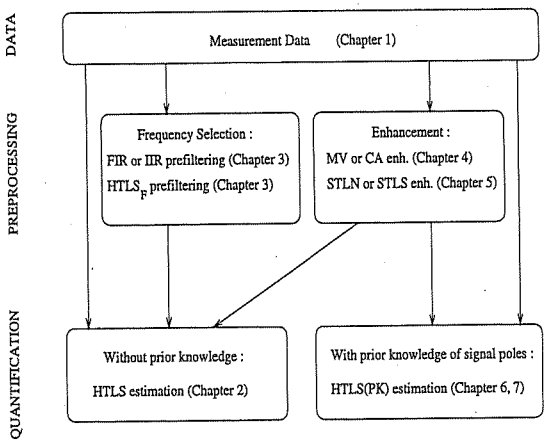


Figure 1.3: Relationships between the new techniques developed in this thesis from an application's point of view.

Chapter 2

Subspace-based Parameter estimation

In this chapter, an overview of various parameter estimation techniques is first given. Then, basic principles of SVD and subspace-based parameter estimation techniques are reviewed, followed by a description of the concise theory of a subspace-based parameter estimation method, called HTLS, with which all the new methods developed in this thesis are compared. Finally, general procedures of Monte-Carlo simulations used for evaluating different estimation methods are described.

2.1 Parameter estimation techniques

In the past years, tremendous effort has been put in developing parameter estimation techniques for the problem modeled in (1.4). This study aims at developing more accurate estimation methods using modern numerical techniques and based on the subspace separation. Before reviewing the existing subspace-based estimation algorithms, which is a noninteractive time-domain method, other methods like spectral estimation in the frequency domain and interactive non-linear data-fitting in the time domain are introduced for comparison.

2.1.1 Time-domain methods versus frequency-domain methods

Traditional methods for resolving closely-spaced sinusoids use the power spectrum or the discrete (fast) Fourier transform (DFT, FFT) of the measurement (time-domain) data sets. It is well known that these discrete transforms to the frequency domain limit the frequency resolution to the reciprocal of the observation interval length. The modern approach to overcoming this fundamental limit and achieving high resolution from a short data segment is by using prior knowledge of the model function formulation and fitting this model in the time domain. The problem of spectral estimation is then reduced to that of estimating the model parameters. In the following, we show more restrictions of frequency-domain methods and advantages of time-domain methods in the context of NMR spectroscopy.

For data modeled in (1.4), the FFT spectrum displays K peaks and the sharpness of these peaks is governed by the damping factor d_k. The larger d_k the wider the peak. As $d_k \to 0$ we approach the line spectrum of a non-damped sum of sinusoids. One of the frequency-domain methods in NMR spectroscopy consists of determining frequencies, amplitudes and damping factors from respectively the spectral peak position, area and width at half height.

Besides the problem of frequency resolution pointed out earlier, other problems arise when applying the FFT to measurement data sets in real-world applications. In NMR spectroscopy, for example, there is frequently an annoying baseline in the FFT spectrum, or a broad hump that can be modeled as a sum of noise peaks with large damping factors. The position of the baseline is critical for a correct quantification by means of a frequency-domain method, such as non-model function fitting (measurement of peak locations, cutting and weighing, triangulation, integration) and model function fitting [13]. In model function fitting, a mathematical lineshape function, for example, a sum of Lorentzian lines as in our model (1.3), is fitted to the spectral data. Due to the mathematically complicated form of the spectral model function, this is a very time-consuming procedure. A major problem arises if the data are incomplete at either end or the data set is truncated so that it does not completely cover the decay of the damping parameters. This leads to leakage of peaks into the sidelobes of the transform such that the FFT estimate of the spectrum deviates from the "true" spectrum. There are various filtering methods to reduce this effect, e.g., multiply the decaying FID by a bell-shaped window function. More methods have been devised to deconvolve artifacts originating from the FFT of data, such as linear prediction(LP)-based deconvolution, the maximum entropy method, etc. (see e.g. [3] and the references therein). However, the highest precision is usually attainable only if the ultimate step of the estimation of parameters is applied to the original data in the measurement domain.

In the last few years, there is an increasing demand to perform model function fitting in the measurement domain, or time domain, using signal processing tools. Generally, time-domain algorithms are less time consuming, in view of the mathematically straight-forward decay functions. But the main advantage of processing the data in the time domain over frequency domain is that absence of data points at either end of the data record need not complicate a quantification. This allows truncation of initial data points without affecting the signal parameters and avoids the requirement of baseline correction and deconvolution in practical applications. In addition, time-domain methods are evolving rapidly and resulting in superior precision by making use of results of numerical analysis and signal processing.

In view of the above features, our data quantification is carried out directly in the time domain in the present dissertation. FFT spectra are used here only for visualizing and interpreting time-domain data, as well as an aid for identifying particular areas of interest.

2.1.2 Noninteractive methods versus interactive methods

Time-domain methods can be divided into two classes: interactive and noninteractive methods.

Interactive methods are not restricted to the model function (1.3). But they require significant involvement of the user, such as choosing the proper model function, providing good starting values, gathering prior knowledge, and judging the convergence to the correct (global) minimum. Clearly, this may be rather expensive in terms of human effort. Note however that the imposition of prior knowledge usually has a beneficial effect on the stability of the convergence of an iterative process and on the resolution and accuracy of the estimated parameters. An example of this class is the VARPRO estimation method [16][17].

Noninteractive methods are always "black box" methods, that is, they require only minimal user involvement. The model function is restricted to (1.3) and minimal expert knowledge is required. Although this model may be inadequate for signals containing non-exponential components, in which case the quantification is mathematical rather than physical, the fit can most often be made quite well. This is achieved by decomposing each deviating non-exponential spectral component into a sum of exponentially damped sinusoids [18]. A price to pay for the relative ease of operation is that the computed parameter estimates are less accurate than those obtained with more elaborate

interactive methods, especially at low SNR's, since little prior knowledge can be included. Examples of noninteractive methods are SVD and subspace-based methods, and Evans-Fischl-based methods [19]. In this study we seek to improve the resolution and accuracy of subspace-based estimation algorithms of this class.

2.2 Subspace-based parameter estimation

By subspace-based parameter estimation, we mean that the noisy signal space is subdivided into a "signal" and a "noise" subspace using SVD or other numerical techniques. There are various subspace-based estimation methods, depending on how these subspaces are determined, and how the pole information is extracted from the subspace and its assumed properties. Some well-known estimation methods of this type are the linear prediction-based methods [20][21], the matrix pencil (MP) method [22] and Kung et al.'s state-space-based method [23] (which is called HSVD in the NMR literature [4]). All these methods start from a Hankel/Toeplitz data matrix and find the "signal" subspace by truncating the SVD of this matrix to rank K, the number of exponentials in the underlying signal. The accuracy of these methods is improved by making use of the total least squares (TLS) technique [24][6]. The forward and backward LP methods, based on TLS and called TLS-FLP and TLS-BLP, have been proposed in [25], while [26][27][28] describe how to use the TLS principle in Kung et al.'s state-space-based method. The latter method is called the Hankel TLS (HTLS) method in NMR spectroscopy.

Since HTLS manifests the best performance, including resolution and parameter accuracy, among these techniques [28], this study starts with the HTLS estimation method and tries to improve the resolution and accuracy of the HTLS parameter estimates via prefiltering, signal enhancement, incorporation of prior knowledge, etc. For clarity of exposition of the HTLS estimation method, as well as other subspace-based estimation techniques presented later, we reformulate the basic principles and the relationships between the SVD and signal subspace estimation [29][61][62], before introducing the HTLS estimation algorithm..

2.2.1 SVD and signal subspace estimation

This subsection is mainly based on De Moor's [29].

Assume that an $L \times M$ Hankel matrix

$$\mathbf{X} = \begin{bmatrix} x_0 & x_1 & \cdots & x_{M-1} \\ x_1 & x_2 & \cdots & \cdot \\ \vdots & \vdots & \vdots & \vdots \\ x_{L-1} & \cdot & \cdots & x_{N-1} \end{bmatrix} \quad L \geq M \geq K, N = L + M - 1 \qquad (2.1)$$

consisting of N uniformly sampled data points $x_n, n = 0, 1, \ldots, N-1$, and a model-order estimate K are given.

With the basic assumption of additive noise in (1.4), the data matrix $\mathbf{X}$ can be decomposed into two unknown matrices: an exact data matrix $\overset{\circ}{\mathbf{X}}$ and an additive perturbation matrix $\mathbf{E}$, so that

$$\mathbf{X} = \overset{\circ}{\mathbf{X}} + \mathbf{E}$$

where $\mathbf{X}, \overset{\circ}{\mathbf{X}}, \mathbf{E} \in \mathbb{C}^{L \times M}$, and typically, $\text{rank}(\mathbf{X}) = \text{rank}(\mathbf{E}) = \min\{L, M\}$, while $\text{rank}(\overset{\circ}{\mathbf{X}}) = K \leq \min\{L, M\}$. Computing the SVD of the exact rank-K matrix $\overset{\circ}{\mathbf{X}}$ yields:

$$\overset{\circ}{\mathbf{X}} = \overset{\circ}{\mathbf{U}} \overset{\circ}{\mathbf{\Sigma}} \overset{\circ}{\mathbf{V}}^H = [\overset{\circ}{\mathbf{U}}_1 \; \overset{\circ}{\mathbf{U}}_2] \begin{bmatrix} \overset{\circ}{\mathbf{\Sigma}}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} [\overset{\circ}{\mathbf{V}}_1 \; \overset{\circ}{\mathbf{V}}_2]^H \qquad (2.2)$$

where ^{H} denotes Hermitian conjugation or conjugate transposition, $\overset{\circ}{\mathbf{U}}_1 \in \mathbb{C}^{L \times K}$, $\overset{\circ}{\mathbf{U}}_2 \in \mathbb{C}^{L \times (L-K)}$, $\overset{\circ}{\mathbf{\Sigma}}_1 \in \mathbb{C}^{K \times K}$, $\overset{\circ}{\mathbf{V}}_1 \in \mathbb{C}^{M \times K}$, $\overset{\circ}{\mathbf{V}}_2 \in \mathbb{C}^{M \times (M-K)}$. In this way, the conjugated row space of $\overset{\circ}{\mathbf{X}}$ coincides with the column space of $\overset{\circ}{\mathbf{V}}_1$ [1], and the columns of $\overset{\circ}{\mathbf{V}}_2$ generate an orthogonal basis for the null space of $\overset{\circ}{\mathbf{X}}$ [2]:

$$\mathcal{R}(\overset{\circ}{\mathbf{X}}^H) = \mathcal{R}(\overset{\circ}{\mathbf{V}}_1), \quad \mathcal{N}(\overset{\circ}{\mathbf{X}}) = \mathcal{R}(\overset{\circ}{\mathbf{V}}_2)$$

Let $\check{\mathbf{U}}_1 \check{\mathbf{\Sigma}}_1 \check{\mathbf{V}}_1^H$ be the SVD of $(\overset{\circ}{\mathbf{X}} \overset{\circ}{\mathbf{V}}_1 + \mathbf{E} \; \overset{\circ}{\mathbf{V}}_1)$, and $\check{\mathbf{U}}_2 \check{\mathbf{\Sigma}}_2 \check{\mathbf{V}}_2^H$ be the SVD of $(\mathbf{E} \; \overset{\circ}{\mathbf{V}}_2)$. Then, under the following three conditions:

C1. The exact data matrix $\overset{\circ}{\mathbf{X}}$ should be orthogonal to the noise matrix $\mathbf{E}$ in the sense that $\overset{\circ}{\mathbf{X}}^H \mathbf{E} = \mathbf{0}$;

[1] $\mathcal{R}(\cdot)$ denotes the range of the matrix in brackets
[2] $\mathcal{N}(\cdot)$ denotes the null space of the matrix in brackets

C2. The matrices $\overset{\circ}{\mathbf{V}}_1$ and $\overset{\circ}{\mathbf{V}}_2$ must be orthogonal to each other in the inner product generated by the matrix $\mathbf{E}^H\mathbf{E}$: $\overset{\circ}{\mathbf{V}}_1^H \mathbf{E}^H\mathbf{E}\, \overset{\circ}{\mathbf{V}}_2 = \mathbf{0}$;

C3. The smallest singular value of $\breve{\mathbf{\Sigma}}_1$ must be larger than the largest singular value of $\breve{\mathbf{\Sigma}}_2$;

the rank K of $\overset{\circ}{\mathbf{X}}$ and the subspaces generated by $\overset{\circ}{\mathbf{V}}_1$ and $\overset{\circ}{\mathbf{V}}_2$ (the row and null space of $\overset{\circ}{\mathbf{X}}$) can be estimated consistently, while the original exact column space of $\overset{\circ}{\mathbf{X}}$, represented by $\overset{\circ}{\mathbf{U}}_1$, cannot be recovered from the SVD of $\mathbf{X}$, as shown below.

$$\begin{aligned}
\mathbf{X} &= \overset{\circ}{\mathbf{X}} + \mathbf{E} \\
&= (\overset{\circ}{\mathbf{X}}\overset{\circ}{\mathbf{V}}_1 + \mathbf{E}\overset{\circ}{\mathbf{V}}_1)\overset{\circ}{\mathbf{V}}_1^H + (\mathbf{E}\overset{\circ}{\mathbf{V}}_2)\overset{\circ}{\mathbf{V}}_2^H \\
&= (\breve{\mathbf{U}}_1\breve{\mathbf{\Sigma}}_1\breve{\mathbf{V}}_1^H)\overset{\circ}{\mathbf{V}}_1^H + (\breve{\mathbf{U}}_2\breve{\mathbf{\Sigma}}_2\breve{\mathbf{V}}_2^H)\overset{\circ}{\mathbf{V}}_2^H \\
&= [\breve{\mathbf{U}}_1 \;\; \breve{\mathbf{U}}_2]\begin{bmatrix} \breve{\mathbf{\Sigma}}_1 & \mathbf{0} \\ \mathbf{0} & \breve{\mathbf{\Sigma}}_2 \end{bmatrix}\begin{bmatrix} \breve{\mathbf{V}}_1^H\overset{\circ}{\mathbf{V}}_1^H \\ \breve{\mathbf{V}}_2^H\overset{\circ}{\mathbf{V}}_2^H \end{bmatrix} \qquad (2.3)
\end{aligned}$$

Suppose that the above three assumptions are all satisfied. Since $\mathcal{R}(\breve{\mathbf{U}}_1) = \mathcal{R}(\overset{\circ}{\mathbf{X}}\overset{\circ}{\mathbf{V}}_1 + \mathbf{E}\overset{\circ}{\mathbf{V}}_1)$ and $\mathcal{R}(\breve{\mathbf{U}}_2) = \mathcal{R}(\mathbf{E}\overset{\circ}{\mathbf{V}}_2)$, the assumptions C1 and C2 guarantee that $\overset{\circ}{\mathbf{V}}_1^H(\overset{\circ}{\mathbf{X}}^H + \mathbf{E}^H)\mathbf{E}\overset{\circ}{\mathbf{V}}_2 = \mathbf{0}$, i.e., $\breve{\mathbf{U}}_1^H\breve{\mathbf{U}}_2 = \mathbf{0}$. Considering the assumption C3, it is easy to see that Eq. (2.3) is an SVD of $\mathbf{X}$.

Further assume that $\mathbf{E}^H\mathbf{E}$ is a scalar multiple of the identity matrix

$$\mathbf{E}^H\mathbf{E} = \mu^2\mathbf{I}_M = L\nu^2\mathbf{I}_M \qquad (2.4)$$

where ν^2 is the variance of the independently and identically distributed zero-mean noise element in $\mathbf{E}$ asymptotically, as the number of measurements and hence $L \to \infty$. This is a special case of assumption C2. In this case, the SVD of $\mathbf{X}$ can be explicitly expressed in terms of the SVD of $\overset{\circ}{\mathbf{X}}$

$$\begin{aligned}
\mathbf{X} &= \overset{\circ}{\mathbf{X}} + \mathbf{E} \\
&= \overset{\circ}{\mathbf{U}}_1\overset{\circ}{\mathbf{\Sigma}}_1\overset{\circ}{\mathbf{V}}_1^H + \mathbf{E}\overset{\circ}{\mathbf{V}}_1\overset{\circ}{\mathbf{V}}_1^H + \mathbf{E}\overset{\circ}{\mathbf{V}}_2\overset{\circ}{\mathbf{V}}_2^H \\
&= [(\overset{\circ}{\mathbf{U}}_1\overset{\circ}{\mathbf{\Sigma}}_1 + \mathbf{E}\overset{\circ}{\mathbf{V}}_1)(\overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K)^{-\frac{1}{2}} \;\; \mathbf{E}\overset{\circ}{\mathbf{V}}_2\,\mu^{-1}] \\
&\quad \begin{bmatrix} \overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K & \mathbf{0} \\ \mathbf{0} & \mu\mathbf{I}_{M-K} \end{bmatrix}^{\frac{1}{2}}\begin{bmatrix} \overset{\circ}{\mathbf{V}}_1^H \\ \overset{\circ}{\mathbf{V}}_2^H \end{bmatrix}
\end{aligned}$$

$$= [\mathbf{U}_1 \; \mathbf{U}_2]\begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 \end{bmatrix}\begin{bmatrix} \mathbf{V}_1^H \\ \mathbf{V}_2^H \end{bmatrix} \quad \text{for } L \geq M \qquad (2.5)$$

Upon examination of these relationships, it can be concluded that under the assumptions we have made,

- The exact singular values of $\overset{\circ}{\mathbf{X}}$ can be calculated from
$$\overset{\circ}{\mathbf{\Sigma}}_1 = \sqrt{\mathbf{\Sigma}_1^2 - \mu^2\mathbf{I}_K}$$
- The row space of $\overset{\circ}{\mathbf{X}}$ can be recovered exactly, i.e., $\overset{\circ}{\mathbf{V}}_1 = \mathbf{V}_1$, $\overset{\circ}{\mathbf{V}}_2 = \mathbf{V}_2$
- The column space of $\overset{\circ}{\mathbf{X}}$ can not be recovered, but the canonical angles between the column space of $\overset{\circ}{\mathbf{X}}$, represented by $\mathcal{R}(\overset{\circ}{\mathbf{U}}_1)$, and the column space $\mathcal{R}(\mathbf{U}_1)$ generated by the first K left singular vectors of $\mathbf{X}$, can be computed. The cosine of these canonical angles are the diagonal elements of the diagonal matrix, denoted by $\mathbf{C}_c$ as follows:
$$\begin{aligned}
\mathbf{C}_c &= \overset{\circ}{\mathbf{U}}_1^H\mathbf{U}_1 \\
&= \overset{\circ}{\mathbf{U}}_1^H(\overset{\circ}{\mathbf{U}}_1\overset{\circ}{\mathbf{\Sigma}}_1 + \mathbf{E}\overset{\circ}{\mathbf{V}}_1^H)(\overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K)^{-\frac{1}{2}} \\
&= \overset{\circ}{\mathbf{\Sigma}}_1(\overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K)^{-\frac{1}{2}} \\
&= \overset{\circ}{\mathbf{\Sigma}}_1\mathbf{\Sigma}_1^{-1} \qquad (2.6)
\end{aligned}$$
which characterizes the bias of the signal subspace estimate $\mathbf{X}$ and provides an insight as to how well the matrix $\mathbf{X}$ approximates the exact matrix $\overset{\circ}{\mathbf{X}}$.

Despite the fact that $\overset{\circ}{\mathbf{U}}_1$ cannot be consistently estimated from the SVD of $\mathbf{X}$, it is possible to find the estimate of $\overset{\circ}{\mathbf{X}}$, for example, in the least squares (LS) sense, from the SVD of $\mathbf{X}$ if all the three assumptions are satisfied and $\mathbf{E}^H\mathbf{E} = \mu^2\mathbf{I}$.

LS estimation of $\overset{\circ}{\mathbf{X}}$

Approximate $\mathbf{X}$ by a matrix $\mathbf{X}_{\mathrm{LS}}$ of rank K in the least squares sense:

$$\min_{\mathbf{X}_{\mathrm{LS}} \in \mathbb{C}^{L\times M}, \mathrm{rank}(\mathbf{X}_{\mathrm{LS}}) = K \leq \min\{L,M\}} \|\mathbf{X} - \mathbf{X}_{\mathrm{LS}}\|_F^2$$

Its classical solution is

$$\mathbf{X}_{\mathrm{LS}} = \mathbf{U}_1(\overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K)^{\frac{1}{2}}\overset{\circ}{\mathbf{V}}_1^H = \mathbf{U}_1\mathbf{\Sigma}_1\mathbf{V}_1^H$$

which is simply a truncated SVD of $\mathbf{X}$ to rank K.

In the following presentation, we frequently use $\mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H$ to denote such a rank-K truncation, where $\mathbf{U}_K = \mathbf{U}_1 \in \mathbb{C}^{L\times K}, \mathbf{V}_K = \mathbf{V}_1 \in \mathbb{C}^{M\times K}$ are truncated matrices containing the first K singular vectors of $\mathbf{U} \in \mathbb{C}^{L\times L}, \mathbf{V} \in \mathbb{C}^{M\times M}$, and $\mathbf{\Sigma}_K = \mathbf{\Sigma}_1 = \mathrm{diag}(\sigma_1, \cdots, \sigma_K) \in \mathbb{C}^{K\times K}$ is the truncated matrix of $\mathbf{\Sigma} \in \mathbb{C}^{L\times M}$. Also, $\mathbf{U}_2$ and $\mathbf{V}_2$, the bases of orthogonal complements of $\mathcal{R}(\mathbf{U}_1)$ and $\mathcal{R}(\mathbf{V}_1)$, are represented by $\overline{\mathbf{U}}_K$ and $\overline{\mathbf{V}}_K$.

SVD truncation can be considered as filtering out part of the noise by discarding the nonprincipal singular values/vectors and finding the signal subspace basis $\mathbf{U}_K$ or $\mathbf{V}_K$. The success of SVD-based subspace methods critically depends on the assumptions C1, C2, C3, which in practice however are never satisfied exactly (except maybe for condition C3). A nice feature of the SVD however is its robustness with respect to (mild) violations of these conditions. If for instance $\|\overset{\circ}{\mathbf{X}}^H\mathbf{E}\|$ is small (where $\|.\|$ is any unitarily invariant norm), the SVD of $\mathbf{X}$ will still deliver good approximations. The smaller $\|\overset{\circ}{\mathbf{X}}^H\mathbf{E}\|$ gets, the better the approximations will be.

2.2.2 HTLS: a subspace-based parameter estimation algorithm

If the data are noise-free, then $\mathbf{X} = \overset{\circ}{\mathbf{X}} \in \mathbb{C}^{L\times M}$ in (2.1) has a Vandermonde decomposition of the following form:

$$\begin{aligned}\mathbf{X} = \overset{\circ}{\mathbf{X}} &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1^1 & z_2^1 & \cdots & z_K^1 \\ \vdots & & & \vdots \\ z_1^{L-1} & z_2^{L-1} & \cdots & z_K^{L-1} \end{bmatrix} \begin{bmatrix} c_1 & & & 0 \\ & c_2 & & \\ & & \ddots & \\ 0 & & & c_K \end{bmatrix} \\ &\quad \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1^1 & z_2^1 & \cdots & z_K^1 \\ \vdots & & & \vdots \\ z_1^{M-1} & z_2^{M-1} & \cdots & z_K^{M-1} \end{bmatrix}^T \\ &= \mathbf{S}_K\mathbf{C}_K\mathbf{T}_K^T \end{aligned} \tag{2.7}$$

where ^{T} denotes matrix transposition, $\mathbf{S}_K \in \mathbb{C}^{L\times K}$ and $\mathbf{T}_K \in \mathbb{C}^{M\times K}$ are Vandermonde matrices composed of the K signal poles, and $\mathbf{C}_K \in \mathbb{C}^{K\times K}$ is a diagonal matrix composed of K complex-valued amplitudes. One observation from the above Vandermonde decomposition is that the rank of such a signal-only Hankel matrix $\mathbf{X}$ is K according to the rank inequality [30] below

$$\begin{aligned} \mathrm{rank}(\mathbf{S}_K) &= \mathrm{rank}(\mathbf{C}_K) = \mathrm{rank}(\mathbf{T}_K) = K \\ \mathrm{rank}(\mathbf{S}_K\mathbf{C}_K\mathbf{T}_K^T) &\geq \mathrm{rank}(\mathbf{S}_K) + \mathrm{rank}(\mathbf{C}_K) + \mathrm{rank}(\mathbf{T}_K^T) - 2K \\ \mathrm{rank}(\mathbf{S}_K\mathbf{C}_K\mathbf{T}_K^T) &\leq \min\{\mathrm{rank}(\mathbf{S}_K), \mathrm{rank}(\mathbf{C}_K), \mathrm{rank}(\mathbf{T}_K^T)\} \end{aligned}$$

The rank-K property of $\mathbf{X}$ results in

$$\mathbf{X} \overset{\mathrm{SVD}}{=} \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H \overset{\mathrm{truncate}}{=} \mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H \tag{2.8}$$

Another observation is that $\mathbf{S}_K, \mathbf{T}_K$ have the shift-invariant property, i.e.,

$$\mathbf{S}_K^{\uparrow} = \mathbf{S}_{K\downarrow}\mathbf{Z}_K, \quad \mathbf{T}_K^{\uparrow} = \mathbf{T}_{K\downarrow}\mathbf{Z}_K, \quad \mathbf{Z}_K = \mathrm{diag}(z_1, z_2, \ldots, z_K)$$

where the up (down) arrow stands for deleting the top (bottom) row, and $\mathbf{Z}_K \in \mathbb{C}^{K\times K}$ is a diagonal matrix composed of the K signal poles, $z_k, k = 1, \ldots, K$.

Equating the two decompositions (2.8) and (2.7), we have the shift-invariant property of $\mathbf{U}_K$ and $\mathbf{V}_K^*$ also:

$$\mathbf{U}_K^{\uparrow} = \mathbf{U}_{K\downarrow}\mathbf{Z}_K^{(U)}, \quad \mathbf{V}_K^{*\uparrow} = \mathbf{V}_{K\downarrow}^*\mathbf{Z}_K^{(V)}$$

where * denotes conjugation, $\mathbf{Z}_K^{(U)} \in \mathbb{C}^{K\times K}$ and $\mathbf{Z}_K^{(V)} \in \mathbb{C}^{K\times K}$ are similarity transforms of $\mathbf{Z}_K$.

For noisy data, the equation of SVD truncation in (2.8) does not hold and there is no exact solution $\mathbf{Z}_K^{(U)}, \mathbf{Z}_K^{(V)}$ of the shift-invariant property in $\mathbf{U}_K, \mathbf{V}_K$ anymore. However, based on the LS estimation of $\overset{\circ}{\mathbf{X}}$ derived in the previous subsection, we can approximate $\mathbf{X}$ by the truncated SVD of

$$\mathbf{X} \overset{\mathrm{SVD}}{=} \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H \overset{\mathrm{truncate}}{\approx} \mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H$$

and compute the TLS solution $\mathbf{Z}_K^{(U)}$ or $\mathbf{Z}_K^{(V)}$ of the set of equations

$$\mathbf{U}_K^{\uparrow} \approx \mathbf{U}_{K\downarrow}\mathbf{Z}_K^{(U)}, \quad or \quad \mathbf{V}_K^{*\uparrow} \approx \mathbf{V}_{K\downarrow}^*\mathbf{Z}_K^{(V)}$$

Then, the estimates of signal-poles $z_k, k = 1, 2, \ldots, K$ can be computed as the eigenvalues of $\mathbf{Z}_K^{(U)}$ or $\mathbf{Z}_K^{(V)}$.

The linear parameters are found by filling in the obtained poles in the model function (1.4) and fitting these equations to the original data with the LS method, as described in [23][27] and shown in the following outline of the HTLS algorithm.

Algorithm 1

Outline of the HTLS algorithm

Given. A model-order estimate K and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel matrix $\mathbf{X}$ as in (2.1).

Step 1: SVD. Compute a truncated SVD of $\mathbf{X} \approx \mathbf{U}_K \mathbf{\Sigma}_K \mathbf{V}_K^H$, where $\mathbf{U}_K$ ($\mathbf{V}_K$) are the first K left (right) singular vectors, and $\mathbf{\Sigma}_K$ is the diagonal matrix composed of the first K singular values.

Step 2: TLS. Compute the TLS solution $\mathbf{Z}_K^{(U)}$ of $\mathbf{U}_K^{\uparrow} \approx \mathbf{U}_{K\downarrow} \mathbf{Z}_K^{(U)}$, extract the signal poles $z_k, k = 1, 2, \ldots, K$ from the eigenvalues of $\mathbf{Z}_K^{(U)}$, and find the nonlinear parameters f_k, d_k, $k = 1, \cdots, K$ using the relation $z_k = e^{(-d_k + j2\pi f_k)/f_s}$.

Step 3: (LS). Compute the LS solution $\mathbf{c} = [c_1, c_2, ..., c_K]^T$ of the set

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1^1 & z_2^1 & \cdots & z_K^1 \\ \vdots & & & \vdots \\ z_1^{N-1} & z_2^{N-1} & \cdots & z_K^{N-1} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_K \end{bmatrix} \approx \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix}$$

based on the SVD of the Vandermonde matrix at the left-hand-side, and then estimate the linear parameters a_k and φ_k from $c_k = a_k e^{j\varphi_k}, k = 1, 2, \ldots, K$.

The pole-estimation in step 2 can also be performed in the row space just by solving $\mathbf{V}_K^{*\uparrow} \approx \mathbf{V}_{K\downarrow}^{*} \mathbf{Z}_K^{(V)}$ instead of $\mathbf{U}_K^{\uparrow} \approx \mathbf{U}_{K\downarrow} \mathbf{Z}_K^{(U)}$ and retrieving the eigenvalues of $\mathbf{Z}_K^{(V)}$, instead of $\mathbf{Z}_K^{(U)}$.

It is to be noted that the HTLS algorithm performs well on $\mathbf{X} = \overset{\circ}{\mathbf{X}} + \mathbf{E}$ when $\overset{\circ}{\mathbf{X}}^H \mathbf{E} = \mathbf{0}$ is satisfied asymptotically, and $\mathbf{E}^H \mathbf{E}$ is a scalar multiple of identity matrix, or, the noise is asymptotically white. This is due to the assumptions C1, C2 and (2.4) in the previous subsection.

For differently sized data matrices $\mathbf{X}_{L \times M}$, with $L + M - 1 = N$, the HTLS algorithm gives rather accurate estimates at around $1/3 \leq M/N \leq 2/3$ [26]. But the best estimates are obtained at $M/N \approx 0.5$. Hence, $\mathbf{X}$ is suggested to be as square as possible, i.e., $L = M(+1)$, in the following study.

2.3 Monte-Carlo simulation procedure

In the next chapters, the performance of different estimation algorithms are compared and evaluated via extensive Monte-Carlo experiments on simulated signals. Here we describe the simulation procedure in general.

The objective is to resolve the sinusoids of interest in simulated signals and estimate their signal parameters using a limited number of noisy observations by means of different methods. Various scenarios are simulated in which SNR, number of data points, estimated order, matrix size, and other parameters are varied over a specified range. All computations are performed using Matlab [31].

The simulated signals studied in this thesis are described in Appendix A. Their FFT spectra are depicted in figures. Following NMR spectroscopy convention, these spectra are reversed, i.e., from positive frequency to negative frequency, and expressed mostly using the real part of the FFT, which gives sharper peaks and clearer view of phases. Each of these signals is represented by N data points, uniformly sampled at f_s Hz and modeled by the Kth order exponential function perturbed by white Gaussian noise given by (1.4). The corresponding signal parameters f_k, d_k, a_k and $\phi_k = \varphi_k * 180/\pi$ (i.e., ϕ_k expresses the phase φ_k in degrees) are listed in tables. The real and imaginary components of the Gaussian noise have standard deviation σ_ν.

In our Monte Carlo simulations, we compare the performance of different methods in terms of resolution and parameter accuracy. Unless specified otherwise, 200 noise trials are realized for each considered noise standard deviations σ_ν. For each case, the same (noisy) data are used for all the methods. The results are shown in tables and figures. Typically, when σ_ν are varied, the failure rates are plotted versus σ_ν, while the (relative) root mean-squared error (RMSE) values of the estimated parameters, together with the corresponding Cramér-Rao (CR) lower bounds, are plotted versus the peak SNR. Alternatively, these results, as well as the bias and standard deviation of the estimated parameters, may be listed in tables in some cases. The computation of the above notations, such as peak SNR, CR bound, failure, RMSE, bias and standard deviation, are explained in the following.

Peak SNR:

The SNR of peak k is expressed in decibels and given by $10\log_{10}(a_k^2/(2\sigma_\nu^2))$.

CR bound:

The CR lower bounds [22] on the statistical errors in the parameter estimates are used for accuracy evaluation. The values of CR bounds are derived from σ_ν, the number of data points used and the exact parameter values of the complex exponentially damped model specified in (1.4).

Failure or bad run:

A *failure*, or a bad run, occurs if not all peaks are resolved within specified intervals lying symmetrically around the exact frequencies. The defined intervals for the series of simulated signals used in this thesis are also listed in Appendix A. They are chosen based on the CR bounds at which the frequency intervals of CR bounds for the overlapping peaks touch one another. Take *Signal-1* for example. The half-width of the frequency intervals in which each peak should be resolved are respectively 0.0094 Hz and 0.0106 Hz. In fact, these numbers are the CR frequency bounds for each peak at the noise standard deviation $\sigma_\nu = 0.2$. As such, the sum of the contiguous half intervals 0.0094 Hz+0.0106 Hz= 0.02 Hz is exactly the same as the frequency difference between the two sinusoids. The failure rate is computed to indicate the percentage of failures or bad runs that an algorithm fails to resolve all peaks within these specified intervals.

RMSE, bias and standard deviation:

The values of (relative) RMSE, bias and standard deviation of the signal parameter estimates f_k, d_k, a_k and ϕ_k are computed using the *good* runs only (excluding failures). To be more specific, RMSE values are calculated for f_k and ϕ_k, and relative RMSE values with respect to exact parameter values are calculated for d_k and a_k. Although the exclusion of failures is not statistically sound, the same procedure was applied in [26][27] and is mainly justified from the NMR application as follows. In NMR spectroscopic data measurements, each peak of interest characterizes a certain metabolite and is expected to appear around an approximately known frequency. This information allows us to decide whether a run is successful or not. The corresponding parameter estimates only contain useful metabolite information when a peak estimate is found within the expected frequency interval. Therefore, failures should be excluded in our RMSE computation since it makes no sense to compare the parameters of a peak of interest with those of a noise peak that might appear far away from the expected frequency region. In addition, provided the failure rate of the methods under study are comparable, this procedure allows us to point out which method provides the best parameter estimates in case the peaks can be resolved.

Chapter 3

Prefiltering techniques

A typical SVD and subspace-based estimation method yields automatically all the $4K$ parameters when fitting a noisy data sets composed of K exponentially damped sinusoids modeled by (1.4). It can filter out part of the noise by truncating the SVD of the Hankel data matrix to rank K and discarding the non-significant singular values and vectors, as done by the HTLS Algorithm 1 introduced in the previous chapter. However, such SVD filtering is not always satisfactory especially at low SNR. Moreover, in some applications such as NMR spectroscopy [35][49], it frequently occurs that only some (fewer than K) of the spectral components that reside in a selected frequency region are of interest. To further suppress the noise and/or enhance sinusoids of interest, it is often desired to prefilter the input data. In this chapter, we propose prefiltering methods which can be applied for two purposes:

1. When all the sinusoids are of interest and the SNR's are low, a bandpass filter can be designed to encompass all the sinusoids and to reduce the noise outside the passband.

2. When only some of the sinusoids are of interest with known frequencies but unknown damping factors and amplitudes as typically occurs in NMR spectroscopy, a prefiltering technique can be designed to select only the wanted peaks and suppress the unwanted ones and the noise.

An LP-ZOOM technique has been proposed in [50] based on linear prediction. By selectively fitting the areas containing peaks of interest rather than fitting the whole spectrum, LP-ZOOM reduces the prediction filter length as well as the computation time, and produces more accurate parameter estimates

compared to the ordinary LP method. However, the improvement in accuracy is limited by the linear prediction principle.

Another prefiltering technique has been proposed in [51]. This method applies bandpass finite impulse response (FIR) prefiltering to the data prior to applying the MP estimation algorithm. The essence of this method is to multiply the original Hankel data matrix with a square, full-rank FIR filter matrix. However, the FIR filter matrix used in [51], referred to as $\mathbf{H}_{\mathrm{FIR}c}$ in this chapter, is circularly constructed from the filter impulse response without zero-padding. It implements discrete circular convolution without zero-padding, and therefore introduces an end-effect distortion [52]. Moreover, the restriction in $\mathbf{H}_{\mathrm{FIR}c}$ that the length of an FIR filter be equal to that of the input sometimes imposes a difficulty for the filter design.

In the following sections of this chapter, we present several new prefiltering techniques. First, we illustrate in Section 3.1 why our prefiltering using an FIR or IIR filter is performed via the multiplication of the Hankel data matrix with a filter matrix. Then, in Section 3.2, an improved filter matrix that implements the FIR prefiltering by means of discrete convolution with zero-padding, as well as other filter matrices for the backward and forward infinite impulse response (IIR) prefiltering are constructed. These matrices are applied prior to the HTLS method, referred to as prefiltered HTLS in Section 3.3. Subsequently, a theoretical analysis on a special case of two exponentially damped sinusoids is given in Section 3.4, revealing the relationship between the singular values/vectors of the prefiltered and original data matrices. Furthermore, the HTLS$_F$ prefiltering technique is developed in Section 3.5, where prefiltering is accomplished based on a pre-estimation of the signal parameters using HTLS, instead of an FIR or IIR filter. Finally, comparison and combination of different prefiltering techniques are made and the benefits of these new techniques are confirmed through simulations in Section 3.6.

Note that the proposed prefiltering technique can be used for other subspace and SVD-based methods as well, such as the LP method, the MP method and Kung et al.'s method.

3.1 Hankel matrix and filtering

The SVD filtering, which is crucial to the subspace-oriented estimation algorithms, is based on the rank-K property of Hankel/Toeplitz data matrices in the absence of noise. However, a Hankel/Toeplitz matrix constructed from the filtered data, instead of the original noiseless data, does not possess the rank-K property anymore. Such changes in the rank of Hankel matrices, as is the case

for HTLS, are elucidated below from the signal poles' point of view.

In the absence of noise, the signal modeled by (1.4) as a sum of K exponentials is in fact an autoregressive and moving average (ARMA) signal with a rational z-transform:

$$\mathrm{X}(\mathrm{z}) = \sum_{k=1}^{K} \frac{c_k}{1 - z_k \mathrm{z}^{-1}} = \frac{\sum_{i=0}^{K-1} b_i \mathrm{z}^{-i}}{1 + \sum_{k=1}^{K} a_k \mathrm{z}^{-k}}$$

Consider a general ARMA model of

$$\breve{\mathrm{X}}(\mathrm{z}) = \frac{\sum_{i=0}^{r_b} b_i \mathrm{z}^{-i}}{\sum_{k=0}^{r_a} a_k \mathrm{z}^{-k}}, \quad a_0 \neq 0$$

In the time domain, its impulse response $\breve{x}_n$ can be calculated from

$$\breve{x}_n = \frac{1}{a_0}\Big(\sum_{i=0}^{r_b} b_i \delta_{n-i} - \sum_{k=1}^{r_a} a_k \breve{x}_{n-k}\Big)$$

using the initial conditions $\breve{x}_{-1} = \breve{x}_{-2} = \cdots = \breve{x}_{-r_a} = 0$. The forward recursion of this model can be written in matrix notation

$$\begin{bmatrix} & & & & \breve{x}_0 \\ & 0 & & \breve{x}_0 & \breve{x}_1 \\ & & . & \breve{x}_1 & \breve{x}_2 \\ & . & & & \vdots \\ \boxed{\begin{matrix} \breve{x}_0 & \breve{x}_1 & . & . & \breve{x}_{r_a} \\ \breve{x}_1 & \breve{x}_2 & . & . & \breve{x}_{r_a+1} \\ \vdots & \vdots & & & \vdots \\ \vdots & \vdots & & & \vdots \end{matrix}} & & & & \end{bmatrix} \begin{bmatrix} a_{r_a} \\ a_{r_a-1} \\ \vdots \\ \vdots \\ a_0 \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{r_b} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Note that the matrix in the box is a Hankel matrix with $r_a + 1$ columns. As long as its number of rows is larger than r_a, the following statements are straight-forwardly proven from the rank deficiency of the Hankel matrix.

- If $r_a > r_b$, the Hankel matrix in the box, or starting from $\breve{x}_0$, has rank r_a, which is the number of poles.
- If $r_a \leq r_b$, the Hankel matrix starting from $\breve{x}_{r_b - r_a + 1}$ has rank r_a.

Extending the Hankel matrix in the box with extra columns on the right hand does not increase its rank since these extra columns of the extended Hankel

matrix can always be found in a Hankel matrix in the box with more rows at the bottom. Hence the above statements also hold for a Hankel matrix with a bigger size.

For a signal modeled as a sum of K exponentially damped sinusoids as in the model (1.4) of x_n, we have $r_a = K, r_b = K-1$. Hence the rank-K property of the Hankel matrix $\mathbf{X}$ in (2.1) is proven again here from the above conclusion.

Once these data are fed to a filter with a z-transfer function $\mathsf{H}(z)$, the z-transform of the filtered data is $\tilde{\mathsf{X}}(z) = \mathsf{X}(z)\mathsf{H}(z)$. A tilde, $\tilde{}$, is used above the variables to represent the filtered counterpart in this chapter. Denote the r_a, r_b parameters of $\tilde{\mathsf{X}}(z)$ by $r_{a(\tilde{x}_n)}, r_{b(\tilde{x}_n)}$ and we can see from the following that the new Hankel matrix constructed from the filtered data $\tilde{x}_n$ has not necessarily rank K.

Suppose an IIR filter with $r_a = r_{a(IIR)}$ and $r_b = r_{b(IIR)}$ is applied. Then $r_{a(\tilde{x}_n)} = K + r_{a(IIR)}$ and $r_{b(\tilde{x}_n)} = K - 1 + r_{b(IIR)}$ when no pole-zero cancellation occurs. Generally, $r_{a(IIR)} \geq r_{b(IIR)}$, thus $r_{a(\tilde{x}_n)} > r_{b(\tilde{x}_n)}$ and the new Hankel matrix starting from $\tilde{x}_0$ has rank $K + r_{a(IIR)}$ as a result of the first statement above. It is well known that the increase in the rank of the Hankel matrix and hence the number of unknowns results in larger variances of the estimated parameters. This is certainly to be avoided in accurate parameter estimation.

Suppose an FIR filter with $r_a = 0$ and $r_b = r_{b(FIR)}$ is applied. Then $r_{a(\tilde{x}_n)} = K$ and $r_{b(\tilde{x}_n)} = K - 1 + r_{b(FIR)}$ in general. According to the second statement above, the new Hankel matrix starting from $\tilde{x}_{r_{b(FIR)}}$ has rank K, while the Hankel matrix starting from $\tilde{x}_0$ has generally rank $K+1$. Therefore, the first $r_{b(FIR)}$ filtered data should be discarded to preserve the rank-K property. This unfortunately leads to a loss of accuracy of the estimated parameters. Intuitively this is to be expected because a lot of useful information is lost by deleting the initial part of the filtered signal which includes the portion of the decaying signal with the highest SNR.

To solve these problems of FIR and IIR prefiltering, we propose FIR and IIR filter matrices in the next section. The FIR or IIR prefiltering is then implemented via multiplication of the original Hankel data matrix $\mathbf{X}$ with the FIR or IIR filter matrix. The prefiltered matrix, or the product obtained thereof, keeps the rank-K property and can readily be processed by an SVD-based estimation algorithm.

3.2 FIR and IIR filter matrices

Most time-domain filtering processes are linear, and can thus be implemented by means of a matrix multiplication. Clearly stated, we should find a filter matrix $\mathbf{H} \in \mathbb{C}^{p\times p}$ such that $\tilde{\mathbf{x}} = \mathbf{H}\mathbf{x} \in \mathbb{C}^{p\times 1}$ corresponds to the filtered data vector, when $\mathbf{x} \in \mathbb{C}^{p\times 1}$ is the input data vector. If the input and output are row vectors instead of column vectors, the filter matrix is simply transposed and multiplied from the right as in $\tilde{\mathbf{x}}^T = \mathbf{x}^T\mathbf{H}^T$. The following two filter matrices, $\mathbf{H}_{\mathrm{FIR}t}$ and $\mathbf{H}_{\mathrm{IIR}}$, are constructed with $\mathbf{H}_{\mathrm{FIR}t}$ implementing the FIR filtering without end effect and $\mathbf{H}_{\mathrm{IIR}}$ implementing the IIR filtering. For comparison, the circular FIR filter matrix $\mathbf{H}_{\mathrm{FIR}c}$ used in [51] is reviewed as well.

3.2.1 FIR filter matrix: $\mathbf{H}_{\mathrm{FIR}}$

Suppose q is the length of a digital filter $\hat{h}_n$ and w_n is the window function, then we have the impulse response of the FIR filter $h_n = \hat{h}_n w_n$, $n = 1, 2, \cdots, q$. The output data from the FIR filter can be found using the discrete convolution or discrete circular convolution (denoted by $\otimes$ in this thesis) of input data $x_1, x_2, \ldots, x_p$ and the filter impulse response $h_1, h_2, \ldots, h_q$.

Because the discrete circular convolution assumes that both the sampled data $x_1, x_2, \ldots, x_p$ and $h_1, h_2, \ldots, h_q$ be one period of their respective infinite periodic functions, the imposed periodicity introduces what is known as an end effect [52]. This is a special phenomenon that arises when using a discrete circular convolution to approximate the continuous convolution. It can be avoided by appending sufficient zeros to both $x_1, x_2, \ldots, x_p$ and $h_1, h_2, \ldots, h_q$ to be convolved.

The following Toeplitz filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ below achieves such discrete circular convolution with zero padding.

$$\mathbf{H}_{\mathrm{FIR}t} \triangleq \begin{bmatrix} h_\tau & \cdots & h_1 & 0 & \cdots & \cdots & 0 \\ \vdots & & h_2 & \ddots & \ddots & & \vdots \\ h_q & & \vdots & & h_1 & \ddots & \vdots \\ 0 & \ddots & \vdots & & h_2 & \ddots & 0 \\ \vdots & \ddots & h_q & & \vdots & & h_1 \\ \vdots & & \ddots & \ddots & \vdots & & \vdots \\ 0 & \cdots & \cdots & 0 & h_q & \cdots & h_\tau \end{bmatrix} \in \mathbb{C}^{p\times p},$$

$$\text{where } \tau = \lfloor \frac{q+1}{2} \rfloor \tag{3.1}$$

The verification of FIR filtering using $\mathbf{H}_{\mathrm{FIR}t}$ can be done by writing out the result of discrete circular convolution with zero padding, that is,

$$
\begin{aligned}
&[\ \tilde{x}_1 \quad \tilde{x}_2 \ \cdots \ \tilde{x}_{p+q-1}]^T \\
&= [x_1\ x_2 \ \cdots \ x_p \ \underbrace{0\ 0 \ \cdots \ 0}_{q-1}]^T \otimes [h_1\ h_2 \ \cdots \ h_q \ \underbrace{0\ 0 \ \cdots \ 0}_{p-1}]^T
\end{aligned}
$$

$$
= \begin{cases}
\begin{bmatrix}
h_1x_1 & & & & & \\
h_1x_2+ & h_2x_1 & & & & \\
\vdots & & & \ddots & & \\
h_1x_q+ & h_2x_{q-1}+ & \cdots+\cdots & & & h_qx_1 \\
h_1x_{q+1}+ & h_2x_q+ & \cdots+\cdots & & & h_qx_2 \\
\vdots & & & & & \vdots \\
h_1x_p+ & h_2x_{p-1}+ & \cdots+\cdots & & & h_qx_{p+1-q} \\
 & h_2x_p+ & h_3x_{p-1}+ & \cdots+\cdots & & h_qx_{p+2-q} \\
 & & \ddots & & & \vdots \\
 & & & & h_{q-1}x_p+ & h_qx_{p-1} \\
 & & & & & h_qx_p
\end{bmatrix} & \text{if } q \le p \\
\begin{bmatrix}
h_1x_1 & & & & \\
h_1x_2+ & h_2x_1 & & & \\
\vdots & & & \ddots & \\
h_1x_p+ & h_2x_{p-1}+ & \cdots+ & & h_px_1 \\
h_2x_p+ & h_3x_{p-1}+ & \cdots+ & & h_{p+1}x_1 \\
\vdots & & & & \vdots \\
h_{q-p+1}x_p+ & h_{q-p}x_{p-1}+ & \cdots+ & & h_qx_1 \\
h_{q-p+2}x_p+ & h_{q-p+1}x_{p-1}+ & \cdots+ & h_qx_2 & \\
\vdots & & & & \\
h_{q-1}x_p+ & h_qx_{p-1} & & & \\
h_qx_p & & & &
\end{bmatrix} & \text{if } q > p
\end{cases}
$$

$$
= \begin{bmatrix}
h_1 & 0 & \cdots & \cdots & \cdots & 0 \\
\vdots & \ddots & \ddots & & & \vdots \\
h_\tau & \cdots & h_1 & 0 & & 0 \\
\vdots & & \ddots & \ddots & & \vdots \\
h_q & h_{q-1} & \cdots & h_1 & 0 & 0 \\
0 & h_q & h_{q-1} & \cdots & h_1 & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \cdots & \ddots & 0 \\
0 & & 0 & h_q & h_{q-1} & \cdots & h_1 \\
\vdots & & & \ddots & \ddots & & \vdots \\
0 & & & & 0 & h_q & \cdots & h_\tau \\
\vdots & & & & & \ddots & \ddots & \vdots \\
0 & & \cdots & \cdots & \cdots & & 0 & h_q
\end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}
$$

$$
= \mathbf{H}_{\mathrm{FIR0}}\mathbf{x} \qquad \text{(say)} \tag{3.2}
$$

Here, $\mathbf{H}_{\mathrm{FIR0}}$ is an FIR filter matrix without end effect. But it gives distortion at the tails of the filtered output due to the discontinuity caused by the imposed initial conditions. If we eliminate the filtered result in $\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{p+q-1}$ that correspond to zeros appended to x_n (or non-signal part) and adjust the time shift caused by $h_1, h_2, \cdots, h_q$ (the indices of $h_1, h_2, \cdots, h_q$ have a time delay of $\lfloor \frac{q-1}{2} \rfloor$, since the modulus of the filter impulse response is often symmetrical about time 0 for a filter with symmetric passband about its filter center), we end up with a truncated square FIR filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ of (3.1) in the outer box, which is obtained by deleting the $\tau - 1$ top rows and the $q - \tau$ bottom rows of the above filter matrix. A prefiltering using $\mathbf{H}_{\mathrm{FIR}t}$ removes most part of the distorted output and keeps those filtered data with highest SNR. Of course further truncation, as shown in the inner box, eliminates the distortion completely, but the number of output data available is very limited and the SNR is seriously decreased. Computer simulations indicate that the truncated square filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ gives best results among the several possible filter matrices discussed above. Therefore, we will only use $\mathbf{H}_{\mathrm{FIR}t}$ as FIR filter matrix in this chapter and compare $\mathbf{H}_{\mathrm{FIR}t}$ with $\mathbf{H}_{\mathrm{FIR}c}$ given below.

If we write out the discrete circular convolution result without zero padding, then for $p = q$, a circular filter matrix $\mathbf{H}_{\mathrm{FIR}c}$ used in [51] is obtained:

$$
\begin{aligned}
&[\tilde{x}_1 \ \ \tilde{x}_2 \quad \cdots \quad \tilde{x}_p]^T \\
&\qquad = [x_1 \ x_2 \ \cdots \ x_p]^T \otimes [h_1 \ h_2 \ \cdots \ h_p]^T
\end{aligned}
$$

$$= \begin{bmatrix} h_p & h_{p-1} & h_{p-2} & \cdots & h_1 \\ h_1 & h_p & h_{p-1} & \cdots & h_2 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ h_{p-2} & & \ddots & \ddots & h_{p-1} \\ h_{p-1} & h_{p-2} & \cdots & h_1 & h_p \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}$$

$$= \mathbf{H}_{\mathrm{FIR}c} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}$$

The advantages of $\mathbf{H}_{\mathrm{FIR}t}$ compared to the circular filter matrix $\mathbf{H}_{\mathrm{FIR}c}$ proposed in [51] lie in the zero-padding, which eliminates the end effect and allows an arbitrary choice of the filter length q as well. The end effect is caused by the summation of the discrete circular convolution result at both ends, if nonzero, of the cycle. In our case, the exponentially damped sinusoidal signal and the filter impulse response of a bandpass filter are both infinitely long in the time domain, that is, they never die out to exactly zero if a limited number of data samples is used. Therefore, zero-padding is always necessary for the discrete circular convolution here in order to avoid end effects. The benefits of $\mathbf{H}_{\mathrm{FIR}t}$ are more pronounced when the signal has small damping factors and/or the required FIR filter has a narrow passband in the frequency domain and thus a wide impulse response in the time domain.

Fig. 3.1 compares the results of filtering a single exponential using $\mathbf{H}_{\mathrm{FIR}c}$ and $\mathbf{H}_{\mathrm{FIR}t}$. It can be seen that there is an end-effect distortion at the tail when $\mathbf{H}_{\mathrm{FIR}c}$ is used, while the end effect is avoided when $\mathbf{H}_{\mathrm{FIR}t}$ is used.

3.2.2 IIR filter matrix: $\mathbf{H}_{\mathrm{IIR}}$

In applications in which we do not need to process the signals in real time, e.g. in NMR data quantification, we can consider both the backward and forward recursion.

Given the *backward recursion* of an rth-order IIR filter

$$\tilde{X}(z) = \frac{b_0 z^r + b_1 z^{r-1} + \cdots + b_{r-1} z + b_r}{a_0 z^r + a_1 z^{r-1} + \cdots + a_{r-1} z + a_r} X(z)$$

$$\text{or} \quad a_r \tilde{x}_n = b_r x_n + b_{r-1} x_{n+1} + \cdots + b_0 x_{n+r} - a_{r-1} \tilde{x}_{n+1} - \cdots - a_1 \tilde{x}_{n+r-1} - a_0 \tilde{x}_{n+r}, \qquad a_0 \neq 0, a_r \neq 0$$

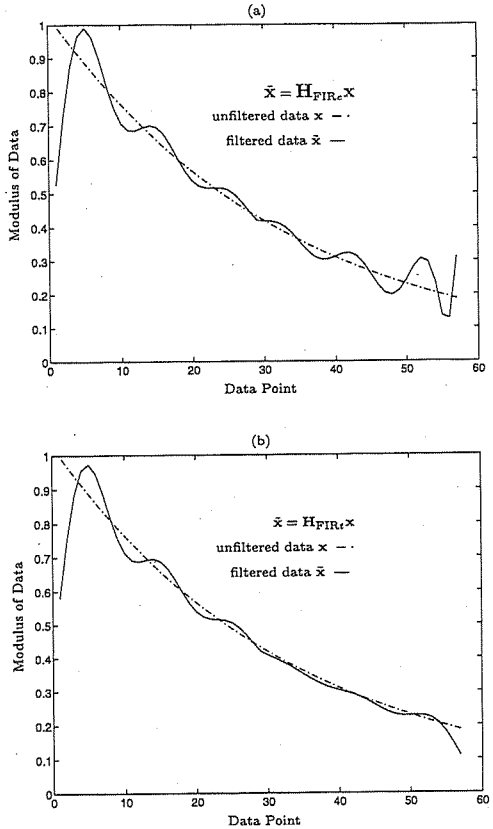


Figure 3.1: An example of FIR filtering using the filter matrix (a) $\mathbf{H}_{\mathrm{FIR}c}$ and (b) $\mathbf{H}_{\mathrm{FIR}t}$. The unfiltered data $\mathbf{x}$ is the first sinusoid of *Signal-4*, i.e., $e^{-0.02n\Delta t - j2\pi 0.2n\Delta t}$. The filter is the bandpass rectangular filter as used in Example 2 in the simulation section of this chapter.

with the initial conditions:

$$\begin{aligned} x_{p+1} &= x_{p+2} = \cdots = x_{p+r} = 0, \\ \tilde{x}_{p+1} &= \tilde{x}_{p+2} = \cdots = \tilde{x}_{p+r} = 0, \end{aligned}$$

we can set up a full-rank upper-triangular filter matrix for the backward recursion

$$\mathbf{H}_{\text{IIR}} = \begin{bmatrix} g_1 & g_2 & \cdots & g_p \\ 0 & g_1 & \cdots & g_{p-1} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_1 \end{bmatrix} \in \mathbb{C}^{p \times p} \tag{3.3}$$

with the elements

$$\begin{aligned} g_1 &= b_r/a_r, \\ g_i &= \begin{cases} (b_{r-i+1} - a_{r-1}g_{i-1} - a_{r-2}g_{i-2} - \cdots - a_{r-i+1}g_1)/a_r, & \text{for } 2 \le i \le r+1 \\ (-a_{r-1}g_{i-1} - a_{r-2}g_{i-2} - \cdots - a_0 g_{i-r})/a_r, & \text{for } r+2 \le i \le p \end{cases} \end{aligned}$$

This is verified by comparing the result of $\tilde{\mathbf{x}} = \mathbf{H}_{\text{IIR}}\mathbf{x}$ with that of the backward recursion.

The *forward recursion* of an IIR filter starts with the first samples of the signal instead of the last ones, as shown below.

Given the forward recursion of an rth-order IIR filter

$$\begin{aligned} \tilde{X}(z) &= \frac{b_0 + b_1 z^{-1} + \cdots + b_{r-1} z^{-(r-1)} + b_r z^{-r}}{a_0 + a_1 z^{-1} + \cdots + a_{r-1} z^{-(r-1)} + a_r z^{-r}} X(z) \\ \text{or } a_0 \tilde{x}_n &= b_0 x_n + b_1 x_{n-1} + \cdots + b_r x_{n-r} \\ &\quad - a_1 \tilde{x}_{n-1} - \cdots - a_{r-1}\tilde{x}_{n-(r-1)} - a_r \tilde{x}_{n-r}, \\ &\qquad\qquad a_0 \neq 0, a_r \neq 0 \end{aligned}$$

with the initial conditions:

$$\begin{aligned} x_{-r+1} &= x_{-r+2} = \cdots = x_{-1} = x_0 = 0, \\ \tilde{x}_{-r+1} &= \tilde{x}_{-r+2} = \cdots = \tilde{x}_{-1} = \tilde{x}_0 = 0, \end{aligned}$$

we can set up a full-rank lower-triangular filter matrix for the forward recursion

$$\mathbf{H}_{\text{IIR}} = \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ g_p & g_{p-1} & \cdots & g_1 \end{bmatrix} \in \mathbb{C}^{p \times p}$$

with the elements

$$\begin{aligned} g_1 &= b_0/a_0, \\ g_i &= \begin{cases} (b_{i-1} - a_1 g_{i-1} - a_2 g_{i-2} - \cdots - a_{i-1}g_1)/a_0, & \text{for } 2 \le i \le r+1 \\ (-a_1 g_{i-1} - a_2 g_{i-2} - \cdots - a_r g_{i-r})/a_0, & \text{for } r+2 \le i \le p \end{cases} \end{aligned}$$

Actually, the filter matrix for forward IIR can be obtained from that of backward IIR, that is, by first reversing the label sequence of a and b respectively, then calculating g_i in the same way, and finally forming a lower-triangular matrix or the transposition of an upper-triangular matrix.

For damped signals, a backward recursion, rather than a forward recursion, is recommended because the initial conditions can be easily fulfilled. That is, the signal should die out to zero after the pth data sample when using a backward recursion.

The phase distortion of an IIR or FIR (for non-linear phase) filter does not affect the parameter estimates. This is because the phases are calculated in a later step by fitting the model function using the estimated poles to the *original data*, not to the filtered data.

3.3 Prefiltered HTLS

By prefiltered HTLS we mean that a full-rank filter matrix $\mathbf{H}$ is multiplied with the Hankel data matrix $\mathbf{X}$ before applying the estimation algorithm HTLS. As formulated below, a right (resp., left) multiplication with a full-rank matrix $\mathbf{H}_r$ (resp., $\mathbf{H}_l$) of appropriate dimensions retains the rank-K property of the prefiltered data matrix and shift-invariant properties of $\tilde{\mathbf{U}}_K$ (resp., $\tilde{\mathbf{V}}_K^*$) in the absence of noise. These properties are required for a correct use of the HTLS method as shown below:

$$\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r = \tilde{\mathbf{U}}_K \tilde{\mathbf{\Sigma}}_K \tilde{\mathbf{V}}_K^H = \sum_{k=1}^{K} \tilde{\sigma}_k \tilde{\mathbf{u}}_k \tilde{\mathbf{v}}_k^H \;\Rightarrow\; \tilde{\mathbf{U}}_K^{\uparrow} = \tilde{\mathbf{U}}_{K\downarrow} \tilde{\mathbf{Z}}_K^{(U)}$$

$$\text{resp.,} \quad \tilde{\mathbf{X}} = \mathbf{H}_l\mathbf{X} = \tilde{\mathbf{U}}_K\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{V}}_K^H = \sum_{k=1}^{K}\tilde{\sigma}_k\tilde{\mathbf{u}}_k\tilde{\mathbf{v}}_k^H \Rightarrow \tilde{\mathbf{V}}_K^{*\uparrow} = \tilde{\mathbf{V}}_{K\downarrow}^{*}\tilde{\mathbf{Z}}_K^{(V)}$$

$\mathbf{H}_r$ and $\mathbf{H}_l$ are the filter matrices used in the case of right and left multiplication, respectively. Since a right (resp., left) multiplication with a full-rank square matrix does not change the column (resp., row) space of $\mathbf{X}$, there exists a unitary matrix $\mathbf{\Omega}$ such that $\tilde{\mathbf{U}} = \mathbf{U}\mathbf{\Omega}$ (resp., $\tilde{\mathbf{V}} = \mathbf{V}\mathbf{\Omega}$). Furthermore, it can be proven that $\mathbf{\Omega} = \begin{bmatrix} \mathbf{\Omega}_1 & 0 \\ 0 & \mathbf{\Omega}_2 \end{bmatrix}$ with $\mathbf{\Omega}_1 \in \mathbb{C}^{K\times K}$ (see Theorem 3-i at the end of the next section).

Algorithm 2

Outline of the prefiltered HTLS algorithm

Given. *A model-order estimate K of all peaks, an FIR or IIR bandpass filter encompassing the frequency range of interest, and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel data matrix $\mathbf{X}$ as in (2.1).*

Step 1. *Construct an FIR filter matrix $\mathbf{H} = \mathbf{H}_{\text{FIR}}$ as in (3.1) or an IIR filter matrix $\mathbf{H} = \mathbf{H}_{\text{IIR}}$ as in (3.3), where $\mathbf{H} \in \mathbb{C}^{p\times p}$. For the prefiltering via a right multiplication with the filter matrix, $p = M$ and $\mathbf{H}_r = \mathbf{H}^T$ are taken.*

Step 2. *Prefilter $\mathbf{X}$ by matrix multiplication:*

$$\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$$

Step 3. *Apply the pole-estimation part (Step 1-2) of the HTLS Algorithm 1 to $\tilde{\mathbf{X}}$ to find f_k and d_k, $k = 1, \cdots, K$.*

Step 4. *Apply the linear-parameter-estimation part (Step 3) of the HTLS Algorithm 1, to the original data x_n, $n = 0, 1, \cdots, N-1$, to find a_k and φ_k, $k = 1, \cdots, K$.*

For prefiltering via a left multiplication of the filter matrix, simply replace $p = M$ and $\mathbf{H}_r = \mathbf{H}^T$ by $p = L$ and $\mathbf{H}_l = \mathbf{H}$ in step 1, and then perform $\tilde{\mathbf{X}} = \mathbf{H}_l\mathbf{X}$ instead of $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$ in step 2. In this case, $\mathbf{V}_K$, instead of $\mathbf{U}_K$, is used for computing f_k, d_k in the HTLS Algorithm 1.

The remainder of this chapter considers only the case of *right* multiplication, i.e., the prefiltering is performed as $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$.

Observe that $\tilde{\mathbf{X}}$ is no longer a Hankel matrix. Each row of $\tilde{\mathbf{X}}$ is the filtered output of the corresponding row of $\mathbf{X}$. Since $\mathbf{X}$ is a Hankel matrix composed of L rows of shifted data segments, such row-by-row filtering can be visualized as segment-by-segment filtering. The rectangular truncation of each segment introduces the Gibbs phenomenon, which can be reduced by weighting each segment with an appropriate *window*. This can be realized via matrix multiplication $\mathbf{X}\mathbf{W}$, where $\mathbf{W}$ is a full-rank diagonal weighting matrix. In this way, the prefiltering proceeds as $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{W}\mathbf{H}_r$. Since $\mathbf{H}_r$ and $\mathbf{W}$, if any, are square and of full rank, the size and the effective rank of $\tilde{\mathbf{X}}$ are the same as those of $\mathbf{X}$. Therefore the computational work involved in the SVD of $[\tilde{\mathbf{U}}_{K\downarrow} \quad \tilde{\mathbf{U}}_K^{\uparrow}]$ for computing the TLS solution $\tilde{\mathbf{Z}}_K^{(U)}$ in the succeeding HTLS estimation is basically unchanged.

3.4 SVD relationships between the prefiltered and original data matrices

In the absence of noise, i.e., $\mathbf{X} = \overset{\circ}{\mathbf{X}}, \mathbf{E} = \mathbf{0}$, the singular values and vectors of the Hankel data matrix $\mathbf{X}$ always depend on the model parameters. Such dependency has been revealed in [53] for a simple signal with only two sinusoids. After prefiltering, the SVD of the prefiltered matrix $\tilde{\mathbf{X}}$ depends not only on the model parameters, but also on the filter matrix. The relationships between the SVD of $\tilde{\mathbf{X}}$, the SVD of $\mathbf{X}$ and the model parameters are investigated in this section in order to justify the prefiltering method and give some insight in the effects of prefiltering.

3.4.1 Definitions and implications

First of all, we give some useful definitions and their implications.

Definition 1 *$\mathbf{s}_k$ and $\mathbf{t}_k$ are the L-tuple and M-tuple vectors of the kth exponentially damped sinusoid in model (1.4), i.e.,*

$$\mathbf{s}_k \triangleq [1, z_k, z_k^2, \cdots, z_k^{L-1}]^T, \quad \mathbf{t}_k \triangleq [1, z_k, z_k^2, \cdots, z_k^{M-1}]^T$$

Definition 2 *Some factors are defined as* $\alpha_{ki} \triangleq \mathbf{s}_k^H\mathbf{s}_i$, $\beta_{ki} \triangleq \mathbf{t}_k^T\mathbf{t}_i^*$, $\tilde{\alpha}_{ki} \triangleq \mathbf{s}_k^H\mathbf{H}_l^H\mathbf{H}_l\mathbf{s}_i$, $\tilde{\beta}_{ki} \triangleq \mathbf{t}_k^T\mathbf{H}_r\mathbf{H}_r^H\mathbf{t}_i^*$, $\eta_{ki} \triangleq \mathbf{v}_k^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_i$.

Obviously, the factors α_{ki} and β_{ki} are the correlation of the kth and ith exponentially damped sinusoids. Since they are very similar ($\alpha_{ki} = \beta_{ki}$ for $L = M$), we only comment on one of them, say, β_{ki}, as the prefiltering via a right multiplication of $\mathbf{H}_r$ is considered. In general, $|\beta_{ki}|/\sqrt{\beta_{kk}\beta_{ii}} \leq 1$ according to Cauchy's inequality. With the concept of a normalized correlation $|\beta_{ki}|/\sqrt{\beta_{kk}\beta_{ii}}$, two extreme cases of signals can be described using the time-domain data model (1.4).

- When the two peaks *coincide*, i.e., $z_k = z_i$, then $|\beta_{ki}|/\sqrt{\beta_{kk}\beta_{ii}}$ reaches its maximum, 1, since $\beta_{ki} = \beta_{kk} = \beta_{ii}$ in this case.
- When the two peaks are *ideally nonoverlapping*, i.e., $d_k = d_i = 0$ and $(f_i - f_k)/f_s = \pm 0.5$, then $|\beta_{ki}|/\sqrt{\beta_{kk}\beta_{ii}}$ reaches its minimum, 0 if M is even, or $1/M$ if M is odd. This is because $\beta_{kk} = \beta_{ii} = \|\mathbf{t}_k\|^2 = M$, and

$$\begin{aligned}\beta_{ki} &= \sum_{n=0}^{M-1}(z_k z_i^*)^n = \frac{1-(z_k z_i^*)^M}{1-(z_k z_i^*)} \\ &= \frac{1-\exp[-M(d_k+d_i)/f_s + j2M\pi(f_k-f_i)/f_s]}{1-\exp[-(d_k+d_i)/f_s + j2\pi(f_k-f_i)/f_s]} \\ &= \begin{cases} 0 & (M \text{ even}) \\ 1 & (M \text{ odd}) \end{cases}\end{aligned}$$

The degree of overlap can also be analyzed in terms of the discrete Fourier transform (DFT) according to the following Theorem. Note that the element index of a vector is denoted in parenthesis in this chapter, since the vector subscript has been reserved for distinguishing different sinusoids.

Theorem 1 *Let $\mathcal{T}_k(m), m = 1, 2, \ldots, M$ be the M-point DFT of $\mathbf{t}_k(n)$, $n = 1, 2, \ldots, M$:*

$$\mathcal{T}_k(m) = \sum_{n=0}^{M-1} \mathbf{t}_k(n) e^{-j2\pi mn/M}$$

Then,

$$\beta_{ki} = \sum_{n=1}^{M} \mathbf{t}_k(n)\mathbf{t}_i^*(n) = \frac{1}{M}\sum_{m=1}^{M} \mathcal{T}_k(m)\mathcal{T}_i^*(m)$$

which implies that β_{ki} can be expressed as the correlation of $\mathbf{t}_i(n)$ and $\mathbf{t}_k(n)$, or the correlation of $\mathcal{T}_i(m)$ and $\mathcal{T}_k(m)$ scaled by a factor $1/M$.

Proof. Let $\mathbf{y}(l)$ be the correlation sum of $\mathbf{t}_k$ and $\mathbf{t}_i$:

$$\mathbf{y}(l) = \sum_{n=1}^{M} \mathbf{t}_k^*(n)\mathbf{t}_i(l+n)$$

Then $\mathbf{y}(0) = \mathbf{t}_k^H \mathbf{t}_i$. According to correlation theorem [52], the DFT of $\mathbf{y}(l)$ is $\mathcal{Y}(m) = \mathcal{T}_k^*(m)\mathcal{T}_i(m)$. Using the DFT property of $\mathbf{y}(0) = \frac{1}{M}\sum_{m=1}^{M}\mathcal{Y}(m)$, we have a form of Parseval's theorem

$$\mathbf{t}_k^H \mathbf{t}_i = \frac{1}{M}\sum_{m=1}^{M} \mathcal{T}_k^*(m)\mathcal{T}_i(m)$$

Hence, the theorem is proven by conjugation. □

Considering that $\mathcal{T}_k(m)$ has the form $a_k/[d_k + j2\pi(m/M - f_k/f_s)f_s]$ for exponentials, the smaller d_k, the narrower the peak k is in the spectrum. When the two peaks, k and i, are *approximately nonoverlapping*, that is, they are sufficiently far apart and have sufficiently small damping factors, their correlation is small as can be deduced intuitively from the frequency domain.

$\tilde{\alpha}_{ki}$ and $\tilde{\beta}_{ki}$ are simply the filtered counterpart of α_{ki} and β_{ki}. Similarly, $|\tilde{\alpha}_{ki}|/\sqrt{\tilde{\alpha}_{kk}\tilde{\alpha}_{ii}} \leq 1, |\tilde{\beta}_{ki}|/\sqrt{\tilde{\beta}_{kk}\tilde{\beta}_{ii}} \leq 1$, and $\tilde{\alpha}_{ki} = \tilde{\beta}_{ki}$ for $L = M$. The relationship between $\tilde{\alpha}_{ki}$ and α_{ki}, $\tilde{\beta}_{ki}$ and β_{ki} depends mainly on the filter matrix. Here we are going to *assume an ideal rectangular bandpass filter with unit gain at passband*, which means that the frequency response of the bandpass filter is 1 inside the passband and 0 outside the passband and the lower and upper cut-off frequencies can be denoted by f_l and f_u. This is a useful simplification in that it delivers neat expressions which provide insights in the prefiltering method [1]. As indicated in the following theorem, the 2-norm of such a rectangular filter matrix is 1, i.e., the first singular value of $\mathbf{H}_r$ is 1. Hence, $\tilde{\beta}_{kk} = \mathbf{t}_k^T \mathbf{H}_r \mathbf{H}_r^H \mathbf{t}_k^* \leq \|\mathbf{H}_r\|_2 \mathbf{t}_k^T \mathbf{t}_k^* = \beta_{kk}$.

Theorem 2 *Suppose $\mathbf{H}$ is a $q \times q$ filter matrix of a rectangular digital bandpass filter, i.e., when $\mathbf{x} \in \mathbb{C}^{q\times 1}$ is the input data vector, $\mathbf{Hx} \in \mathbb{C}^{q\times 1}$ is the filtered output vector, and the r-point DFT of this filter can be written as*

$$|\mathcal{H}(m)| = \begin{cases} 1 & m = m_1, \cdots, m_2 \quad \textit{(inside passband)} \\ 0 & m = 1, \cdots, m_1 - 1, m_2 + 1, \cdots, r \quad \textit{(outside passband)} \end{cases}$$

[1] To realize a rectangular bandpass filter, a filter matrix of infinite dimension should be used. This is impossible in practice but is used here to simplify the theoretical analysis. That is why the filter is called ideal.

where m_1 and m_2 are lower and upper cut-off frequencies of the digital filter and $r \geq q$. Then $\|\mathbf{H}\|_2 = 1$, or, the first singular value of $\mathbf{H}$ is $\sigma_{H1} = 1$.

Proof. Let $\mathbf{x} \in \mathbb{C}^{q\times 1}$, $\tilde{\mathbf{x}} = \mathbf{H}\mathbf{x} \in \mathbb{C}^{q\times 1}$, $\|\mathbf{x}\|_2^2 = \frac{1}{r}\sum_{m=1}^{r} |\mathcal{X}(m)|^2 = 1$ (normalized). Then $\tilde{\mathcal{X}}(m) = \mathcal{X}(m)\mathcal{H}(m)$, where $\mathcal{X}, \tilde{\mathcal{X}}$ and $\mathcal{H}$ are the r-point DFT of $\mathbf{x}, \tilde{\mathbf{x}}$ and the filter impulse response, respectively. Using DFT properties, it follows that

$$\begin{aligned}
\sigma_{H1}^2 &= \|\mathbf{H}\|_2^2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{H}\mathbf{x}\|_2^2 = \max_{\|\mathbf{x}\|_2=1} \|\tilde{\mathbf{x}}\|_2^2 \\
&= \max_{\|\mathbf{x}\|_2=1} \frac{1}{r}\sum_{m=1}^{r} |\tilde{\mathcal{X}}(m)|^2 \quad \text{(Parseval's theorem)} \\
&= \max_{\|\mathbf{x}\|_2=1} \frac{1}{r}\sum_{m=1}^{r} |\mathcal{X}(m)\mathcal{H}(m)|^2 \quad \text{(convolution theorem)} \\
&= \max_{\|\mathbf{x}\|_2=1} \frac{1}{r}\sum_{m=m_1}^{m_2} |\mathcal{X}(m)|^2 \\
&\leq \frac{1}{r}\sum_{m=1}^{r} |\mathcal{X}(m)|^2 \\
&= \|\mathbf{x}\|_2^2 \quad \text{(Parseval's theorem)} \\
&= 1
\end{aligned}$$

The equality holds when the nonzero spectrum of signal $\mathbf{x}$ is completely inside the passband of the filter, i.e., $\mathcal{X}$ is zero outside the passband. On the other hand, $\tilde{\mathbf{x}} = \mathbf{H}\mathbf{x} = 0$ when $\mathcal{X}$ is zero inside the passband. □

3.4.2 SVD expressions of X and $\tilde{\mathbf{X}}$ in terms of signal parameters

The above definitions are useful in that they simplify the expressions of $\mathbf{X}$, $\tilde{\mathbf{X}}$ and their singular values/vectors as functions of the signal parameters and the filter matrix. For example, the Vandermonde decomposition of $\mathbf{X}$ in (2.7) can be written in the form

$$\mathbf{X} = [\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_K]\ \mathrm{diag}(c_1, c_2, \cdots, c_K)\ [\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_K]^T = \sum_{k=1}^{K} c_k \mathbf{s}_k \mathbf{t}_k^T$$

To express the SVD in terms of the signal parameters, we invoke the following elementary properties relating eigenvalue decompositions and SVD's. For $k = 1, \cdots, K$, the first K nonzero eigenvalues, $\tilde{\lambda}_k$, of $\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H$ and $\tilde{\mathbf{X}}^H\tilde{\mathbf{X}}$ are $\tilde{\sigma}_k^2$. The corresponding eigenvectors of $\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H$ and $\tilde{\mathbf{X}}^H\tilde{\mathbf{X}}$ are $\tilde{\mathbf{u}}_k$ and $\tilde{\mathbf{v}}_k$, respectively. When $K = 2$, the prefiltered data matrix is simply

$$\tilde{\mathbf{X}} = (c_1\mathbf{s}_1\mathbf{t}_1^T + c_2\mathbf{s}_2\mathbf{t}_2^T)\mathbf{H}_r = (\sigma_1\mathbf{u}_1\mathbf{v}_1^H + \sigma_2\mathbf{u}_2\mathbf{v}_2^H)\mathbf{H}_r$$

Based on this description and our earlier definitions, the following relationships between the SVD and the signal parameters are proven in Theorem 3-**ii** at the end of this section. The left and right singular vectors $\tilde{\mathbf{u}}_k$ and $\tilde{\mathbf{v}}_k$ $(k = 1, 2)$ can be written as the linear combinations of $\mathbf{s}_1, \mathbf{s}_2$ and $\mathbf{t}_1^*, \mathbf{t}_2^*$:

$$\tilde{\mathbf{u}}_k = \tilde{b}_{uk1}\mathbf{s}_1 + \tilde{b}_{uk2}\mathbf{s}_2, \qquad \tilde{\mathbf{v}}_k = \mathbf{H}_r^H(\tilde{b}_{vk1}\mathbf{t}_1^* + \tilde{b}_{vk2}\mathbf{t}_2^*)$$

The eigenvalues $\tilde{\lambda}_k = \tilde{\sigma}_k^2$ $(k = 1, 2)$ are the roots of the quadratic equation:

$$\begin{aligned}
&\tilde{\lambda}^2 - [(\alpha_{11}\tilde{\beta}_{11}|c_1|^2 + \alpha_{22}\tilde{\beta}_{22}|c_2|^2 + 2\mathrm{Re}(\alpha_{12}^*\tilde{\beta}_{12}c_1c_2^*)]\tilde{\lambda} \\
&+|c_1|^2|c_2|^2(\alpha_{11}\alpha_{22} - \alpha_{12}\alpha_{21})(\tilde{\beta}_{11}\tilde{\beta}_{22} - \tilde{\beta}_{12}\tilde{\beta}_{21}) = 0
\end{aligned}$$

If no filter matrix is applied, or $\mathbf{H}_r = \mathbf{I}$, then $\tilde{\beta}_{ki} = \beta_{ki}$, the above quadratic equation reduces to that of λ_k, and $\mathbf{u}_k = b_{uk1}\mathbf{s}_1 + b_{uk2}\mathbf{s}_2$, $\mathbf{v}_k = b_{vk1}\mathbf{t}_1^* + b_{vk2}\mathbf{t}_2^*$, as given in [53].

Consider a special case of two ideally nonoverlapping sinusoids without applying prefiltering. Suppose L and M are even, it can be easily verified that $\alpha_{11} = \alpha_{22} = L, \beta_{11} = \beta_{22} = M, \alpha_{12} = \alpha_{21} = 0, \beta_{12} = \beta_{21} = 0$. This means that $\mathbf{s}_1$ and $\mathbf{s}_2$, $\mathbf{t}_1^*$ and $\mathbf{t}_2^*$ are already orthogonal and can be chosen as the singular vectors $\mathbf{u}_1$ and $\mathbf{u}_2$, $\mathbf{v}_1$ and $\mathbf{v}_2$ after appropriate scaling. The solution λ_k to the quadratic equation in this case is also quite simple. Without loss of generality, the peak with the largest power is numbered as the first peak. Then $\lambda_1 = \sigma_1^2 = \alpha_{11}\beta_{11}|c_1|^2 = LM|c_1|^2, \lambda_2 = \sigma_2^2 = \alpha_{22}\beta_{22}|c_2|^2 = LM|c_2|^2$.

When the two-peak signal is more general, and not ideally nonoverlapping, closed-form solutions of σ_k and $\tilde{\sigma}_k$ $(k = 1, 2)$ are always obtainable from the general root expressions of the quadratic equations.

3.4.3 SVD relationships between X and $\tilde{\mathbf{X}}$

In the previous subsection, we have derived the explicit expressions of $\tilde{\sigma}_k, \sigma_k$ etc. in function of the signal parameters. Here, we are going to compare directly the singular values/vectors of the filtered and unfiltered data matrix.

Recall that $\tilde{\mathbf{U}}_K = \mathbf{U}_K\mathbf{\Omega}_1$ (or see Theorem 3-**i**). For $K = 2$, we define the 2×2 matrix $\mathbf{\Omega}_1$ as $[\omega_{ki}]$, where ω_{ki} is the element on the kth row and

ith column of $\mathbf{\Omega}_1$. After some manipulations (see Theorem 3-**iii** at the end of this section), we get the general formulae below, which clearly reveal the relationships between the SVD of the filtered data matrix and the original one:

$$\begin{aligned}
\tilde{\sigma}_1^2 &= \frac{1}{2}\left[(\sigma_1^2\eta_{11}+\sigma_2^2\eta_{22}) + \sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2+(\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22})^2}\right] \\
\tilde{\sigma}_2^2 &= \frac{1}{2}\left[(\sigma_1^2\eta_{11}+\sigma_2^2\eta_{22}) - \sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2+(\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22})^2}\right] \\
|\omega_{11}|^2 &= |\omega_{22}|^2 = \frac{1}{2} + \frac{\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22}}{2\sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2+(\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22})^2}} \\
|\omega_{21}|^2 &= |\omega_{12}|^2 = \frac{1}{2} - \frac{\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22}}{2\sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2+(\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22})^2}}
\end{aligned} \tag{3.4}$$

In general, the factor $\eta_{ki} = \mathbf{v}_k^H \mathbf{H}_r \mathbf{H}_r^H \mathbf{v}_i$ depends on both the signal parameters and the filter matrix. Expressing $\mathbf{v}_k$ as a linear combination of $\mathbf{t}_k^*$, we get:

$$\eta_{ki} = b_{vk1}^* b_{vi1} \bar{\beta}_{11} + b_{vk2}^* b_{vi2} \bar{\beta}_{22} + b_{vk1}^* b_{vi2} \bar{\beta}_{12} + b_{vk2}^* b_{vi1} \bar{\beta}_{21} \tag{3.5}$$

For simplification, we consider three ideal cases of filtering:

a. both sinusoids (peak 1 and 2) are successfully selected.

b. the first sinusoid (peak 1) is successfully selected and the second sinusoid (peak 2) is completely suppressed.

c. the second sinusoid (peak 2) is successfully selected and the first sinusoid (peak 1) is completely suppressed.

By *successful selection* and *complete suppression* we mean that the filtering retains the wanted peak without distortion. This can be realized under the assumption that an ideal rectangular filter is designed and the two peaks of the original signal have both zero damping factors such that infinitely long tails in the spectrum are avoided. The values of the factors $\bar{\beta}_{ki}$ and thus η_{ki} for the three ideal cases of filtering are derived in Theorem 3-**iv** and listed in Table 3.1. Substituting the values of η_{ki} into (3.4) yields neat expressions of the SVD relationship between $\tilde{\mathbf{X}}$ and $\mathbf{X}$ as listed in the same table. These relationships clearly indicate that the wanted peaks are emphasized while the unwanted peaks are suppressed successfully after prefiltering via matrix multiplication.

Of course the results listed in Table 3.1 are derived in Theorem 3-**iii, iv** using the following assumption: (1) the filter is an ideal rectangular bandpass filter preserving the wanted peaks and ruling out the unwanted peaks in the

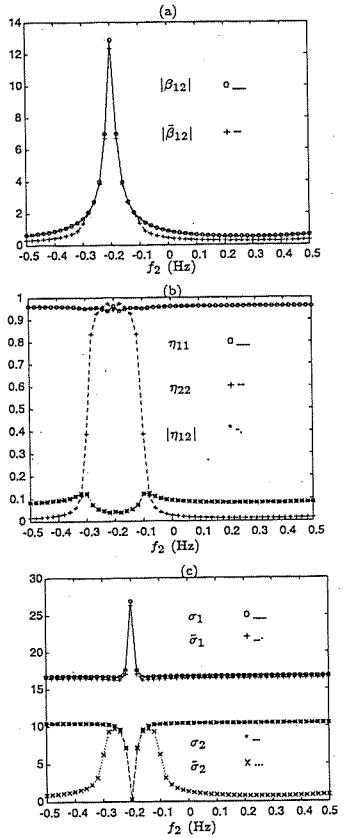


Figure 3.2: (a) $|\beta_{12}|$ and $|\bar{\beta}_{12}|$ (b) $|\eta_{11}|$, $|\eta_{22}|$ and $|\eta_{12}|$ (c) σ_1, σ_2, $\bar{\sigma}_1$ and $\bar{\sigma}_2$ of a two-peak signal with $f_1 = -0.2$ Hz, f_2 variable, $d_1 = 0.03$ rad/s, $d_2 = 0.05$ rad/s, and $c_1 = c_2 = 1$. $f_l = f_1 - 0.1$ Hz, $f_u = f_1 + 0.1$ Hz, $q = 99$, $L = 62$, $M = 60$, $N = 121$.

filter characteristics	a. select both peaks	b. select the 1st peak	c. select the 2nd peak
$\tilde{\beta}_{11} =$	β_{11}	β_{11}	0
$\tilde{\beta}_{22} =$	β_{22}	0	β_{22}
$\tilde{\beta}_{12} =$	β_{12}	0	0
$\tilde{\beta}_{21} =$	β_{21}	0	0
$\eta_{11} =$	1	1	0
$\eta_{22} =$	1	0	1
$\eta_{12} =$	0	0	0
$\eta_{21} =$	0	0	0
$\tilde{\sigma}_1 =$	σ_1	σ_1	σ_2
$\tilde{\sigma}_2 =$	σ_2	0	0
$\Omega_1 =$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

Table 3.1: Three ideal cases of filtering using a rectangular filter.

spectrum; (2) the damping factors of the sinusoids are zero (which enables a successful selection). In practice, the filter passband is not exactly rectangular and the signal to be processed is damped. But, again, a nice feature of the SVD is its robustness with respect to mild violations of these conditions. If both peaks are to be selected and the two peaks are sufficiently narrow and not far away in the spectrum, which means that d_1 and d_2 are close to 0, then the two peaks can be enhanced considerably by a proper design of the filter preserving the two peaks and suppressing the noise outside the passband. If only one of the two peaks is wanted, the selected peak can also be emphasized, provided that the wanted and unwanted peaks are sufficiently narrow and well separated, that is, d_1 and d_2 are close to 0 and $|f_2 - f_1|/f_s$ is close to 0.5. To give an intuitive idea of these results in general cases, we show in Fig. 3.2 the variations of $|\tilde{\beta}_{12}|, |\eta_{ki}|, \sigma_k$ and $\tilde{\sigma}_k$ $(k, i = 1, 2)$ when one of the two peaks varies along the frequency axis and peak 1 is selected. It can clearly be seen that $\tilde{\sigma}_1 \approx \sigma_1, \tilde{\sigma}_2 \approx \sigma_2$ when f_2 is around $f_1 = -0.2$ Hz or inside the filter passband, that is, when both peaks are selected; while $\tilde{\sigma}_1 \approx \sigma_1, \tilde{\sigma}_2 \approx 0$ when f_2 is around $f_1 + 0.5$ Hz $= 0.3$ Hz, which corresponds to the case that the two peaks are nonoverlapping and only the first peak is selected.

Theorem 3 *Let $\mathbf{X}$ be the rank-K Hankel data matrix of dimension $L \times M$ as defined in (2.1), and $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$ be the $L \times M$ filtered data matrix where $\mathbf{H}_r$*

is full rank and square. Suppose the SVD of $\mathbf{X}$ and $\tilde{\mathbf{X}}$ are $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^H$ and $\tilde{\mathbf{X}} = \tilde{\mathbf{U}}\tilde{\Sigma}\tilde{\mathbf{V}}^H$.

i. *There exists a unitary matrix $\Omega \in \mathbb{C}^{L\times L}$ such that $\tilde{\mathbf{U}} = \mathbf{U}\Omega$, where $\Omega = \begin{bmatrix} \Omega_1 & 0 \\ 0 & \Omega_2 \end{bmatrix}$ and $\Omega_1 \in \mathbb{C}^{K\times K}$.*

ii. *For a signal with $K = 2$ exponentials, the singular vectors $\tilde{\mathbf{u}}_k$ and $\tilde{\mathbf{v}}_k$ $(k = 1, 2)$ can be expressed as linear combinations of $\mathbf{s}_1$, $\mathbf{s}_2$ and $\mathbf{t}_1^*$, $\mathbf{t}_2^*$ defined in definition 1*

$$\tilde{\mathbf{u}}_k = \tilde{b}_{uk1}\mathbf{s}_1 + \tilde{b}_{uk2}\mathbf{s}_2, \qquad \tilde{\mathbf{v}}_k = \mathbf{H}_r^H(\tilde{b}_{vk1}\mathbf{t}_1^* + \tilde{b}_{vk2}\mathbf{t}_2^*)$$

the eigenvalues $\tilde{\lambda}_k$, $k = 1, 2$, are the roots of the following quadratic equation using definition 2

$$\tilde{\lambda}^2 - [\alpha_{11}\tilde{\beta}_{11}|c_1|^2 + \alpha_{22}\tilde{\beta}_{22}|c_2|^2 + 2Re(\alpha_{12}^*\tilde{\beta}_{12}c_1c_2^*)]\tilde{\lambda} + |c_1|^2|c_2|^2(\alpha_{11}\alpha_{22} - \alpha_{12}\alpha_{21})(\tilde{\beta}_{11}\tilde{\beta}_{22} - \tilde{\beta}_{12}\tilde{\beta}_{21}) = 0$$

and $\tilde{\nu}_{uk} = \tilde{b}_{uk1}/\tilde{b}_{uk2}$, $k = 1, 2$ are the roots of the quadratic equation

$$(\alpha_{21}\tilde{\beta}_{22}|c_2|^2 + \alpha_{11}\tilde{\beta}_{21}c_1^*c_2)\tilde{\nu}_u^2 + (\alpha_{22}\tilde{\beta}_{22}|c_2|^2 - \alpha_{11}\tilde{\beta}_{11}|c_1|^2 + \alpha_{12}\tilde{\beta}_{21}c_1^*c_2 - \alpha_{21}\tilde{\beta}_{12}c_1c_2^*)\tilde{\nu}_u - (\alpha_{12}\tilde{\beta}_{11}|c_1|^2 + \alpha_{22}\tilde{\beta}_{12}c_1c_2^*) = 0$$

iii. *For a signal with $K = 2$ exponentials, define $\Omega = \begin{bmatrix} \omega_{11} & \omega_{12} \\ \omega_{21} & \omega_{22} \end{bmatrix}$. The relationship between the singular values $\tilde{\sigma}_k$ and σ_k as well as the relationship between the singular vectors $\tilde{\mathbf{u}}_k$ and $\mathbf{u}_k$ can be expressed as*

$$\begin{aligned}
\tilde{\sigma}_1^2 &= \frac{1}{2}\left[(\sigma_1^2\eta_{11} + \sigma_2^2\eta_{22}) + \sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2}\right] \\
\tilde{\sigma}_2^2 &= \frac{1}{2}\left[(\sigma_1^2\eta_{11} + \sigma_2^2\eta_{22}) - \sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2}\right] \\
|\omega_{11}|^2 &= |\omega_{22}|^2 = \frac{1}{2} + \frac{\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22}}{2\sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2}} \\
|\omega_{21}|^2 &= |\omega_{12}|^2 = \frac{1}{2} - \frac{\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22}}{2\sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2}}
\end{aligned}$$

iv. *For a signal with ideally nonoverlapping exponentials, apply a rectangular filter. Then,*

1. we have

$\tilde{\beta}_{ki} = \beta_{ki}$ *when both peak k and i are totally encompassed inside the filter passband*

$\tilde{\beta}_{ki} = 0$ *when either peak k or i is totally ruled out of the filter passband*

2. using the classification in Subsection 3.4.3, we have

a. $\tilde{\beta}_{11} = \beta_{11}$, $\tilde{\beta}_{22} = \beta_{22}$, $\tilde{\beta}_{12} = \beta_{12}$, $\tilde{\beta}_{21} = \beta_{21}$;
$\eta_{11} = \eta_{22} = 1$, $\eta_{12} = \eta_{21} = 0$.

b. $\tilde{\beta}_{11} = \beta_{11}$, $\tilde{\beta}_{22} = \tilde{\beta}_{12} = \tilde{\beta}_{21} = 0$;
$\eta_{11} = 1$, $\eta_{22} = \eta_{12} = \eta_{21} = 0$.

c. $\tilde{\beta}_{22} = \beta_{22}$, $\tilde{\beta}_{11} = \tilde{\beta}_{12} = \tilde{\beta}_{21} = 0$;
$\eta_{22} = 1$, $\eta_{11} = \eta_{12} = \eta_{21} = 0$.

Proof.

i. Let

$$\Omega = \mathbf{U}^{\mathbf{H}}\tilde{\mathbf{U}} = \begin{bmatrix} \Omega_1 & \Omega_4 \\ \Omega_3 & \Omega_2 \end{bmatrix}$$

Since $\mathbf{U}$ and $\tilde{\mathbf{U}}$ are both unitary matrices, Ω is also unitary and

$$\tilde{\mathbf{U}} = \mathbf{U}\Omega = [\mathbf{U}_K,\ \mathbf{U}_{M-K}] \begin{bmatrix} \Omega_1 & \Omega_4 \\ \Omega_3 & \Omega_2 \end{bmatrix}$$

From

$$\mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r = \tilde{\mathbf{U}}_K\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{V}}_K^H$$

$$\Rightarrow \quad \tilde{\mathbf{U}}_K = \mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\tilde{\mathbf{V}}_K\tilde{\mathbf{\Sigma}}_K^{-1}$$

$$\Rightarrow \quad \Omega_1 = \mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\tilde{\mathbf{V}}_K\tilde{\mathbf{\Sigma}}_K^{-1}, \text{ and } \Omega_3 = 0$$

Substituting $\Omega_3 = 0$ into $\Omega^H\Omega = \mathbf{I}$ and $\Omega\Omega^H = \mathbf{I}$ leads to $\Omega_4 = 0$. It is straight forward that Ω_1 and Ω_2 are unitary matrices.

ii. Using the Vandermonde decomposition of $\mathbf{X}$ and $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$, we have

$$\begin{aligned} \tilde{\mathbf{X}}\tilde{\mathbf{X}}^H &= (c_1\mathbf{s}_1\mathbf{t}_1^T + c_2\mathbf{s}_2\mathbf{t}_2^T)\mathbf{H}_r\mathbf{H}_r^H(c_1\mathbf{s}_1\mathbf{t}_1^T + c_2\mathbf{s}_2\mathbf{t}_2^T)^H \\ &= |c_1|^2\tilde{\beta}_{11}\mathbf{s}_1\mathbf{s}_1^H + c_1c_2^*\tilde{\beta}_{12}\mathbf{s}_1\mathbf{s}_2^H + c_1^*c_2\tilde{\beta}_{21}\mathbf{s}_2\mathbf{s}_1^H \\ &\quad +|c_2|^2\tilde{\beta}_{22}\mathbf{s}_2\mathbf{s}_2^H \end{aligned} \tag{3.6}$$

A direct observation is that the singular vectors $\tilde{\mathbf{u}}_k$ can be written as a linear combination of $\mathbf{s}_1, \mathbf{s}_2$

$$\begin{aligned} \tilde{\mathbf{u}}_k &= \tilde{b}_{uk1}\mathbf{s}_1 + \tilde{b}_{uk2}\mathbf{s}_2 \\ \text{or} \quad \tilde{\mathbf{u}}_k &= \tilde{b}_{uk2}(\tilde{\nu}_{uk}\mathbf{s}_1 + \mathbf{s}_2), \quad \text{when} \quad \tilde{b}_{uk2} \neq 0, \ \ \tilde{\nu}_{uk} = \tilde{b}_{uk1}/\tilde{b}_{uk2} \end{aligned}$$

Hence,

$$\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H(\tilde{\nu}_{uk}\mathbf{s}_1 + \mathbf{s}_2) = \tilde{\lambda}_k(\tilde{\nu}_{uk}\mathbf{s}_1 + \mathbf{s}_2) \quad \text{for } k = 1, 2$$

Using (3.6), it can be expanded as

$$\begin{aligned} &\tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{11}|c_1|^2\mathbf{s}_1 + \alpha_{12}\tilde{\beta}_{11}|c_1|^2\mathbf{s}_1 + \tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{12}c_1c_2^*\mathbf{s}_1 \\ +\ &\alpha_{22}\tilde{\beta}_{12}c_1c_2^*\mathbf{s}_1 + \tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{21}c_1^*c_2\mathbf{s}_2 + \alpha_{12}\tilde{\beta}_{21}c_1^*c_2\mathbf{s}_2 \\ +\ &\tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{22}|c_2|^2\mathbf{s}_2 + \alpha_{22}\tilde{\beta}_{22}|c_2|^2\mathbf{s}_2 = \tilde{\lambda}_k\tilde{\nu}_{uk}\mathbf{s}_1 + \tilde{\lambda}_k\mathbf{s}_2 \end{aligned} \tag{3.7}$$

Generally, $\mathbf{s}_1, \mathbf{s}_2$ are linearly independent, that is, if $b_1\mathbf{s}_1 + b_2\mathbf{s}_2 = 0$, then $b_1 = b_2 = 0$. Applying this to (3.7) gives us

$$\begin{aligned} \tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{11}|c_1|^2 \ +\ &\alpha_{12}\tilde{\beta}_{11}|c_1|^2 + \tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{12}c_1c_2^* \\ &+\alpha_{22}\tilde{\beta}_{12}c_1c_2^* = \tilde{\lambda}_k\tilde{\nu}_{uk} \end{aligned} \tag{3.8}$$

$$\begin{aligned} \tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{21}c_1^*c_2 \ +\ &\alpha_{12}\tilde{\beta}_{21}c_1^*c_2 + \tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{22}|c_2|^2 \\ &+\alpha_{22}\tilde{\beta}_{22}|c_2|^2 = \tilde{\lambda}_k \end{aligned} \tag{3.9}$$

Substituting, respectively,

$$\tilde{\lambda}_k = \tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{21}c_1^*c_2 + \alpha_{12}\tilde{\beta}_{21}c_1^*c_2 + \tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{22}|c_2|^2 + \alpha_{22}\tilde{\beta}_{22}|c_2|^2$$

and

$$\tilde{\nu}_{uk} = (\tilde{\lambda}_k - \alpha_{12}\tilde{\beta}_{21}c_1^*c_2 - \alpha_{22}\tilde{\beta}_{22}|c_2|^2)/(\alpha_{11}\tilde{\beta}_{21}c_1^*c_2 + \alpha_{21}\tilde{\beta}_{22}|c_2|^2)$$

obtained from (3.9), into (3.8) leads us to the quadratic equations of $\tilde{\nu}_u$ and $\tilde{\lambda}$ in the theorem.

Similarly, starting from the expansion of $\tilde{\mathbf{X}}^H\tilde{\mathbf{X}}$ instead of $\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H$, we get

$$\tilde{\mathbf{v}}_k = \mathbf{H}_r^H(\tilde{b}_{vk1}\mathbf{t}_1^* + \tilde{b}_{vk2}\mathbf{t}_2^*)$$

iii. Equating

$$\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H = \tilde{\mathbf{U}}_K\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{U}}_K^H$$

and

$$\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H = \mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{V}_K\mathbf{\Sigma}_K\mathbf{U}_K^H$$

and using $\tilde{\mathbf{U}}_K = \mathbf{U}_K\mathbf{\Omega}_1$, we get the relationship between $\tilde{\mathbf{\Sigma}}_K$ and $\mathbf{\Sigma}_K$

$$\mathbf{\Omega}_1\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{\Sigma}}_K\mathbf{\Omega}_1^H = \mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{V}_K\mathbf{\Sigma}_K \tag{3.10}$$

For $K = 2$, we have

$$\mathbf{\Omega}_1\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{\Sigma}}_K\mathbf{\Omega}_1^H = \begin{bmatrix} \tilde{\sigma}_1^2|\omega_{11}|^2 + \tilde{\sigma}_2^2|\omega_{12}|^2 & \tilde{\sigma}_1^2\omega_{11}\omega_{21}^* + \tilde{\sigma}_2^2\omega_{22}^*\omega_{12} \\ \tilde{\sigma}_1^2\omega_{11}^*\omega_{21} + \tilde{\sigma}_2^2\omega_{22}\omega_{12}^* & \tilde{\sigma}_1^2|\omega_{21}|^2 + \tilde{\sigma}_2^2|\omega_{22}|^2 \end{bmatrix}$$

$$\mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{V}_K\mathbf{\Sigma}_K = \begin{bmatrix} \sigma_1^2\mathbf{v}_1^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_1 & \sigma_1\sigma_2\mathbf{v}_1^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_2 \\ \sigma_1\sigma_2\mathbf{v}_2^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_1 & \sigma_2^2\mathbf{v}_2^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_2 \end{bmatrix}$$
$$= \begin{bmatrix} \sigma_1^2\eta_{11} & \sigma_1\sigma_2\eta_{12} \\ \sigma_1\sigma_2\eta_{21} & \sigma_2^2\eta_{22} \end{bmatrix}$$

Using (3.10) and $\mathbf{\Omega}_1^H\mathbf{\Omega}_1 = \mathbf{\Omega}_1\mathbf{\Omega}_1^H = \mathbf{I}$, we have

$$\begin{aligned} \tilde{\sigma}_1^2|\omega_{11}|^2 + \tilde{\sigma}_2^2(1 - |\omega_{11}|^2) &= \sigma_1^2\eta_{11}, && (3.11)\\ \tilde{\sigma}_1^2(1 - |\omega_{11}|^2) + \tilde{\sigma}_2^2|\omega_{11}|^2 &= \sigma_2^2\eta_{22}, && (3.12)\\ \tilde{\sigma}_1^2\omega_{11}^*\omega_{21} + \tilde{\sigma}_2^2\omega_{22}\omega_{12}^* &= \sigma_1\sigma_2\eta_{21}, && (3.13)\\ |\omega_{11}| = |\omega_{22}|, \quad & |\omega_{21}| = |\omega_{12}|, && (3.14) \end{aligned}$$

First derive $\tilde{\sigma}_k$ from (3.11) and (3.12):

$$\begin{aligned} \tilde{\sigma}_1^2 &= \frac{\sigma_1^2\eta_{11}|\omega_{11}|^2 - \sigma_2^2\eta_{22}(1 - |\omega_{11}|^2)}{2|\omega_{11}|^2 - 1} \\ \tilde{\sigma}_2^2 &= \frac{-\sigma_1^2\eta_{11}(1 - |\omega_{11}|^2) + \sigma_2^2\eta_{22}|\omega_{11}|^2}{2|\omega_{11}|^2 - 1} \end{aligned} \tag{3.15}$$

then substitute these into (3.13). After some manipulations and using (3.14), we have

$$(|\omega_{11}|^4 - |\omega_{11}|^2)[4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2] + \sigma_1^2\sigma_2^2|\eta_{21}|^2 = 0$$

Solve for $|\omega_{11}|^2$ from this equation and put it back into (3.15). Considering $\tilde{\sigma}_1 \geq \tilde{\sigma}_2$, the theorem is proven.

iv. 1. Let $\mathbf{x}_k = \mathbf{t}_k^*/\|\mathbf{t}_k\|_2, \tilde{\mathbf{x}}_k = \mathbf{H}_r^T\mathbf{x}_k$, then $\|\mathbf{x}\|_2^2 = \frac{1}{r}\sum_{m=1}^r |\mathcal{X}(m)|^2 = 1$ (normalized), and $\tilde{\mathcal{X}}_k(m) = \mathcal{X}_k(m)\mathcal{H}(m), m = 1, 2, \ldots, r$, where $\mathcal{X}_k, \tilde{\mathcal{X}}_k$ and $\mathcal{H}$ are the r-point DFT of $\mathbf{x}_k, \tilde{\mathbf{x}}_k$ and the filter impulse response, respectively. Using DFT properties, it follows that

$$\begin{aligned} \frac{\tilde{\beta}_{ki}}{\sqrt{\beta_{kk}\beta_{ii}}} &= \frac{\mathbf{t}_k^T\mathbf{H}_r\mathbf{H}_r^H\mathbf{t}_i^*}{\|\mathbf{t}_k\|_2\|\mathbf{t}_i\|_2} = \mathbf{x}_k^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{x}_i = \tilde{\mathbf{x}}_k^H\tilde{\mathbf{x}}_i \\ &= \frac{1}{r}\sum_{m=1}^r \tilde{\mathcal{X}}_k^*(m)\tilde{\mathcal{X}}_i(m) \quad \text{(see Theorem (1))} \\ &= \frac{1}{r}\sum_{m=1}^r \mathcal{X}_k^*(m)\mathcal{X}_i(m)|\mathcal{H}(m)|^2 \text{ (convolution theorem)} \\ &= \begin{cases} \frac{1}{r}\sum_{m=1}^r \mathcal{X}_k^*(m)\mathcal{X}_i(m) = \mathbf{x}_k^H\mathbf{x}_i = \frac{\beta_{ki}}{\sqrt{\beta_{kk}\beta_{ii}}} \\ \qquad \text{(when both peak } k \text{ and } i \text{ are totally} \\ \qquad \text{encompassed by the filter passband)} \\ 0 \qquad \text{(when either peak } k \text{ or } i \text{ is totally} \\ \qquad \text{ruled out of the filter passband)} \end{cases} \end{aligned}$$

2. Using the above results, the values of $\tilde{\beta}_{ki}$ under the assumptions **a**, **b**, **c** in Table 3.1 are easily derived. Since $\eta_{ki} = \mathbf{v}_k^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_i = b_{vk1}^*b_{vi1}\tilde{\beta}_{11} + b_{vk2}^*b_{vi2}\tilde{\beta}_{22} + b_{vk1}^*b_{vi2}\tilde{\beta}_{12}$ $+b_{vk2}^*b_{vi1}\tilde{\beta}_{21}$ from (3.5), using the values of $\tilde{\beta}_{ki}$, we have

a. When both peak 1 and peak 2 are selected, $\eta_{ki} = b_{vk1}^*b_{vi1}\beta_{11} + b_{vk2}^*b_{vi2}\beta_{22} + b_{vk1}^*b_{vi2}\beta_{12} + b_{vk2}^*b_{vi1}\beta_{21} = \mathbf{v}_k^H\mathbf{v}_i$. Due to the orthogonality and normalization properties of singular vectors, $\mathbf{v}$, it follows that $\eta_{kk} = 1$ and $\eta_{ki} = 0, (k \neq i)$.

b. When only peak 1 is selected, and peak 1 and 2 are two ideally nonoverlapping peaks, $\mathbf{v}_k = \mathbf{t}_k^*/\|\mathbf{t}_k\|$ from **ii**, i.e., $|b_{vkk}| = 1/\|\mathbf{t}_k\| = 1/\sqrt{\beta_{kk}}, b_{vki} = 0, (k \neq i)$. From the results of $\tilde{\beta}_{ki}$, we get $\eta_{ki} = b_{vk1}^*b_{vi1}\beta_{11}$, that is, $\eta_{11} = 1$ and $\eta_{22} = \eta_{12} = \eta_{21} = 0$.

c. When only peak 2 is selected, the proof is similar to that of **b**.

□

The theorem for the case of left multiplication is similar and omitted.

3.5 SNR relationships between the prefiltered and original data matrices

The previous section is based on the assumption of *pure* sinusoids without noise. In this section, we examine the SNR relationship between the prefiltered and original data matrices *when there is noise present* with a noise covariance matrix of $\mathbf{E}^H\mathbf{E} = \mu^2\mathbf{I} = L\nu^2\mathbf{I}$, as assumed in (2.4) of Chapter 2, Section 2.2.1.

In Monte-Carlo studies, we have been using the *peak* SNR, which is defined in Chapter 2, Section 2.3. It is computed from the peak amplitude, a_k, and the variance of the complex-valued white noise, $2\sigma_\nu^2$, as shown below.

$$\text{SNR} = 10\log_{10}\left(\frac{a_k^2}{2\sigma_\nu^2}\right) \quad \text{for peak } k$$

After prefiltering, however, the noise is not white anymore.

In this section, we use another definition of SNR [29], which is matrix oriented and denoted by SNR_m. Thus, the SNR_m for the data matrix $\mathbf{X} = \overset{\circ}{\mathbf{X}} + \mathbf{E}$ is

$$\text{SNR}_m = 10\log_{10}\frac{\|\overset{\circ}{\mathbf{X}}\|_F^2}{\|\mathbf{E}\|_F^2}\ (\text{dB}) = 10\log_{10}\frac{\sum_{k=1}^K \sigma_k^2}{M\mu^2}\ (\text{dB})$$

After prefiltering, we have $\tilde{\mathbf{X}} = \overset{\circ}{\tilde{\mathbf{X}}} + \tilde{\mathbf{E}} = \overset{\circ}{\mathbf{X}}\mathbf{H}_r + \mathbf{E}\mathbf{H}_r$ and

$$\widetilde{\text{SNR}}_m = 10\log_{10}\frac{\|\overset{\circ}{\tilde{\mathbf{X}}}\|_F^2}{\|\tilde{\mathbf{E}}\|_F^2}\ (\text{dB}) = 10\log_{10}\frac{\sum_{k=1}^K \tilde{\sigma}_k^2}{\|\mathbf{H}_r\|\mu^2}\ (\text{dB})$$

To simplify our study, we continue to use the assumptions of *idealized* cases of prefiltering in the previous section.

Comparison of $\|\mathbf{E}\|_F^2$ and $\|\tilde{\mathbf{E}}\|_F^2$:

In Theorem 2, it is shown that $\|\mathbf{H}\|_2 = \sigma_{H1} = 1$ for an ideal rectangular bandpass filter with a unit gain at passband. It follows that

$$\|\mathbf{H}_r\|_F^2 = \|\mathbf{H}\|_F^2 = \sum_{i=1}^M \sigma_{Hi}^2 \le M\sigma_{H1}^2 = M$$

which means that

$$\|\tilde{\mathbf{E}}\|_F^2 = \text{trace}(\tilde{\mathbf{E}}^H\tilde{\mathbf{E}}) = \text{trace}(\mathbf{H}_r^H\mathbf{H}_r)\mu^2 \le \|\mathbf{E}\|_F^2 = \text{trace}(\mathbf{E}^H\mathbf{E}) = M\mu^2$$

Actually, for FIR filter matrices defined in Section 3.2, we can find that

$$\|\mathbf{H}_{\text{FIR}t}\|_F^2 < \|\mathbf{H}_{\text{FIR}0}\|_F^2 = M\sum_{i=1}^q |h_i|^2 = \frac{M}{q}\sum_{m=1}^q |\mathcal{H}(m)|^2 = M\frac{f_u - f_l}{f_s}$$

Since $(f_u - f_l)/f_s < 1$ is always smaller than 1, this is a tighter bound for $\|\mathbf{H}\|_F^2$.

Right now, we have not found such a tighter bound for the IIR filter matrix defined in Section 3.2.

Comparison of $\|\overset{\circ}{\mathbf{X}}\|_F^2$ and $\|\overset{\circ}{\tilde{\mathbf{X}}}\|_F^2$:

Here, we consider the ideal cases of prefiltering when the signal is composed of only one and two sinusoids.

1. When there is only one sinusoid in the noiseless signal, it can be proven that $\tilde{\sigma}_1 = \sigma_1$ if this sinusoid is successfully selected. The proof is similar to that of Theorem 3 and thus omitted.

2. When there are two sinusoids in the noiseless signal, it can be found from Table 3.1 that

   a. $\tilde{\sigma}_1 + \tilde{\sigma}_2 = \sigma_1 + \sigma_2$, if both sinusoids are successfully selected;

   b. $\tilde{\sigma}_1 + \tilde{\sigma}_2 = \sigma_1 < \sigma_1 + \sigma_2$, if the first sinusoid is successfully selected and the second sinusoid is completely suppressed;

   c. $\tilde{\sigma}_1 + \tilde{\sigma}_2 = \sigma_2 < \sigma_1 + \sigma_2$, if the second sinusoid is successfully selected and the first sinusoid is completely suppressed.

Comparison of SNR_m and $\widetilde{\text{SNR}}_m$:

1. When there is only one sinusoid in the noiseless signal and this sinusoid is successfully selected, then the SNR_m is improved due to prefiltering since the SNR_m increment is

   $$\widetilde{\text{SNR}}_m - \text{SNR}_m = 10\log_{10}\frac{\tilde{\sigma}_1^2}{\|\mathbf{H}_r\|_F^2\mu^2}\cdot\frac{M\mu^2}{\sigma_1^2} = 10\log_{10}\frac{M}{\|\mathbf{H}_r\|_F^2}\ (dB)$$

   which is positive.

2. When there are two sinusoids in the noiseless signal, then

a. The SNR_m is improved due to prefiltering if both sinusoids are successfully selected since the SNR_m increment is

$$\widetilde{\mathrm{SNR}}_m - \mathrm{SNR}_m = 10\log_{10}\frac{\tilde{\sigma}_1^2+\tilde{\sigma}_2^2}{\|\mathbf{H}_r\|_F^2\mu^2}\cdot\frac{M\mu^2}{\sigma_1^2+\sigma_2^2} = 10\log_{10}\frac{M}{\|\mathbf{H}_r\|_F^2} \quad (dB)$$

which is positive.

b. The SNR_m is not always improved due to prefiltering if the first sinusoid is successfully selected and the second sinusoid is completely suppressed since the SNR_m increment is

$$\widetilde{\mathrm{SNR}}_m - \mathrm{SNR}_m = 10\log_{10}\frac{\tilde{\sigma}_1^2+\tilde{\sigma}_2^2}{\|\mathbf{H}_r\|_F^2\mu^2}\cdot\frac{M\mu^2}{\sigma_1^2} \quad (dB)$$

which is not always positive.

c. The SNR_m is not always improved due to prefiltering if the second sinusoid is successfully selected and the first sinusoid is completely suppressed since the SNR_m increment is

$$\widetilde{\mathrm{SNR}}_m - \mathrm{SNR}_m = 10\log_{10}\frac{\tilde{\sigma}_1^2+\tilde{\sigma}_2^2}{\|\mathbf{H}_r\|_F^2\mu^2}\cdot\frac{M\mu^2}{\sigma_2^2} \quad (dB)$$

which is not always positive.

In summary, when all the sinusoids are successfully selected, the SNR_m increment is

$$\widetilde{\mathrm{SNR}}_m - \mathrm{SNR}_m = 10\log_{10}\frac{M}{\|\mathbf{H}_r\|_F^2}\begin{cases} > 10\log_{10}\frac{f_s}{f_u-f_l} & \text{if } \mathbf{H}_r = \mathbf{H}_{\mathrm{FIR}t} \\ = 10\log_{10}\frac{f_s}{f_u-f_l} & \text{if } \mathbf{H}_r = \mathbf{H}_{\mathrm{FIR}0} \\ > 0 \text{ (at least)} & \text{if } \mathbf{H}_r = \mathbf{H}_{\mathrm{IIR}} \end{cases}$$

However, when only part of the sinusoids are selected, the total signal power is reduced, although the power of the selected sinusoids remains the same. Therefore, the SNR_m is not always increased.

3.6 HTLS_F prefiltering

Another prefiltering technique, denoted by HTLS_F, is based on the repetitive use of HTLS and is therefore called HTLS_F prefiltering. It is motivated by the technique proposed by Pijnappel et al. [32] for Kung's method. The idea is to first invoke HTLS to get a rough estimation of all K sinusoids comprising

the original signal and then subtract the signal that fits the unwanted peaks from the original one. As a result, the residual signal essentially consists of the wanted peaks and white noise only, which is actually a prefiltered signal. Finally, the wanted peaks in the selected region can be estimated by the HTLS method again. This frequency-selective technique is called HTLS_F-HTLS in this thesis and outlined below.

Algorithm 3

Outline of the HTLS_F-HTLS algorithm

Given. The model order estimates K and K_s of the original and residual signal respectively, and N uniformly sampled original data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel data matrix $\mathbf{X}$ as in (2.1).

Step 1. Apply the HTLS Algorithm 1 to $\mathbf{X}$ using model order K.

Step 2. Reconstruct an intermediate signal $\tilde{x}_n$ from the estimates of unwanted peaks.

Step 3. Arrange the prefiltered or residual data $\tilde{x}_n = x_n - \tilde{x}_n$ into an $L \times M$ Hankel matrix $\tilde{\mathbf{X}}$.

Step 4. Apply the HTLS Algorithm 1 to $\tilde{\mathbf{X}}$ using model order K_s.

The filtering effect of HTLS_F is different from that of a bandpass FIR or IIR filter due to the infinitely long tails of exponentially damped sinusoids in the spectrum. HTLS_F, in ideal cases, completely eliminates the unwanted peaks in the whole spectrum, without suppressing the tails of the wanted peaks. But a bandpass FIR or IIR filter not only keeps the tails of the unwanted peaks inside the passband but also reduces the tails of the wanted peaks outside the passband, especially if the wanted and unwanted peaks are strongly overlapping. However, when the unwanted peaks are also difficult to resolve so that the estimation of $\tilde{x}_n$ in Steps 1-2 of Algorithm 3 is not at all accurate, the final parameter estimation in Step 4 of Algorithm 3 based on the residual signal cannot be accurate either. In this case, a combination of the two prefiltering schemes called HTLS_F-$\mathbf{H}_{\mathrm{FIR}t}$-HTLS and HTLS_F-$\mathbf{H}_{\mathrm{IIR}}$-HTLS may help, where $\mathbf{H}_{\mathrm{FIR}t}$-HTLS and $\mathbf{H}_{\mathrm{IIR}}$-HTLS instead of HTLS is invoked in Step 4 of the HTLS_F-HTLS Algorithm 3 to further emphasize the wanted region and suppress the unwanted one.

Note that HTLS is not the only choice for prefiltering in Step 1. Besides HTLS_F prefiltering, we can use HSVD_F prefiltering, LP_F prefiltering, MP_F prefiltering, etc., prior to the final HTLS estimation. However, we only use HTLS_F prefiltering in this chapter.

3.7 Simulations

In this section, results of computer simulations are presented to confirm the theoretical predictions of earlier sections. Three different simulation examples are studied in various scenarios.

The algorithms considered here are the HTLS Algorithm 1 and FIR prefiltering, IIR prefiltering, HTLS_F prefiltering, followed by the HTLS Algorithm 1, outlined respectively in Sections 2.2.2, 3.3 and 3.6. Two types of FIR filters are tried out for different simulated signals: (1) a rectangular bandpass filter; (2) an exponential filter with frequency response $1/[1+4\pi^2(f-f_0)^2/\gamma^2]$, where f_0 is the center of the passband and γ is a constant controlling the bandwidth. Both filters use the Kaiser window with β parameter of 5.658 as the window function to truncate the impulse response [54]. These two types of FIR filter matrices are referred to by the subscript 1 and 2, respectively. For example, $\mathbf{H}_{\text{FIR}c1}$ and $\mathbf{H}_{\text{FIR}t1}$ are the circular and the FIR filter matrices generated from the first type of FIR filter, while $\mathbf{H}_{\text{FIR}c2}$ and $\mathbf{H}_{\text{FIR}t2}$ correspond to the second type. Similarly, $\mathbf{H}_{\text{IIR}}$ denotes the IIR filter matrix. In Example 1, the signal is real-valued and a 2nd-order IIR filter is used which is symmetric in positive and negative frequencies. Its z-transfer function is $\mathsf{H}(\mathsf{z}) = \xi(\mathsf{z}-1)(\mathsf{z}+1)/[(\mathsf{z}-\rho e^{j2\pi f_0})(\mathsf{z}-\rho e^{-j2\pi f_0})]$, where the poles are located at equal distance from the two sinusoids, $f_0 = (f_1+f_2)/2$. In Example 2, the signals are complex and we use the complex non-symmetric 1st-order IIR filters with transfer function $\mathsf{H}(\mathsf{z}) = \xi[\mathsf{z} - e^{j2\pi(f_0+f_s/2)}]/(\mathsf{z}-\rho e^{j2\pi f_0})$. ρ is a factor, fixed at 1.25 in our examples, controlling the bandwidth of the filter. ξ is a factor chosen such that the filter has a unit gain at the poles.

Example 1: *Signal-3*

In [51], a real-valued signal, called *Signal-3* in this thesis and described in Fig. A.3 and Table A.5, A.6 in Appendix A, has been simulated to study the effects of FIR prefiltering using a circular filter matrix $\mathbf{H}_{\text{FIR}c}$. Here, this signal is studied again using the prefiltered HTLS method and different filter matrices. Only the first type of FIR filters with a rectangular passband is considered here. Suppose both sinusoids are of interest and the frequency region is known. We choose the parameters for the HTLS and prefiltered HTLS algorithms as follows. $N = 30, M = q = 15, f_l = 0.0796$ Hz, $f_u = 0.191$ Hz. Since this signal is far from dying out at the 30th data point, a diagonal full-rank weighting matrix $\mathbf{W}$ is constructed from the Kaiser window with $\beta = 5.658$ to compensate the Gibbs phenomenon induced by the truncation of input data as discussed in section 3.3. When a filter matrix, say $\mathbf{H}$, is applied with or without weighting matrix $\mathbf{W}$, the method is called **H**-**W**-HTLS or **H**-HTLS, respectively.

Different prefiltering schemes using FIR filters and the HTLS method are compared as shown in Fig. 3.3. Although $\mathbf{H}_{\text{FIR}c1}$-HTLS lowers the failure rate of HTLS considerably, $\mathbf{H}_{\text{FIR}t1}$-HTLS performs better by eliminating the end effect. The reason is that the damping factors of the current signal are small and the 30-point signal has not died out sufficiently. This causes serious end effects affecting the accuracy of the $\mathbf{H}_{\text{FIR}c1}$-HTLS estimates, as mentioned in Section 3.2. The use of a weighting matrix $\mathbf{W}$ improves the resolution of the prefiltering method, $\mathbf{H}_{\text{FIR}c1}$-HTLS, because the signal dies out more quickly after weighting, and as a result, the end effect is reduced. The accuracy of all these algorithms is not much different, although the one using $\mathbf{H}_{\text{FIR}t1}$ seems a bit better. In conclusion, the application of FIR prefiltering using the filter matrix $\mathbf{H}_{\text{FIR}t}$ suppresses the noise outside the passband most effectively and enhances the resolution of all the peaks at comparable accuracy.

Observe that in Fig. 3.3(c), some accuracy curves at low SNR are closer to the CR bound curve than at high SNR. The reason is that the curves here are obtained by averaging over *good* runs, instead of all the 200 runs. Those bad runs or failures, which are eliminated, are typically the results of difficult signals with large noise. The exclusion of failures amounts to increasing the SNR of our simulated noisy signals, while CR bounds are calculated at the assumed SNR. This is where the inconsistency comes from. Actually, as we will see later on, this inconsistency becomes more serious when the failure rate gets higher, and the corresponding RMSE values may be even lower than the CR lower bounds at low SNR.

Example 2: *Signal-4*

This is a complex signal with three exponentially damped sinusoids as shown in Fig. A.4 and described in Tables A.7, A.8 in Appendix A. Suppose only the first peak is of interest as shown in Fig. 3.4, and the frequency, $f_1 = -0.2$ Hz, of this peak is known. When estimating other parameters of this peak, we can of course use the HTLS method to get parameters of all the peaks and pick out the interesting one. But this is a small peak severely corrupted

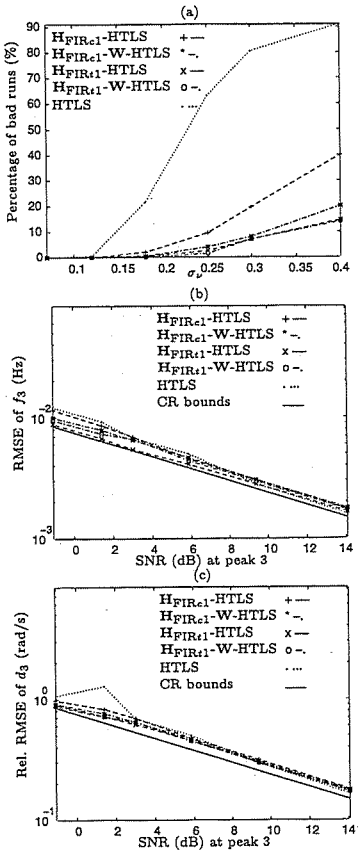


Figure 3.3: Example 1: (a) Failure rate versus σ_ν. (b) RMSE of $f_3 = 0.1$ Hz and (c) Relative RMSE of $d_3 = 0.02\pi$ rad/s versus SNR at peak 3.

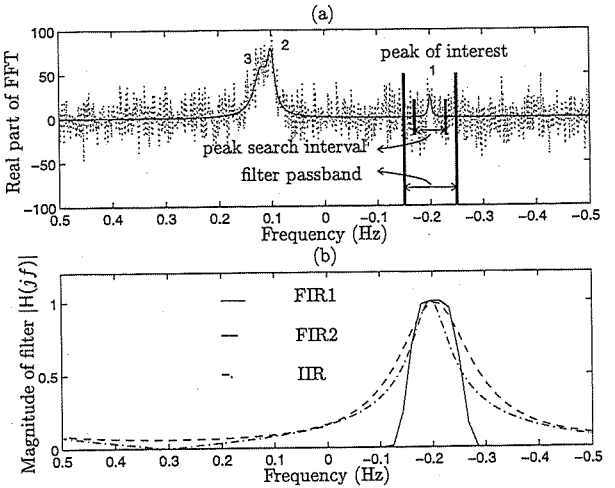


Figure 3.4: Example 2: (a) Noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.5$) spectra of *Signal-4*. (b) The frequency response of the filters used in the prefiltered HTLS method.

by noise and difficult to resolve by HTLS. Note also that this peak has a small damping factor and lies far away from the other two peaks, that is, it is approximately nonoverlapping with the other peaks. Hence, it is worthwhile to try our prefiltering technique in order to extract the interesting peak and enhance its resolution. Different prefiltering schemes are tried out using the parameters $N = 95, M = q = 57, f_l = -0.3$ Hz, $f_u = -0.1$ Hz, $\gamma = 0.5 f_s = 0.5$ Hz. Around the M=57th data point, the signal is dying out. $L = 39 < M$ is taken so that the total number of data points N, and hence the computational work, needed in the algorithm is small. The frequency interval in which the first peak should be resolved is $[f_1 - 0.03$ Hz, $f_1 + 0.03$ Hz $]$. If more than one estimated peak fall into this interval, the one with a frequency closest to f_1 is taken.

Fig. 3.5 compares the results of different methods. It is clear that the reso-

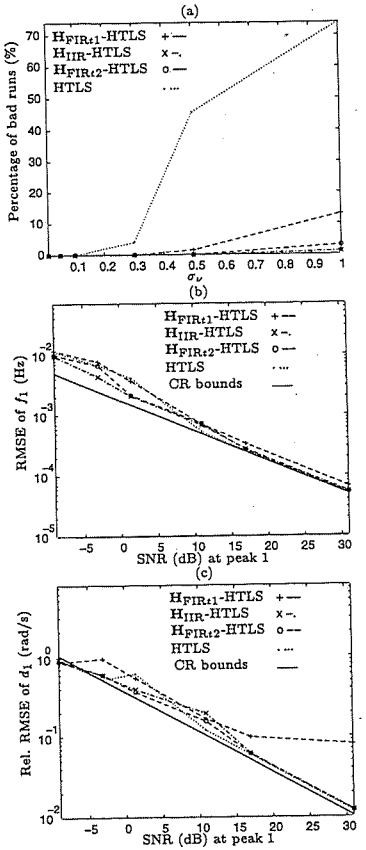


Figure 3.5: Example 2: (a) Failure rate versus σ_ν. (b) RMSE of f_1 and (c) Relative RMSE of d_1 versus SNR at peak 1.

lution of the selected peak 1 is greatly enhanced by all the prefiltering schemes used, especially at low SNR level. A more careful examination shows that $\mathbf{H}_{\mathrm{FIR}t1}$-HTLS is not as good as $\mathbf{H}_{\mathrm{FIR}t2}$-HTLS and $\mathbf{H}_{\mathrm{IIR}}$-HTLS in both resolution and parameter accuracy. In particular, the RMSE value of d_1 obtained by $\mathbf{H}_{\mathrm{FIR}t1}$-HTLS deviates from the CR bound at very high SNR. One explanation is that the FIR filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ introduces distortion due to the discontinuity at the initial conditions. Another possible explanation is that $\mathbf{H}_{\mathrm{FIR}t1}$ is constructed for a rectangular filter with abrupt changes around cut-off frequencies, while the spectrum of one exponentially damped sinusoid spreads smoothly along the frequency axis. The filtered sinusoid does not bear the original shape in the spectrum anymore and the signal characteristic is changed completely. The filter matrix $\mathbf{H}_{\mathrm{FIR}t2}$ corresponds to a bandpass filter with smooth spectrum centered at the peak of interest and hence preserves the signal spectrum better than $\mathbf{H}_{\mathrm{FIR}t1}$. Similarly, a high-order IIR filter with steep roll-off at cut-off frequencies performs poorly compared to a lower-order IIR filter (figures of these experiments are omitted). This artifact is more pronounced at high SNR. Therefore, when selecting only one sinusoid it is suggested to choose a bandpass filter that preserves the original signal characteristics as much as possible. Note that the phase characteristic of the filter is not important since the phase of the sinusoid is estimated from the original data as done in Section 2.2.2, instead of the filtered data.

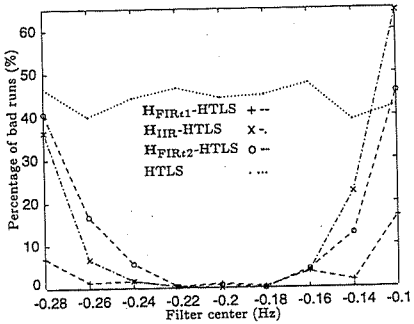


Figure 3.6: Example 2: Failure rate versus varying filter center. $\sigma_\nu = 0.5$.

In NMR spectroscopy, the frequency of a peak corresponds to a specific

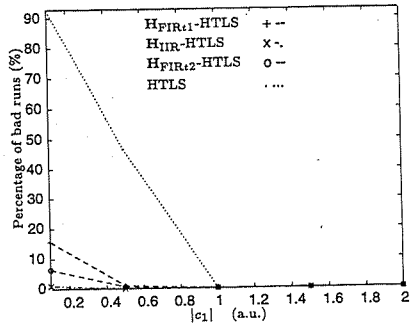


Figure 3.7: Example 2: Failure rate versus varying amplitude $|c_1|$ when other signal parameters are fixed. $\sigma_\nu = 0.5$.

metabolite and is therefore often known *a priori* from NMR tables and the estimation of damping factors and amplitudes is the main concern. In this case, the center frequency of the filter to be designed can be determined in advance. Alternatively the center frequency can also be estimated roughly from the FFT spectrum when the peak SNR is sufficiently high. Fig. 3.6 shows the sensitivity with respect to the chosen filter center. As expected, the prefiltered method based on a filter center slightly deviating from the exact center frequency of interest still performs better than the unfiltered method. The deviation from the exact filter center may be larger for the $H_{FIR\ell 1}$-HTLS method, which uses a filter with flat passband.

If we vary the absolute value of the complex amplitude, $|c_1|$, of the first sinusoid and keep the remaining parameters of this example unchanged, we can see that the smaller the $|c_1|$, the more we benefit from prefiltering, as shown in Fig. 3.7. In other words, prefiltering is more important when the peak of interest has a comparatively low SNR and thus needs to be emphasized.

Fig. 3.8 includes the results of HTLS$_F$ prefiltering also when the damping factor, d_1, of the first sinusoid is varied and the other parameters is fixed. The HTLS$_F$ prefiltering itself does not lower the failure rate of HTLS very much. It gives rather high failure rate at large d_1. This is because the HTLS$_F$ prefiltering is only effective when the unwanted peaks are easily resolved so that a sufficiently accurate first-step estimation can be obtained. When d_1 gets larger,

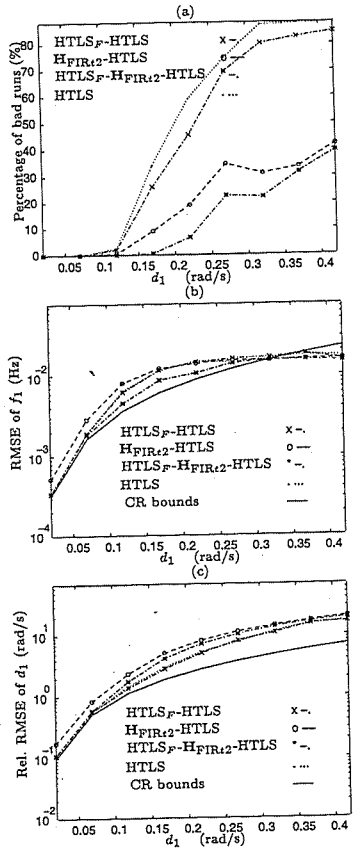


Figure 3.8: Example 2: (a) Failure rate, (b) RMSE of f_1 and (c) Relative RMSE of d_1 versus varying damping factor d_1 when other signal parameters are fixed. $\sigma_\nu = 0.5$.

the spectrum of the first peak becomes wider and wider, and hence more and more overlapping with the second and third peaks. In this case, results of a pre-estimation deteriorates. However, a further emphasis of the wanted peak can be made through an FIR or IIR bandpass filtering once the unwanted tail is removed by the HTLS$_F$ prefiltering. That is why the HTLS$_F$-$\mathbf{H}_{\text{FIRt2}}$-HTLS method gives the best resolution here when the unwanted peaks are not difficult to resolve.

Example 3: *Signal-5*

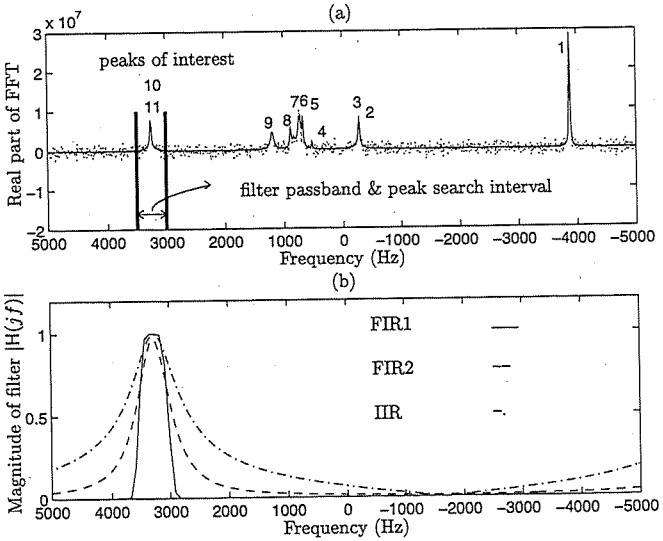


Figure 3.9: Example 3: (a) Noise-free (full line) and noisy (dotted line, $\sigma_\nu = 3 \times 10^4$) spectra of *Signal-5*. (b) The frequency response of the filters used in the prefiltered HTLS method.

Fig. 3.9 shows a complex simulated signal, called *Signal-5* and described in

Appendix A. It represents a typical signal encountered in NMR spectroscopy in which the glycogen peaks, 10 and 11 in this example, at the same known frequency are to be quantified [55]. The important but unknown parameters are the damping factor, which is the inverse of the relaxation time, and the amplitude representing the concentration of the metabolite, as explained in Chapter 1. Since only the glycogen peaks are of interest here and these peaks are not much overlapping with the remaining peaks, we apply several bandpass filters centered at the exact frequency of the glycogen peaks. The parameters for the prefiltered and unfiltered HTLS are: $N = 200, M = q = 121, f_l = 3000$ Hz, $f_u = 3500$ Hz, $\gamma = 0.2 f_s = 2000$ Hz. The region of wanted peaks, as well as the frequency interval in which the two glycogen peaks should be resolved, is defined as $[f_l, f_u]$. The estimated peak with a damping factor closest to d_{10} or d_{11} is classified as the estimate of peak 10 or 11.

An immediate observation of Fig. 3.10 is that the resolution of the prefiltered HTLS methods, $\mathbf{H}_{\text{FIRt2}}$-HTLS and HTLS$_F$-$\mathbf{H}_{\text{FIRt2}}$-HTLS, are much better than the HTLS and HTLS$_F$-HTLS methods. HTLS and HTLS$_F$-HTLS performs well only at very high SNR, e.g. at $\sigma_\nu = 5 \times 10^3$. They performs poorly at low SNR, e.g. at $\sigma_\nu = 3 \times 10^4$ and 5×10^4, since they can not resolve the two glycogen peaks in most of the noise trials while the other prefiltered HTLS methods are able to resolve these peaks in almost all trials. Although the HTLS$_F$ prefiltering does not lowers the failure rate of HTLS very much, it removes the unwanted peaks and their tails at the wanted region such that the HTLS$_F$ prefiltering followed by a bandpass prefiltering, like the HTLS$_F$-$\mathbf{H}_{\text{FIRt2}}$-HTLS in this figure, offers significant improvement. In parameter accuracy, $\mathbf{H}_{\text{FIRt2}}$-HTLS gives large RMSE values because the tails of the wanted peaks are suppressed by the stopband while the tails of the unwanted peaks are enlarged by the passband of the FIR filter. HTLS$_F$-$\mathbf{H}_{\text{FIRt2}}$-HTLS performs better due to the removal of unwanted peaks by HTLS$_F$ prefiltering. Note that the RMSE values of a_{10} and a_{11} at low SNR are lower than the CR bounds. This is due to the fact that prior knowledge of $f_{10} = f_{11} = 3250$ Hz has been used as the center frequency of the filter passband. However, the RMSE values of HTLS at low SNR are not reliable since they are averaged over the *good* runs only, excluding failures, and HTLS has very small numbers of good runs at low SNR.

Fig. 3.11 gives the performance of prefiltered HTLS methods versus the choice of M/N. The range of the optimal values of M/N is at least as large as that of the unfiltered HTLS, which is between 1/3 and 2/3 typically.

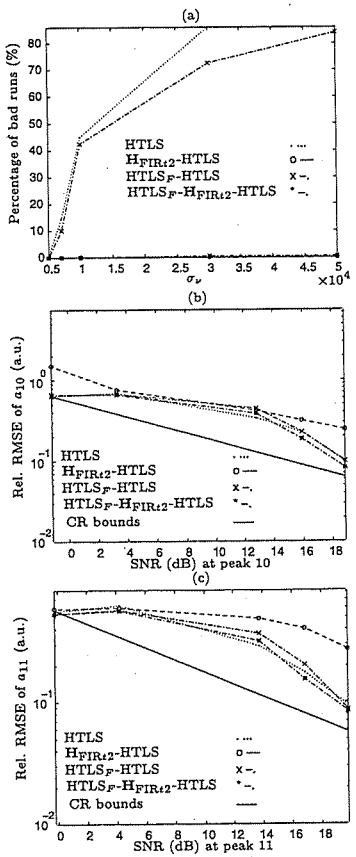


Figure 3.10: Example 3: (a) Failure rate versus σ_ν. Relative RMSE of (b) a_{10} and (c) a_{11} versus SNR at peak 10 and 11, respectively.

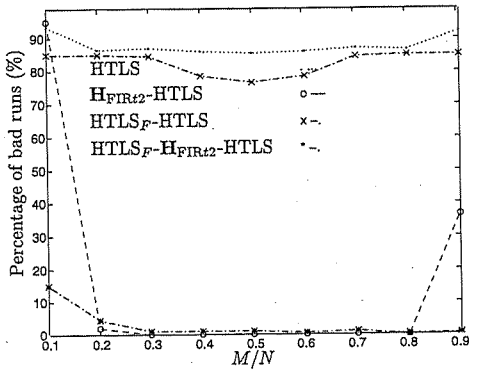


Figure 3.11: Example 3: Failure rate versus M/N. $\sigma_\nu = 3 \times 10^4$.

3.8 Conclusions

In estimating parameters of a sum of K exponentially damped sinusoids, two types of prefiltering techniques have been presented for the HTLS method. These techniques can be applied in two ways. First, when all the K sinusoids are of interest and the SNR's are low, a filter can be designed to encompass all the sinusoids and reduce the noise outside the passband. Secondly, when only some of the sinusoids (fewer than K) are of interest, a filter can be designed to encompass only the wanted peaks and suppress the unwanted ones and the noise.

The first type of prefiltering is realized by constructing a full-rank square filter matrix $\mathbf{H}_{FIRt}$ or $\mathbf{H}_{IIR}$ from an FIR or IIR filter, and then multiplying it with the original Hankel data matrix before applying the HTLS estimation method. One of the characteristics of this prefiltering technique is that it does not increase the rank of the Hankel data matrix in the absence of noise. As a result, the computational work in the following HTLS estimation, as well as the variance of estimates, are not increased either. Appropriate application of the prefiltering technique not only reduces the noise but also suppresses the unwanted peaks outside the filter passband as demonstrated by means of a theoretical analysis.

The second type of prefiltering is the HTLS$_F$ prefiltering. It first invokes HTLS to get a pre-estimation of all K sinusoids comprising the original signal, then subtracts the signal that fits the unwanted peaks from the original one, and finally estimates the parameters of wanted peaks from the residual signal. It is usually applied when fewer than K sinusoids are of interest and the unwanted sinusoids are easily resolvable.

The benefits of the proposed filter matrices, $\mathbf{H}_{\text{FIR}t}$ and $\mathbf{H}_{\text{IIR}}$, are confirmed through simulations. When an FIR or IIR filter is appropriately designed, the resulting resolution is enhanced at comparable parameter accuracy, especially for low SNR. The parameter accuracy can be improved when the FIR or IIR filtering is preceded by the HTLS$_F$ prefiltering, that is, through the combination of the two types of prefiltering techniques.

It should be noted that the FIR and IIR prefiltering technique can be used prior to other SVD and subspace-based methods as well, such as the LP method, the MP method, Kung et al.'s method, etc.

Nevertheless, there still remain open problems with the proposed FIR and IIR prefiltering techniques. Recall that in Section 2.2 of Chapter 2, there is a requirement for the HTLS estimator that the noise covariance matrix be a scalar multiple of an identity matrix, or the noise be asymptotically white. However, when the noisy data matrix $\mathbf{X}$ is prefiltered, say, via $\bar{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$, the filtered noise covariance matrix is generally not a scalar multiple of identity matrix anymore. This can be seen from

$$\bar{\mathbf{X}}^H\bar{\mathbf{X}} = \mathbf{H}_r^H \overset{\circ}{\mathbf{X}}{}^H \overset{\circ}{\mathbf{X}}\, \mathbf{H}_r + \mathbf{H}_r^H\mathbf{E}^H\mathbf{E}\mathbf{H}_r = \mathbf{H}_r^H \overset{\circ}{\mathbf{X}}{}^H \overset{\circ}{\mathbf{X}}\, \mathbf{H}_r + \mu^2\mathbf{H}_r^H\mathbf{H}_r$$

This problem cannot be solved by a whitening operation when the filter matrix $\mathbf{H}_r$ is *square*, which is the choice in this chapter. For example, the following whitening operation after a prefiltering operation using a square filter matrix $\mathbf{H}_r$ returns back to the HTLS algorithm, since

$$\mathbf{H}_r = \mathbf{Q}\mathbf{R}, \quad \bar{\mathbf{X}} = \mathbf{X}\mathbf{H}_r\mathbf{R}^{-1} = \mathbf{X}\mathbf{Q} = \mathbf{U}\boldsymbol{\Sigma}\bar{\mathbf{V}}$$

where $\mathbf{Q}$ is unitary and the left singular vectors of $\bar{\mathbf{X}}$ are exactly the same as those of $\mathbf{X}$.

Once the above problem is alleviated, e.g., as is the case of the HTLS(PK) method that incorporates prior knowledge of known signal poles (see Chapter 7, p. 145), there will be further improvement in resolution and parameter accuracy.

Chapter 4

Minimum variance signal enhancement

In practical signal processing, there are inevitably imperfections in the measurement data. However, parameter estimates of high resolution and accuracy are in demand. Therefore, it is often desired to apply a "cleaning" process, or what we call signal enhancement, to remove noise, measurement distortion, or even theoretical mismatch present in the given data set, before extracting parameters. A variety of signal enhancement algorithms have been developed for such purposes. These enhancement algorithms can be classified with respect to our objectives in mind:

- The enhanced signal itself must be "cleaned up" and smoothed.
- The estimated parameters of the enhanced signal must be closer to the true parameters of noise-free signal.

Though these two aspects are closely related, the latter is the emphasis of this study. In the current and succeeding chapters, we investigate, respectively, suboptimal and optimal signal enhancement techniques with a view to improving resolution and parameter accuracy, which is the main concern throughout this thesis.

This chapter is devoted to subspace-based signal enhancement methods in which the SVD plays a key role. These enhancement algorithms, though suboptimal, deliver good approximation of the original data matrix and good parameter estimates via the use of SVD. The success of the SVD in subspace methods can be justified from its special characteristics, such as the following three frequently cited features:

1. The information on the underlying data model is completely contained in certain subspaces of the signal space, which can be determined from the SVD.
2. The complexity of the model is given by the (approximate) rank of the data matrix, which can be estimated from the singular values.
3. In applications where the data are corrupted by additive noise, it is expected that the SVD has a certain noise filtering effect (also taking into account the relative insensitivity of the SVD with respect to perturbations).

Of course, the success of the SVD also depends on how it is applied in the algorithms. It is the main purpose of this chapter to clarify when and which SVD-based enhancement algorithm is most successful.

4.1 Introduction

Conceptually, a signal enhancement is accomplished through the process of slightly modifying the measured signal so that the modified signal possesses certain attributes or properties. The modification of the given data set serves as a preprocessing step of "cleaning up" the data before estimating the values of the signal parameters. Intuitively, this enhanced signal would provide a better representation of the underlying signal than the original measured signal.

In [56][57][58], Cadzow proposed an enhancement algorithm for the measured data represented in the form of a data matrix, such as the Hankel matrix $\mathbf{X}$ in (2.1). Frequently, the underlying data matrix is known to possess certain theoretical properties such as having a Hankel structure, a fixed rank, or a prescribed singular value characterization. The measurement process, however, usually results in a matrix which fails to possess all (or any) of these properties. Cadzow's method achieves the matrix property matching of Hankel structure and fixed rank via the use of the SVD. The application of Cadzow's enhancement procedure up to convergence followed by an SVD-based linear prediction algorithm has been performed on NMR spectroscopy data in [59][60]. This method works well for estimating nonoverlapping peaks. However, its high computational cost brings into question its practical use in NMR data analysis, and its resolution performance needs further improvement as well.

In the rest of this chapter, an improved signal-enhancement algorithm for exponential data quantification, presented in [61] and applied in [62] is considered and compared with Cadzow's algorithm. As outlined in Section 4.2, this approach is based on the minimum variance (MV) estimation method [29][61],

which first arranges the data in a very rectangular (instead of a square) Hankel structured matrix in order to make the corresponding signal-only data matrix orthogonal to the noise, then computes an estimate of the signal-only data matrix, and finally restores the Hankel structure of the computed estimate. Similar to Cadzow's method, little prior information (except model order) is required and the proposed method can be applied as a preprocessing procedure to a large class of subspace-based signal-estimation methods. Section 4.3 discusses the computational efficiency, choice of matrix sizes, etc., of the MV and Cadzow's enhancement. Simulation results are given in Section 4.4 to demonstrate a significant improvement in resolution performance at a comparable parameter accuracy and a low computational cost. Furthermore, the convergence of the MV and Cadzow's enhancement procedures is found to be unnecessary when followed by an SVD-based estimator such as HTLS, since the accuracy of the estimated parameters deteriorates despite a smoother spectrum, i.e., a "cleaner" spectrum does not guarantee better parameter accuracy. Finally, the proposed signal-enhancement algorithms are applied to the quantitative time-domain analysis of NMR measurement data in Section 3.5.

4.2 Enhancement algorithms

Although the enhancement algorithms discussed in this chapter also start from a Hankel matrix comprising the same data set as in subspace-based parameter estimation algorithms like HTLS, the size of the Hankel data matrix may differ. To distinguish the variables of preprocessing algorithms from that of parameter estimation algorithms, we use a caret, $\hat{}$, above the variables in the preprocessing algorithms. That is, the Hankel data matrix $\hat{\mathbf{X}}$ of dimension $\hat{L} \times \hat{M}$ can be generated by two unknown matrices: an exact data matrix $\hat{\mathring{\mathbf{X}}}$ and an additive perturbation matrix $\hat{\mathbf{E}}$, so that

$$\hat{\mathbf{X}} = \hat{\mathring{\mathbf{X}}} + \hat{\mathbf{E}}$$

Algorithm 4

Outline of the SVD-based data enhancement procedure

Given. *A model-order estimate K, N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, and tolerance ϵ.*

Step 1: Initialization. $\hat{\mathbf{x}} = [x_0, x_1, \ldots, x_{N-1}]$.

Step 2: Iteration. Repeat

(a) *Form an* $\hat{L} \times \hat{M}$ *Hankel matrix* $\hat{\mathbf{X}}$ *from the data vector* $\hat{\mathbf{x}}$ *with* $\hat{L} \geq \hat{M} > K$ *and* $N = \hat{L} + \hat{M} - 1$.

(b) *Compute the SVD of* $\hat{\mathbf{X}} = \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^H$, *where* $\hat{\mathbf{\Sigma}} \in \mathbb{C}^{\hat{L}\times\hat{M}}$ *contains the singular values* $\hat{\sigma}_1, \ldots, \hat{\sigma}_{\hat{M}}$ *with* $\hat{\sigma}_1 \geq \ldots \geq \hat{\sigma}_{\hat{M}}$.

(c) *Correct the singular values by applying a correction function, or rank-shaping function,* f_{corr} *that truncates to rank* K:

$$\hat{\mathbf{X}}_K = \hat{\mathbf{U}}_K f_{corr}(\hat{\mathbf{\Sigma}}_K)\hat{\mathbf{V}}_K^H$$

where $\hat{\mathbf{U}}_K \in \mathbb{C}^{\hat{L}\times K}, \hat{\mathbf{V}}_K \in \mathbb{C}^{\hat{M}\times K}$ *are respectively the first* K *columns of* $\hat{\mathbf{U}} \in \mathbb{C}^{\hat{L}\times\hat{L}}, \hat{\mathbf{V}} \in \mathbb{C}^{\hat{M}\times\hat{M}}$ *and* $\hat{\mathbf{\Sigma}}_K = diag(\hat{\sigma}_1, \cdots, \hat{\sigma}_K)$.

(d) *Set* $\hat{\mathbf{x}}_{\text{old}} = \hat{\mathbf{x}}$. *Compute the "cleaned-up" data vector* $\hat{\mathbf{x}}$ *from the rank-reduced data matrix* $\hat{\mathbf{X}}_K$ *by arithmetic averaging along its anti-diagonals.*

Until

$$\|\hat{\sigma}_{K+1}\| \leq \epsilon \tag{4.1}$$

$$or \quad \|\hat{\mathbf{x}}_{\text{old}} - \hat{\mathbf{x}}\| \leq \epsilon \tag{4.2}$$

After data preprocessing or signal enhancement outlined above, the signal parameters are extracted from the final "cleaned-up" data by means of an appropriate subspace-based signal-estimation algorithm. For example, the $4K$ signal parameters f_k, d_k, a_k and $\varphi_k, k = 1, 2, \ldots, K$ in the exponential data model (1.4) can be estimated using the LP-based method, the MP method, Kung et. al's method, or preferably, the HTLS method, Algorithm 1, outlined in Section 2.2.2 of Chapter 2. The procedure is then simply based on replacing the original noisy data $x_0, x_1, \ldots, x_{N-1}$ used in the $L \times M$ Hankel data matrix $\mathbf{X}$ by the "cleaned-up" data $\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{N-1}$.

In the following subsections, we describe two special forms of data preprocessing, the Cadzow and MV algorithms. The two algorithms differ only in the correction function f_{corr} in Step 2(c).

4.2.1 Cadzow's algorithm

Basically, Cadzow's algorithm modifies a given data set by exploiting two important properties of the data matrix $\hat{\mathbf{X}}$:

1. property of rank K
2. property of Hankel structure

The above preprocessing procedure reduces to *Cadzow's estimation method* for the following choice of f_{corr}

$$\boxed{f_{\text{corr}}(\hat{\mathbf{\Sigma}}_K) = \hat{\mathbf{\Sigma}}_K \quad \text{and} \quad \hat{L} = L,\ \hat{M} = M}$$

i.e. the singular values are not corrected and the Hankel matrix used in the preprocessing procedure has the *same* dimensions as the Hankel matrix used in the chosen signal-parameter-estimation method. For example, if HTLS is used, a square Hankel matrix $\hat{\mathbf{X}}_{\hat{L}\times\hat{M}}$ with $\hat{L} = \hat{M}(+1)$ is considered in the preprocessing procedure. Actually, the first two steps of a Cadzow's iteration are simply the LS estimation procedure given in the subsection 2.2.1.

4.2.2 Minimum variance algorithm

The MV estimation algorithm differs from the Cadzow algorithm in basically two aspects

- The MV algorithm starts from a very rectangular matrix, of which the number of rows is much larger than the number of columns, instead of a square matrix.
- The MV algorithm makes further use of the singular value property of a data matrix in addition to the property of rank K and the Hankel structure.

In terms of the above data preprocessing algorithm, the *MV estimation method* is based on the following choice of f_{corr} [29][61]

$$\boxed{\hat{L} >> \hat{M} > K \quad \text{and} \quad f_{\text{corr}}(\hat{\mathbf{\Sigma}}_K) = (\hat{\mathbf{\Sigma}}_K^2 - \hat{L}\nu_{\text{est}}^2\mathbf{I})\hat{\mathbf{\Sigma}}_K^{-1} \ \text{ with } \ \hat{\sigma}_K^2 > \hat{L}\nu_{\text{est}}^2} \tag{4.3}$$

where ν_{est}^2 is an estimate of the element-wise variance ν^2, which is equal to σ_ν^2 for real noise, or $2\sigma_\nu^2$ for complex noise (recall that in Section 2.3 of Chapter 2, σ_ν is defined as the noise standard deviation of real and complex part, respectively).

An MV iteration is based on the minimum variance estimation of $\overset{\circ}{\hat{\mathbf{X}}}$, or the orthogonal projection of $\overset{\circ}{\hat{\mathbf{X}}}$ onto the column space of $\hat{\mathbf{X}}$, represented as follows:

$$\min_{\hat{\mathbf{Y}}\in\mathbb{C}^{M\times M}} \|\hat{\mathbf{X}}\hat{\mathbf{Y}} - \overset{\circ}{\hat{\mathbf{X}}}\|_F^2$$

Setting to zero the derivatives of the object function

$$\|\hat{\mathbf{X}}\hat{\mathbf{Y}} - \hat{\overset{\circ}{\mathbf{X}}}\|_2^2 = \text{trace}(\hat{\mathbf{Y}}^H\hat{\mathbf{X}}^H\hat{\mathbf{X}}\hat{\mathbf{Y}} + \hat{\overset{\circ}{\mathbf{X}}}^H\hat{\overset{\circ}{\mathbf{X}}} - 2\hat{\mathbf{Y}}^H\hat{\mathbf{X}}^H\hat{\overset{\circ}{\mathbf{X}}})$$

with respect to the elements of $\hat{\mathbf{Y}}$ results in

$$\hat{\mathbf{Y}} = (\hat{\mathbf{X}}^H\hat{\mathbf{X}})^{-1}\hat{\mathbf{X}}^H\hat{\overset{\circ}{\mathbf{X}}}$$

so that the MV estimate of $\hat{\overset{\circ}{\mathbf{X}}}$ is given by

$$\begin{aligned}\hat{\mathbf{X}}_{\text{MV}} &= \hat{\mathbf{X}}\hat{\mathbf{Y}} \\ &= \hat{\mathbf{X}}(\hat{\mathbf{X}}^H\hat{\mathbf{X}})^{-1}\hat{\mathbf{X}}^H\hat{\overset{\circ}{\mathbf{X}}} \\ &= \hat{\mathbf{U}}_K(\hat{\Sigma}_K^2 - \mu^2\mathbf{I}_K)\hat{\Sigma}_K^{-1}\hat{\mathbf{V}}_K^H\end{aligned}$$

The last equality uses the SVD expression of $\hat{\overset{\circ}{\mathbf{X}}}$ and $\mathbf{X}$ in (2.2) and (2.5), as well as the canonical angles expression in (2.6), which are derived in Section 2.2.1 under three assumptions C1, C2, C3 and $\hat{\mathbf{E}}^H\hat{\mathbf{E}} = \mu^2\mathbf{I} = \hat{L}\nu^2\mathbf{I}$. In practice, these algebraic and geometrical assumptions may not be satisfied. But as the number of rows $\hat{L} \to \infty$, *asymptotically*, these conditions are satisfied [29] under statistical assumptions which are commonly observed in practice: the noise elements of $\hat{\mathbf{E}}$ have zero mean, bounded fourth order moments and are independently and identically distributed with possibly unknown variance ν^2. This is why $\hat{L} >> \hat{M}$ and white noise (such that $\hat{\mathbf{E}}^H\hat{\mathbf{E}} = \hat{L}\nu^2\mathbf{I}$) is required in the MV algorithm.

It should be noted that ν_{est}^2 is the estimated noise variance and can be known or computed either from the last data points provided they are pure noise (tail of the signal) or from the noise singular values, i.e. $\nu_{\text{est}}^2 = \frac{1}{(\hat{M}-K)\hat{L}}\sum_{i=K+1}^{\hat{M}}\hat{\sigma}_i^2$. Note that ν_{est} changes when we iterate the method. Hence, in the higher iterations ν_{est} is no longer known but can only be computed as explained above.

4.2.3 Composite property mapping

A detailed proof of the convergence of Cadzow's and MV enhancement algorithms is given in [56] and [61] in terms of composite property mapping. In essence, both algorithms make use of signal properties that identify the underlying useful information in the data set. The original problem is decomposed into several simpler subproblems related to each individual signal property, such as: $\hat{\mathbf{X}}$ has Hankel structure, $\hat{\mathbf{X}}$ has rank K, etc.. The basic requirement of this

simplified approach entails the ability to generate each property solution set for an arbitrary signal matrix, or each mapping operation that maps the signal matrix into the set of optimum signal approximations. Since the signal mappings are each closed in Cadzow's and MV algorithms, the convergence of such composite property mapping, implemented in the form of iterative processing, is assured.

4.3 Computational complexity

Based on the number of operations involved (see e.g., ref. [6] p. 239), it is easy to see that the computationally most time-consuming part of the algorithm, namely the SVD and the computation of the truncated matrix $\hat{\mathbf{X}}_K$, requires significantly less work whenever $\hat{L} >> \hat{M}$ [61]. Define by one flop the amount of work associated with one floating point operation (one addition, multiplication, division,...). If only third order terms are considered the number of flops involved in preprocessing a real $\hat{L} \times \hat{M}$ rank K matrix $\hat{\mathbf{X}}$ of the form (2.1) is given by:

$$\begin{array}{ll} & 6\hat{L}\hat{M}^2 + 20\hat{M}^3 + 2\hat{M}K\hat{L} \quad \text{if} \quad \hat{L} \geq 5\hat{M}/3 \\ & 14\hat{L}\hat{M}^2 + 8\hat{M}^3 + 2\hat{M}K\hat{L} \quad \text{if} \quad \hat{L} < 5\hat{M}/3 \\ \Rightarrow & 22\hat{M}^3 + 2\hat{M}^2K \quad \text{if} \quad \hat{L} = \hat{M} \end{array}$$

Denote by $\hat{L}_{\text{CA}}$ ($\hat{M}_{\text{CA}}$) the number of rows (columns) as used in Cadzow's method, and by $\hat{L}_{\text{MV}}$ ($\hat{M}_{\text{MV}}$) the number of rows (columns) as used in the MV estimation method. We must have: $N = \hat{M}_{\text{CA}} + \hat{L}_{\text{CA}} - 1 = \hat{M}_{\text{MV}} + \hat{L}_{\text{MV}} - 1$, i.e. the number of data points is the same. If $\mathbf{X}$ in the HTLS estimation algorithm is approximately square ($L \approx M$) then also $\hat{L}_{\text{CA}} \approx \hat{M}_{\text{CA}}$. This means that an MV iteration requires roughly only

$$\frac{3\hat{L}_{\text{MV}}\hat{M}_{\text{MV}}^2 + 10\hat{M}_{\text{MV}}^3 + \hat{M}_{\text{MV}}K\hat{L}_{\text{MV}}}{7\hat{L}_{\text{CA}}\hat{M}_{\text{CA}}^2 + 4\hat{M}_{\text{CA}}^3 + \hat{M}_{\text{CA}}K\hat{L}_{\text{CA}}}100\% \tag{4.4}$$

of the number of flops performed by one Cadzow iteration in the preprocessing procedure. Since only the highest order terms are considered the expressions above only hold for large $\hat{L}$, $\hat{M}$. For complex matrices the same factor can be used (depending on the algorithm used for the complex SVD this ratio can alter a little bit). Although flop counting is a crude approach to measuring an algorithm's efficiency [6] it gives us nevertheless a good idea of the possible gain in computational efficiency.

For example, assume that $N = 25$ and $K = 2$ (as in [63, Example 2] or the simulated signal called *Signal-1* in Appendix A in this thesis). Choose

$\hat{M}_{\text{CA}} = \hat{L}_{\text{CA}} = 13$, $\hat{L}_{\text{MV}} = 22$ and $\hat{M}_{\text{MV}} = 4$. Then (4.4) implies that an MV iteration requires 8% of the computation time needed for a Cadzow iteration. In practical computations using Matlab, we obtained a ratio of 12.8% for real matrices and a ratio of 14.9% for complex matrices. Of course, since $N, \hat{M}, \hat{L}$ are small, (4.4) is too crude. If $N = 128$ and $K = 5$, as is the case of the simulated signal, called *Signal-2*, which will be studied extensively further on, then the work involved in preprocessing a 119×10 data matrix $\hat{\mathbf{X}}$ by each MV iteration should, according to (4.4), only be 2% of the work involved in processing a 65×64 matrix as done in a Cadzow's iteration. With Matlab, ratios of 3.6% and 4.4% were respectively obtained for real and complex matrices.

A similar saving can be obtained with the subsequent HTLS algorithm, provided that the preprocessing procedure has been applied up to convergence. This is due to the fact that upon convergence, the preprocessing procedure described above has reduced a full-rank Hankel matrix to one of exactly rank K. In this situation, we can simplify the subsequent parameter estimation as shown below.

- If the number of samples, N, is excessively large, (i.e. $N \gg 2K$), the number of data samples of the enhanced signal used in the pole-estimation algorithm can be reduced from N to as small as $2K-1$ without affecting the resulting estimates. Hence the Hankel data matrix can be reduced in size as much as possible to increase efficiency.

- Any SVD-based parameter estimation method using any matrix size $L \times M$ $(L, M \geq K)$ produces the same estimates after the enhancement. In fact, no SVD is required and the signal poles are simply the eigenvalues of the solution $\check{\mathbf{Z}}$ in $\mathbf{X}^{\uparrow} = \mathbf{X}_{\downarrow}\check{\mathbf{Z}}$. However, if the enhancement is not fully convergent, the most efficient and accurate estimator, HTLS, can be taken.

Of course, this saving is only possible when the preceding enhancement procedure was performed up to convergence. Such preprocessing may take hundreds of iterations depending on the convergence criteria used, which is defined in (4.1) and (4.2).

Another remark is that $\hat{M}$ should not be too small in the MV algorithm in order to guarantee a sufficiently large gap between the smallest signal singular value $\hat{\sigma}_K$ and the largest noise singular value $\hat{\sigma}_{K+1}$ of the $\hat{L} \times \hat{M}$ Hankel matrix $\hat{\mathbf{X}}$. As $\hat{M}$ decreases (i.e. $\hat{L}$ increases since $N = \hat{L} + \hat{M} - 1$ is constant and limited), the singular values of $\hat{\mathbf{X}}$ also decrease. The smallest signal singular value $\hat{\sigma}_K$ decreases quite strongly if $\hat{M} \approx K$, since $\hat{\mathbf{X}}$ contains fewer elements. This implies that the ratio $\hat{\sigma}_K / \hat{\sigma}_{K+1}$ decreases because the gap between the

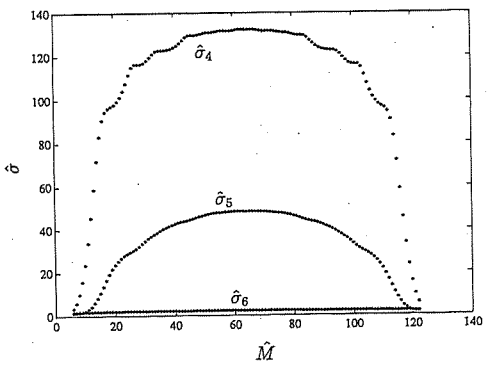


Figure 4.1: The singular values $\hat{\sigma}_{K-1}, \hat{\sigma}_K$ and $\hat{\sigma}_{K+1}$ (i.e., $\hat{\sigma}_4, \hat{\sigma}_5$ and $\hat{\sigma}_6$) of the $(129 - \hat{M}) \times \hat{M}$ Hankel matrix built up with 128 data points of a noisy $(\sigma_\nu = 0.1)$ simulated ^{31}P NMR signal, *Signal-2*, with model order $K = 5$.

smallest signal singular value $\hat{\sigma}_K$ and the largest noise singular value $\hat{\sigma}_{K+1}$ narrows. This can be observed in Fig. 4.1. The (fast) decrease of the smallest signal singular value of $\hat{\mathbf{X}}$ may have a deteriorating effect on the accuracy of the estimated signal subspace and subsequently affect the accuracy of the signal-parameter estimates. Referring to (2.6) and [29][61], a smaller gap means that the canonical angles between the exact signal subspace and the computed one enlarge, thereby increasing the bias of the estimates, especially those of the most difficult peak (smallest in amplitude, overlapping, ...), as can be seen in the simulation study given further on. On the other hand, $\hat{M}$ should not be large since then the signal-only data cannot be orthogonal to the noise, implying that the MV estimation properties no longer hold. For that reason, we choose $\hat{M}$ between $K + 5$ and $2K$.

Finally, the condition that the noise should be white is not always a restriction. If the noise is not white then $\hat{\mathbf{E}}^H\hat{\mathbf{E}} = \mu^2\mathbf{I}$ is no longer valid so that the singular value correction function is no longer correct. However if $\hat{\mathbf{E}}^H\hat{\mathbf{E}}$ is known or can be estimated up to an unknown scalar, then we can factor $\hat{\mathbf{E}}^H\hat{\mathbf{E}} = \hat{\mathbf{R}}^H\hat{\mathbf{R}}$ (or $\hat{\mathbf{E}} = \hat{\mathbf{Q}}\hat{\mathbf{R}}$, $\hat{\mathbf{Q}}^H\hat{\mathbf{Q}} = \mathbf{I}$ if $\hat{\mathbf{E}}$ is directly available) where

$\hat{\mathbf{R}} \in \mathbb{C}^{\hat{M}\times\hat{M}}$ is a nonsingular square root. In this case, we consider the matrix

$$\widetilde{\mathbf{X}} = \hat{\mathbf{X}}\hat{\mathbf{R}}^{-1} = \hat{\overset{\circ}{\mathbf{X}}}\hat{\mathbf{R}}^{-1} + \hat{\mathbf{E}}\hat{\mathbf{R}}^{-1}$$

Observe that $(\hat{\mathbf{E}}\hat{\mathbf{R}}^{-1})^H(\hat{\mathbf{E}}\hat{\mathbf{R}}^{-1})$ is again a multiple of the identity matrix so that we can again apply the same procedure, i.e. compute the SVD of $\widetilde{\mathbf{X}}$, truncate to rank K and correct the singular values (according to (4.3) in order to obtain $\widetilde{\mathbf{X}}_K$. The only modification is that $\widetilde{\mathbf{X}}_K$ should be post-multiplied with $\hat{\mathbf{R}}$, i.e. $\hat{\mathbf{X}}_K = \widetilde{\mathbf{X}}_K\hat{\mathbf{R}}$, before restoring the Hankel structure.

The fact that $\hat{\mathbf{E}}^H\hat{\mathbf{E}}$ or an estimate should be known up to a scalar is not always a trivial requirement. In some cases, this matrix can be estimated during the absence of the signal. For instance, when modeling damped exponential signals as is the case in NMR spectroscopy, $\hat{\mathbf{E}}$ can be computed from the tail of the signal where the exact signal has died out.

4.4 Simulation results

In this section, we illustrate, via a computer simulation study, the differences in resolution enhancement that may occur from the use of the different signal-enhancement procedures when the complex data are fitted to *Signal-2*, described in Fig. A.2 and Tables A.3, A.4 in Appendix A. This signal, modeled by a fifth order exponential function defined by Eq. (1.4), represents a ^{31}P NMR signal from a perfused rat liver.

The signal-parameter-estimation algorithm considered here is HTLS as outlined in the previous section, where the Hankel matrix is as square as possible, i.e. $L \times M = 64 \times 65$. The number of data points used in each fit is $N = 128$ (corresponding to the signal disappearing in the noise in the time domain), the matrix sizes used in the MV method and Cadzow's method are, respectively, 119×10 and 64×65, and the estimated model order is $K = 5$. If no preprocessing is performed, the method is called HTLS. If the MV method, Cadzow's method, or MV followed by Cadzow's method is used as signal-enhancement algorithm prior to HTLS, the method is abbreviated in this Chapter, respectively, as nMV-HTLS, nCA-HTLS or MV-nCA-HTLS, where n indicates the number of iterations performed; n is deleted if only one iteration is performed and n is replaced by c if the method has been used up to convergence. The convergence criterion used is (4.1) where the tolerance is set to $\epsilon = 10^{-6}$. The maximum number of iterations is set to 100.

Various scenarios have been simulated in our Monte Carlo study to compare the different signal-enhancement algorithms. First, the performance of these

peak k	method	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k(°)
1	HTLS	-1374	249	6.9	10
	CA-HTLS	-1371	262	7.0	8
	MV-HTLS	-1374	242	6.1	10
	CR bound	5	30	0.6	6
2	HTLS	-682	247	10.6	10
	CA-HTLS	-681	246	10.6	11
	MV-HTLS	-683	236	9.7	12
	CR bound	4	26	0.7	4
3	HTLS	-263	205	6.2	10
	CA-HTLS	-266	194	6.0	10
	MV-HTLS	-265	185	5.2	11
	CR bound	4	31	0.7	6
4	HTLS	391	710	19.0	28
	CA-HTLS	380	722	18.7	31
	MV-HTLS	357	157	3.2	-1
	CR bound	10	61	1.1	22
5	HTLS	4337	21	0.7	-52
	CA-HTLS	4342	-1	0.8	-59
	MV-HTLS	463	843	16.7	25
	CR bound	19	117	2.0	7

Table 4.1: Parameter estimates and CR lower bound of the noisy ($\sigma_\nu = 2$, see Fig. 4.2) simulated ^{31}P NMR signal, *Signal-2*, using the HTLS, CA-HTLS and MV-HTLS algorithms.

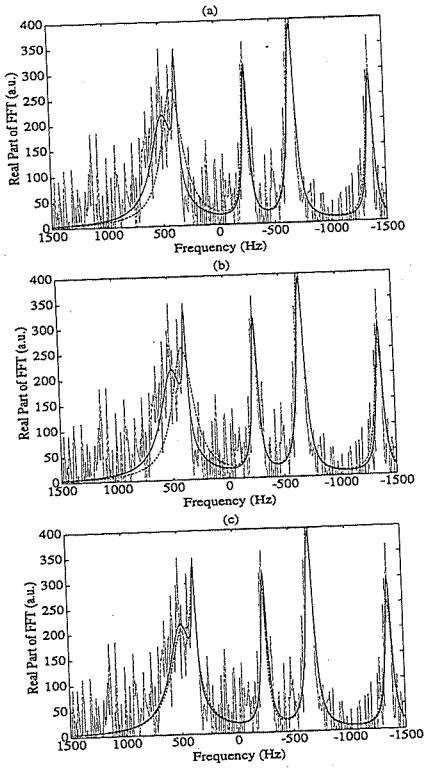


Figure 4.2: Real part of the FFT of the reconstructed simulated ^{31}P NMR signal, *Signal-2*, (dashed line) as estimated by (a) HTLS, (b) CA-HTLS, and (c) MV-HTLS from the same set of noisy data ($\sigma_\nu = 2$, dotted line). The full line represents the exact signal.

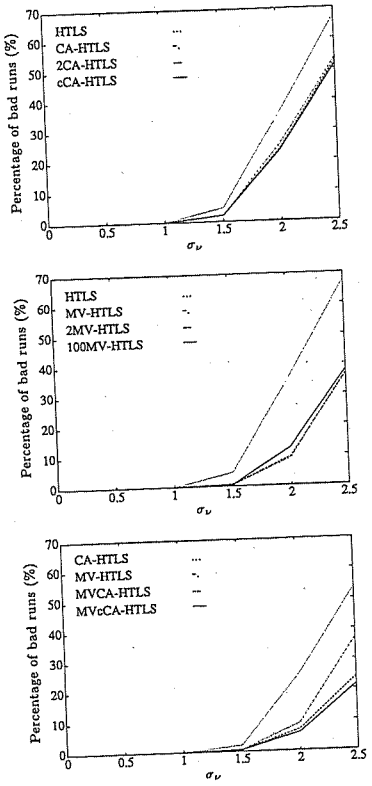


Figure 4.3: Failure rate of HTLS, nCA-HTLS, nMV-HTLS, and MV-nCA-HTLS versus noise standard deviation. Here the MV method does not converge within 100 iterations.

algorithms after one iteration is compared and next the performance upon convergence is considered.

Fig. 4.2 and Table A.3 indicate that peaks 4 and 5 of the signal are two difficult peaks with closely spaced frequencies. Peak 4 lies on the flank of peak 5 in the noise-free case. However, in the presence of white noise, say $\sigma_\nu = 2$, these two peaks are often indistinguishable (dotted line in Fig. 4.2). The detection of these peaks in the presence of noise has been studied using three different algorithms HTLS, CA-HTLS and MV-HTLS, as shown in Fig. 4.2 and Table 4.1. Apparently, only MV-HTLS is able to resolve the closely spaced peaks 4 and 5 in this case. This is also clear from the failure rate plots in Fig. 4.3. These plots indicate that both the MV and Cadzow's enhancement procedures using one iteration improve the resolution significantly. However, the enhanced frequency resolution is better for the MV estimation method than for Cadzow's method. All detected failures in resolution are due to the fact that one of the two overlapping peaks, especially peak 4, cannot be resolved. Furthermore, the occurrence of a failure of MV-HTLS is mostly due to the fact that the estimated peak 4 (or 5) falls outside although close to the specified frequency interval. This is in contrast to Cadzow's method, CA-HTLS, which has more failures. If a failure occurs, peaks 4 and 5 are estimated by only one peak, as illustrated in Fig. 4.2 and Table 4.1, and an additional noise peak appears far away from the exact frequency, often around $\pm$4000Hz. Note also that applying an MV iteration followed by a Cadzow iteration prior to HTLS further reduces the failure rate.

The differences in parameter accuracy between the MV-HTLS and the CA-HTLS methods are small, especially for the estimates of well-resolved peaks 1, 2 and 3. For overlapping peaks 4 and 5, the accuracy of the damping factors and amplitudes of the two methods are comparable as shown in Fig. 4.4 but the frequencies and phases (not shown) are estimated somewhat more accurately by CA-HTLS (maybe due to a smaller bias). Note also that these results are based on average values over all *good* runs. This number differs from one method to the other. It certainly reflects the fact that MV-HTLS is still able to compute accurate estimates of all parameters during the runs in which CA-HTLS fails to resolve peak 4 or 5. The slight loss in accuracy, as discussed earlier in this section, is mainly due to the occurrence of a smaller singular value gap $\hat{\sigma}_K/\hat{\sigma}_{K+1}$ in the $\tilde{L} \times \tilde{M}$ data matrix $\hat{\mathbf{X}}$ used by MV-HTLS compared to the one used by CA-HTLS as shown in Fig. 4.1 and Table 4.2. The increase in the bias of all parameter estimates due to the smaller gap can be observed from Table 4.3. It is to be noted that applying MV-CA-HTLS further improves the accuracy of the estimates wherever CA-HTLS performs better than MV-HTLS.

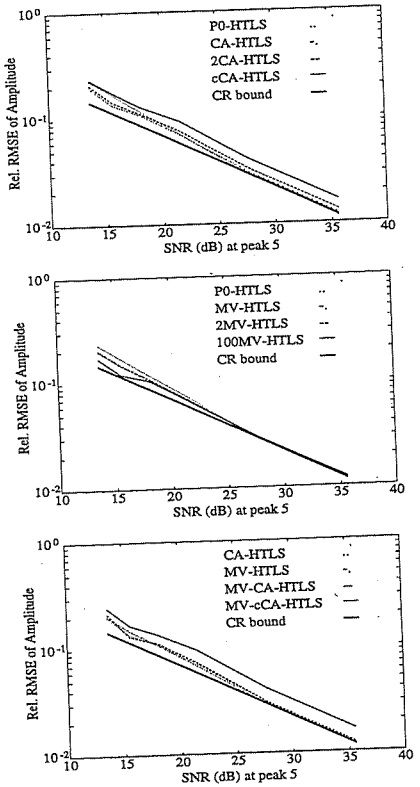


Figure 4.4: Relative RMSE values of a_5 (averaged over good runs) as estimated by HTLS, nCA-HTLS, nMV-HTLS, and MV-nCA-HTLS. The thick full line is the CR bound. $\sigma_\nu = 2$ corresponds to SNR=15.58 dB at peak 5.

method	singular values					
	1st	2nd	3rd	4th	5th	6th
0MV	144.02	122.94	90.75	31.63	24.44	23.99
1MV	139.54	118.97	86.04	24.60	5.75	4.32
2MV	139.39	118.84	85.87	24.30	4.54	1.96
3MV	139.38	118.83	85.85	24.24	4.29	1.63
20MV	139.30	118.77	85.67	23.91	3.42	8.96e-01
500MV	138.99	118.52	84.99	23.26	2.40	2.69e-01
0CA	192.99	152.97	139.79	134.78	49.28	30.31
1CA	192.67	152.43	139.91	134.74	49.05	4.57
2CA	192.58	152.27	139.98	134.82	49.22	2.67
3CA	192.52	152.18	140.02	134.90	49.29	1.75
20CA	192.36	152.03	140.10	135.14	49.39	6.46e-03
51CA	192.36	152.02	140.10	135.14	49.39	3.31e-07
1MV-1CA	187.32	146.52	133.83	130.19	47.25	4.24
1MV-2CA	187.24	146.36	133.90	130.27	47.40	2.55
1MV-20CA	187.05	146.11	134.04	130.55	47.58	8.63e-03
1MV-52CA	187.05	146.11	134.04	130.55	47.58	3.18e-07

Table 4.2: The first $K+1(=6)$ singular values of the noisy ($\sigma_\nu = 2$, see Fig. 4.2) simulated ^{31}P NMR signal, *Signal-2*, processed by different enhancement procedures.

MV-HTLS requires the estimation of the noise variance ν^2_{est} in order to correct the singular values of the Hankel data matrix. For one MV iteration, no differences in accuracy or resolution have been observed between the use of the exact noise variance and the use of one of the estimates, either based on the tail of the signal or based on the average of the squared noise singular values.

When the signal is *very noisy*, it often happens that noise peaks are identified instead of the relevant peaks we are interested in. Overestimating the model order (i.e., taking the order ≥ 5 in our example) at low SNR further reduces the failure rate of both MV-HTLS and CA-HTLS methods, as illustrated in Fig. 4.5, while the differences in failure rate remain relatively the same. However, if K is chosen too large, the accuracy of the computed parameters does not improve and is mostly slightly worse probably due to a smaller accuracy of the estimates of the difficult runs (in which the non-overestimated method fails).

If we increase the number of iterations performed during preprocessing,

peak k/Method	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k (°)
peak 1				
HTLS	-0.28±6.35	-2.98±36.6	0.03±0.72	0.25±7.61
CA-HTLS	-0.24±7.35	5.16±52.0	0.19±0.92	0.27±8.69
MV-HTLS	2.58±7.10	-3.02±46.5	-0.83±0.78	-2.60±7.42
MV-CA-HTLS	0.69±7.44	2.11±49.0	-0.76±0.83	-0.60±8.31
peak 2				
HTLS	0.13±5.91	5.47±35.8	0.14±0.92	-0.04±5.89
CA-HTLS	0.44±6.02	1.34±34.6	0.06±0.92	-0.03±6.11
MV-HTLS	0.74±5.95	4.37±33.6	-0.57±0.85	-0.69±5.65
MV-CA-HTLS	0.17±5.81	0.73±34.4	-0.66±0.89	0.09±5.70
peak 3				
HTLS	-0.26±7.56	1.36±41.5	0.16±0.84	-0.42±9.56
CA-HTLS	0.13±6.98	3.18±42.1	0.23±0.86	-1.12±9.56
MV-HTLS	-1.34±6.92	3.45±38.2	-0.37±0.76	0.43±9.09
MV-CA-HTLS	-0.75±6.81	3.75±40.3	-0.44±0.77	0.16±9.04
peak 4				
HTLS	2.61±17.1	20.1±108.	1.28±2.48	-3.69±35.4
CA-HTLS	0.87±15.1	-9.46±90.9	0.55±1.84	-1.85±36.2
MV-HTLS	-2.05±15.7	-20.6±102.	0.04±2.29	0.75±37.9
MV-CA-HTLS	0.26±14.8	-1.84±107.	0.34±2.01	-1.88±34.9
peak 5				
HTLS	10.7±28.4	-88.3±232.	-1.01±3.81	0.71±9.55
CA-HTLS	0.82±26.9	-57.4±179.	-0.25±3.41	1.02±8.73
MV-HTLS	-15.8±25.7	-69.8±161.	-1.18±3.28	4.83±9.57
MV-CA-HTLS	-6.20±28.3	-43.3±177.	-1.46±3.38	3.02±8.83

Table 4.3: Bias±standard deviation values of all parameter estimates of the noisy ($\sigma_\nu = 2.5$) simulated ^{31}P NMR signal, *Signal-2* (see Fig. 4.3 through Fig. 4.4).

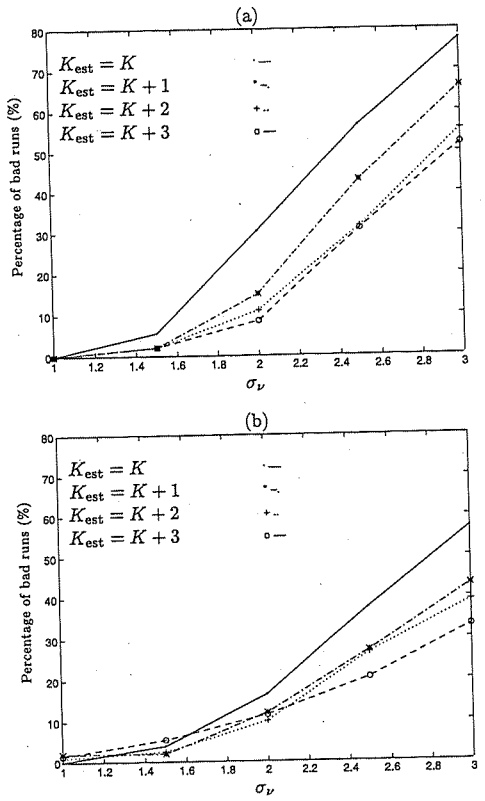


Figure 4.5: Failure rate of (a) CA-HTLS and (b) MV-HTLS versus σ_ν for different estimated model orders K_{est}.

the following conclusions can be drawn. As shown in Fig. 4.3, the failure rate decreases in the first few iterations, after which it hardly changes. To be more specific, the first iteration is the most decisive one while the procedure applied up to convergence hardly improves the resolution. Inspection of the spectra plotted in Fig. 4.6 reveals that the "cleaning" effect of the enhancement procedures becomes smaller as the number of iterations increases. The typical number of iterations required by the different methods is listed in Table 4.4. Although the MV method needs only 4.4% of the computation time of Cadzow's method for each iteration, it needs a much larger number of iterations to achieve the same degree of smoothness in the spectrum. A similar increase in the number of required iterations is needed when Cadzow's method is applied to a rectangular (instead of a square) data matrix. Although the computation time for one iteration reduces significantly, the number of iterations needed for convergence increases at such a rate that the overall computation time is even larger than the time needed to apply Cadzow's method to a square data matrix up to convergence (in contrast to the assertion in [60]). Tests have also been performed with the MV method using square (instead of long) Hankel matrices, and the required number of iterations is comparable to that required by Cadzow's method. *Hence, it is the* size *of the Hankel matrix used and not the applied singular-value correction function which is crucial for a quick convergence.* This is due to the fact that the square matrix has a larger gap between the Kth and $(K+1)$th singular values, as shown in Table 4.2, and averages over more elements when restoring the Hankel structure.

Furthermore, when the number of iterations in the enhancement procedure increases, the accuracy of the estimated parameters as illustrated in Fig. 4.4 remains the same or deteriorates. One reason might be that *the MV and Cadzow's enhancement procedures converge to a suboptimal instead of the optimal solution as pointed out in [29][64], i.e., these procedures converge to a "clean" rank-K Hankel matrix but not to the one closest to the original noisy data matrix.* See Chapter 5 for a comparison of these suboptimal enhancement methods with optimal ones. Another more important reason is that the MV and Cadzow's iterations introduce colored noise into the preprocessed data. As pointed out in [65], the SVD truncation and element averaging along the cross diagonals in each iteration is equivalent to subtraction of a zero-phase filtered version of the original signal from itself. Such a filtering operation masks the noise at the K signal peaks, and thus turns white noise into colored one. The colored noise causes the deviation of the convergent "clean" signal peaks from the exact ones (bias). Therefore, a "cleaner" spectrum does not guarantee better accuracy. *The enormous computational cost and the unsatisfactory results lead to the conclusion that it is not necessary to apply the MV and Cadzow's*

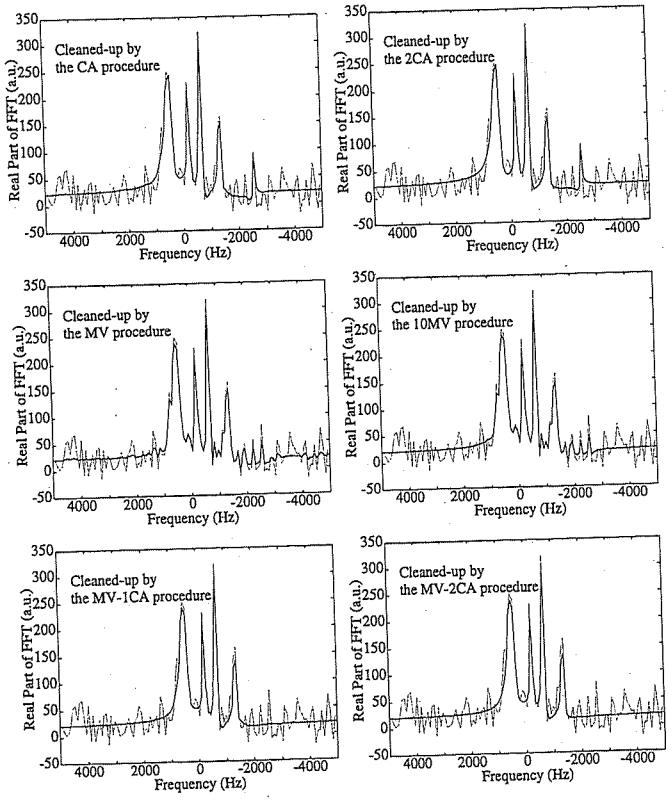


Figure 4.6: "Cleaned-up" spectra (full line) of the same set of noisy data ($\sigma_\nu = 2$, dotted line) using the nMV, nCA and MV-nCA enhancement procedures.

enhancement procedures up to convergence.

enhancement	for visually "clean" spectrum	for convergence
nMV	$n \approx 10$	$n \geq 1000$
nCA	$n \approx 1$ or 2	$n \approx 47$
MV-nCA	$1+n$ and $n \approx$(1 or 2)	$1+n$ and $n \approx 48$

Table 4.4: Typical numbers of iterations required by different enhancement procedures for the simulated noisy ($\sigma_\nu = 2$) ^{31}P NMR signal, *Signal-2*, with model order $K = 5$.

In summary, the MV-HTLS method is the best choice for estimating parameters of signals with overlapping exponentially damped peaks at low computational cost, when the SNR is low and no prior knowledge is available. The model order can be chosen equal to or somewhat larger than the exact one and a matrix size of $K + 5 \leq M \leq 2K$ is suggested for the MV enhancement procedure.

4.5 Application to real-world NMR signals

The application of the enhancement algorithms discussed above to real-world NMR signals is described in this section. The measurement data are assumed to be modeled by the function defined by Eq. (1.4) and perturbed by additive white noise. When $t_0 \neq 0$ and $\phi^{(0)} \neq 0$, the signal parameters estimated by HTLS must be extrapolated to the time origin and the phases must be corrected.

Fig. 4.7 shows the signal to be processed, which is a ^{31}P NMR FID signal from a perfused rat liver. It was obtained in a 4.7 Tesla, 30 cm wide-bore magnet, equipped with a Biospec spectrometer, and acquired at 81.1 MHz. Two hundred fifty-six scans were accumulated using a 60 μs (72°) pulse and a 0.69 s repetition time. The peaks of interest are PDE (phospho-diesters), P_i (inorganic phosphate), and PME (phospho-monoesters) which together form a cluster in the spectrum. The peak at the highest frequency is an external standard with known concentration, which allows absolute concentration estimation from the computed amplitudes (extrapolated to the time origin) of the peaks. The zero-order phase $\phi^{(0)} = \varphi^{(0)} * 180/\pi$ is 23°, determined from the spectrometer phase settings, and the dead time is 0 s. In our fit, 150 data points are taken, starting from the sixth point (x_6) on (the first 5 are deleted in

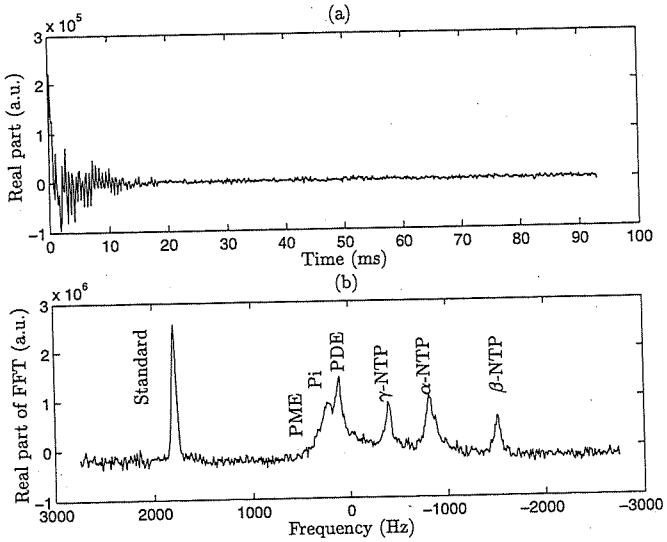


Figure 4.7: (a) ^{31}P-NMR FID signal of perfused rat liver. (b) The corresponding NMR spectrum (after omission of the first five data points).

order to eliminate the broad hump that appears in the original spectrum). The chosen model order, $K = 15$, is slightly overestimated in order to obtain estimates of all three PDE-P_i-PME peaks, among which the smallest PME peak is rather difficult to resolve. The size of the Hankel data matrix is 129×22 in the MV procedure and 76×75 in Cadzow's procedure. The ν_{est} used in the MV method is estimated from the noise singular values.

Table 4.5 presents the extrapolated and phase-corrected parameter estimates of the three overlapping peaks PDE-P_i-PME using the methods HTLS, CA-HTLS and MV-HTLS respectively. CR bounds are provided to indicate the accuracy of our estimates. Based on the physical reality that phases should be close to zero after extrapolation and phase correction, the MV-HTLS method produces the best fit of the three individual peaks. In addition, it provides the

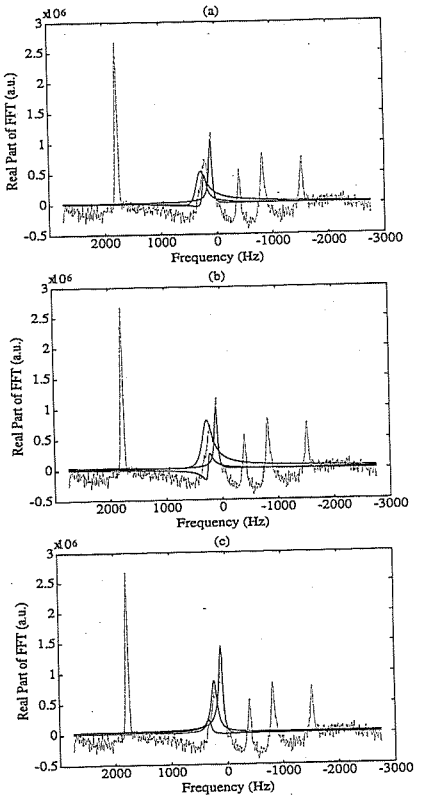


Figure 4.8: Spectra of the exponentially damped sinusoidal components that reconstruct the three overlapping peaks PDE-P_i-PME of the real-world NMR signal, as computed by (a) HTLS (b) CA-HTLS and (c) MV-HTLS.

physically most relevant damping factors. The other two methods, HTLS and CA-HTLS, underestimate the damping factor of the P_i peak and overestimate the damping factor of the PME peak. These can be visualized by drawing the Fourier transform of the exponentially damped sinusoidal components that fit each peak separately as shown in Fig. 4.8. It can be clearly seen that the "P_i" and "PME" peaks obtained by HTLS and CA-HTLS do not bear physical meaning anymore although the overall fits (i.e., the combination of all peaks) by all three methods mentioned above have a similar, small residue. The parameter estimates obtained using cCA-HTLS, cMV-HTLS, MV-CA-HTLS and MV-cCA-HTLS are even worse and are not included here. We therefore conclude that MV-HTLS is the best method for this case.

peak k	method	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k (°)
PDE	HTLS	103± 2	197± 12	37716± 2475	3± 4
	CA-HTLS	103± 2	191± 12	35276± 2382	2± 4
	MV-HTLS	95±11	266± 67	69178±29303	10±24
P_i	HTLS	236± 9	257± 54	24324± 8668	-44±20
	CA-HTLS	249 ±12	232± 75	18487±11958	-85±37
	MV-HTLS	225 ±12	397± 77	60455±20266	-4±19
PME	HTLS	322±13	673± 79	69440±11021	-44± 9
	CA-HTLS	292±13	681± 80	97950±14810	-29± 9
	MV-HTLS	318±23	325±147	13522±10757	13±46

[a] computed from the parameter estimates.

Table 4.5: Real-world NMR signal-parameter estimates and their CR bounds[a] corresponding to the three exponentially damped sinusoidal components which fit the peaks PDE-P_i-PME.

4.6 Conclusions

A new enhancement algorithm based on minimum variance estimation for fitting a sum of damped exponential signals embedded in white noise has been described. Comparison of the MV method and Cadzow's method is made, based on extensive simulations and real-world NMR signals. When nonoverlapping peaks are considered and one iteration of the enhancement procedure is used, both methods are comparable in resolution and parameter accuracy but the MV method is much more efficient. However, when overlapping peaks are

to be quantified, then an MV iteration doubles the resolution independently of the noise variance at comparable parameter accuracy, while the computation time is less than 10% of the time needed to execute Cadzow's iteration, depending on the number of data points and the model order. Considering both resolution and accuracy, one or two iterations of the enhancement procedure are sufficient while the convergence of the procedure is not necessary, offering drastic reduction in the computational cost, which is valuable in practical NMR signal processing. Furthermore, overestimating the model order is helpful in quantifying some difficult peaks.

Similar to the problem of prefiltering techniques in the previous chapter, however, the MV and Cadzow enhancement procedures introduce colored noise into the preprocessed data. This is because truncating the SVD and averaging the elements along the cross diagonals in each iteration is equivalent to subtracting a zero-phase filtered version of the original signal from itself as proven in [65]. Such a filtering operation masks the noise at the K signal peaks, and thus turns white noise into colored noise. New procedures that minimize the effects of the colored noise are being developed (see [66]).

Chapter 5

Structured TLN and structured TLS in signal enhancement

In SVD and subspace-based parameter estimation algorithms, the rank-K matrix, computed from the "signal" subspace of the original Hankel data matrix, does not preserve the Hankel structure anymore. This problem has been handled in the previous chapter where signal-enhancement algorithms at convergence recover both the Hankel structure and the rank-K property of data matrices through composite property mapping. Such preprocessing procedures result in enhanced data matrices of which the elements better represent the underlying signal components than did the original data. With the development of numerical techniques, new algorithms for the problem of approximating structured rank-deficient matrices have emerged. As extensions of the TLS technique, the constrained TLS (CTLS) [67][43], structured TLS (STLS) [68][64] and structured total least norm (STLN) techniques [69][70][71] have been proposed one after another. Recently, CTLS and STLS have been proven to be equivalent [72]. Hence, we will focus only on the application of STLN and STLS to signal enhancement of our parameter estimation problem in this chapter.

The subsequent sections are organized as follows. Section 5.1 formulates the STLN and STLS problem. Section 5.2 gives solutions and outlines of the STLN and STLS algorithms for complex Hankel/Toeplitz structured data matrices. Section 5.3 presents a more accurate parameter estimator, HTLN, for

our problem of exponential data fitting, which starts from the same Hankel data matrix as in HTLS but uses the STLN technique prior to the HTLS parameter estimation method. In Section 5.4, comparisons of the STLN (HTLN), STLS algorithms with the SVD-based suboptimal procedures are made using complex exponential data in terms of computational efficiency and accuracy.

5.1 Formulation of STLN and STLS

Let

$$[\mathbf{A}_{L\times K} \quad \mathbf{B}_{L\times(M-K)}] = \mathbf{X}, \qquad L \geq K, M \geq K+1, N = L+M-1$$

where $\mathbf{A}$ consists of the first K columns and $\mathbf{B}$ consists of the remaining $M-K$ columns of $\mathbf{X}$, which is the Hankel data matrix defined in (2.1). Since the rank of the matrix $\mathbf{X} = [\mathbf{A} \quad \mathbf{B}]$ is K when no noise is added, there exists a $K \times (M-K)$ matrix $\mathbf{Y}$ such that $\mathbf{AY} = \mathbf{B}$. If noise is added, the corresponding set of linear equations $\mathbf{AY} \approx \mathbf{B}$ is no longer compatible, since $[\mathbf{A} \quad \mathbf{B}]$ is typically of full rank. The rank-K property of $[\mathbf{A} \quad \mathbf{B}]$ can be restored by computing appropriate corrections $[\Delta\mathbf{A} \quad \Delta\mathbf{B}]$ such that $[\mathbf{A}+\Delta\mathbf{A} \quad \mathbf{B}+\Delta\mathbf{B}]$ has rank K, i.e., $(\mathbf{A}+\Delta\mathbf{A})\mathbf{Y} = \mathbf{B}+\Delta\mathbf{B}$ is exactly solvable.

TLS problem

The well-known TLS technique for solving $\mathbf{AY} \approx \mathbf{B}$ is formulated as

$$\min_{\Delta\mathbf{A},\Delta\mathbf{B},\mathbf{Y}} \|[\Delta\mathbf{A} \quad \Delta\mathbf{B}]\|_2^2 \quad \text{subject to } (\mathbf{A}+\Delta\mathbf{A})\mathbf{Y} = \mathbf{B}+\Delta\mathbf{B}$$

This method allows to modify both matrices $\mathbf{A}$ and $\mathbf{B}$ by adding correction matrices $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$, respectively, and is justified when $\mathbf{A}$, as well as $\mathbf{B}$, are affected by noise. The computational method generally used for solving TLS is based on the SVD of $[\mathbf{A} \quad \mathbf{B}]$ [6]. But in applications where the matrix $\mathbf{A}$ or $[\mathbf{A} \quad \mathbf{B}]$ has a special structure like Hankel, Toeplitz, etc., the SVD-based methods may not always be appropriate since they do not preserve the special structure. In fact, the matrix $[\Delta\mathbf{A} \quad \Delta\mathbf{B}]$ obtained from the SVD will typically be dense, with no structure, even when $\mathbf{A}$ or $[\mathbf{A} \quad \mathbf{B}]$ is structured.

STLN problem

The above problem can be alleviated by the newly developed technique, STLN, proposed by Rosen, Park and Glick [69] and extended to multiple right-hand sides by Van Huffel, Park and Rosen [71]. The STLN technique preserves

any given affine structure in $\Delta\mathbf{A}$ or $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$, such as Hankel, Toeplitz, or sparse, and can minimize these corrections in the discrete L_p norm, where $p = 1, 2$ or ∞. It can be formulated as

$$\min_{\Delta\mathbf{A},\Delta\mathbf{B},\mathbf{Y}} \quad \|[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]\|_p^2$$
$$\text{subject to} \quad \begin{cases} (\mathbf{A}+\Delta\mathbf{A})\mathbf{Y} = (\mathbf{B}+\Delta\mathbf{B}) \\ [\Delta\mathbf{A}\ \ \Delta\mathbf{B}] \text{ preserves the given structure} \end{cases} \quad p = 1, 2, \infty$$

where $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$ are correction matrices with the given special structure.

This formulation implies that STLN for $p = 2$ reduces to TLS if all elements of $[\mathbf{A}\ \mathbf{B}]$ are subject to error and the structure is not taken into account.

In our application, *only the STLN algorithm for L_2 norm is considered, that is,* $p = 2$. In particular, when $\mathbf{B}$ is only a column vector $\mathbf{B} = \mathbf{b}$ and $\mathbf{X} = [\mathbf{A}\ \mathbf{b}]$ is Hankel/Toeplitz structured, the STLN problem which approximates $\mathbf{A}\mathbf{y} \approx \mathbf{b}$ by the closest compatible set $(\mathbf{A}+\Delta\mathbf{A})\mathbf{y} = (\mathbf{b}+\Delta\mathbf{b})$ of the same structure can be formulated as

$$\min_{\alpha,\beta,\mathbf{y}} \left\| \mathbf{D} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \right\|_2^2 \quad \text{subject to} \quad \mathbf{r}(\alpha,\beta,\mathbf{y}) = \mathbf{0} \tag{5.1}$$

where $\alpha \in \mathbb{C}^{q_\alpha \times 1}$, $\beta \in \mathbb{C}^{q_\beta \times 1}$ are the correction vectors representing, respectively, the q_α, q_β *different* elements of the correction matrix $\Delta\mathbf{A}$, $\Delta\mathbf{b}$ subject to error; $\mathbf{D} \in \mathbb{R}^{(q_\alpha+q_\beta)\times(q_\alpha+q_\beta)}$ is a diagonal matrix with positive weights that takes into account the multiplicity of the elements of α, β in $[\Delta\mathbf{A}\ \ \Delta\mathbf{b}]$; and $\mathbf{r}(\alpha,\beta,\mathbf{y}) = (\mathbf{b}+\Delta\mathbf{b}) - (\mathbf{A}+\Delta\mathbf{A})\mathbf{y}$ is the modified residual vector of the set of linear equations. This formulation of the STLN problem for $p = 2$ is derived in the next section. Actually, it is an optimization problem with quadratic objective function and nonlinear constraints.

STLS problem

When $\mathbf{B}$ has only one column, $\mathbf{B} = \mathbf{b}$, and $\mathbf{X} = [\mathbf{A}\ \mathbf{b}]$ is affinely structured, the STLS technique proposed by De Moor [68][64] can also be used after being extended from real data to complex data. STLS attempts to approximate affinely structured matrices by similarly structured rank-deficient ones, while minimizing an L_2-error criterion, which approximates a given data sequence $\mathbf{x} = [x_0, x_1, \ldots, x_{N-1}]^T$ by $\hat{\mathbf{x}} = [\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{N-1}]^T$ such that the (weighted) sum of squares of residuals is minimized.

Let $\mathbf{W} = \mathrm{diag}(w_0, w_1, \ldots, w_{N-1}) \in \mathbb{R}^{N\times N}$ be a diagonal weighting matrix. Choose N fixed matrices $\mathbf{X}_n \in \mathbb{R}^{L\times M}, n = 0, 1, \ldots, N-1$, representing the

structure of $[\mathbf{A}\ \mathbf{b}]$ so that an affinely structured matrix can be written as an affine matrix function of its components: $\mathbf{X} = \Sigma_{n=0}^{N-1} x_n \mathbf{X}_n$ and $\hat{\mathbf{X}} = \Sigma_{n=0}^{N-1} \hat{x}_n \mathbf{X}_n$. Then STLS is formulated as

$$\min_{\hat{\mathbf{x}}\in\mathbb{C}^{N\times 1}, \mathbf{y}\in\mathbb{C}^{M\times 1}} \quad \Sigma_{n=0}^{N-1} |x_n - \hat{x}_n|^2 w_n \quad \text{subject to} \quad \begin{cases} \hat{\mathbf{X}}\mathbf{y} = \mathbf{0} \\ \mathbf{y}^H\mathbf{y} = 1 \end{cases} \tag{5.2}$$

Remark that the structure of $\hat{\mathbf{X}}$ is forced to be the same as the structure of $\mathbf{X}$ since $\hat{\mathbf{X}}$ is written as an affine combination of the same $\mathbf{X}_n$ as $\mathbf{X}$. This formulation implies that STLS reduces to TLS if no structure of the matrix $\mathbf{X} = [\mathbf{A}\ \mathbf{b}]$ or weighting of its elements is taken into account.

5.2 STLN and STLS algorithms applied to complex Hankel/Toeplitz data matrices

In what follows, we solve the STLN and STLS problem and outline the STLN, using the 2-norm ($p = 2$), and STLS algorithms for data matrices having complex elements and a Hankel structure. The results also hold for Toeplitz matrices, since Toeplitz matrices simply transform to Hankel matrices by permutations. Algorithms for other structures and norms are not treated here and can be found in [69][68].

5.2.1 The STLN algorithms applied to complex Hankel data matrices

We consider two STLN algorithms according to the dimension of the complex Hankel data matrix:

1. $[\mathbf{A}_{L\times K}\ \mathbf{b}_{L\times 1}]$ has $M = K+1$ columns, which corresponds to the single-right-hand side problem $\mathbf{A}\mathbf{y} \approx \mathbf{b}$.

2. $[\mathbf{A}_{L\times K}\ \mathbf{B}_{L\times(M-K)}]$ has $M > K+1$ columns, which corresponds to the multiple-right-hand side problem $\mathbf{A}\mathbf{Y} \approx \mathbf{B}$.

STLN for Hankel structured $[\mathbf{A}_{L\times K}\ \mathbf{b}_{L\times 1}]$

The main trick of STLN in approximating $\mathbf{A}\mathbf{y} \approx \mathbf{b}$ by $(\mathbf{A}+\Delta\mathbf{A})\mathbf{y} = (\mathbf{b}+\Delta\mathbf{b})$ is to use the correction vectors α and β which represent *different* non-zero elements of the correction matrices $\Delta\mathbf{A}$ and $\Delta\mathbf{b}$. Here, the vector α, β and the matrix $\Delta\mathbf{A}, \Delta\mathbf{b}$ are equivalent in the sense that α, β is known when

$\Delta\mathbf{A}, \Delta\mathbf{b}$ is given and vice versa. In this way, the STLN problem turns into an optimization problem (5.1) over the vectors $\alpha, \beta, \mathbf{y}$, based on the following relationship between $\Delta\mathbf{A}$ and α, $\Delta\mathbf{b}$ and β, as shown below.

- To represent the vector $\Delta\mathbf{A}\mathbf{y}$ in terms of α, we define a matrix $\check{\mathbf{Y}} \in \mathbb{C}^{L \times q_\alpha}$ such that $\Delta\mathbf{A}\mathbf{y} = \check{\mathbf{Y}}\alpha$. The construction of $\check{\mathbf{Y}}$ (starting with a zero matrix) is carried out according to the following rule:
If α_k is the $(i,j)^{th}$ element of $\Delta\mathbf{A}$, then y_j is the $(i,k)^{th}$ element of $\check{\mathbf{Y}}$, where $i = 1, \ldots, L$, $j = 1, \ldots, K$, and $k = 1, \ldots, q_\alpha$.
The matrix $\check{\mathbf{Y}}$ consists of the elements of $\mathbf{y}$, with suitable repetition, giving $\check{\mathbf{Y}}$ a special structure. In fact, for Hankel structured $\Delta\mathbf{A}$, we have

$$\begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_K \\ \alpha_2 & \alpha_3 & \cdots & \alpha_{K+1} \\ \vdots & \vdots & & \vdots \\ \alpha_L & \alpha_{L+1} & \cdots & \alpha_{q_\alpha} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_K \end{bmatrix}$$

$$= \begin{bmatrix} y_1 & y_2 & \cdots & y_K & 0 & \cdots & 0 \\ 0 & y_1 & y_2 & \cdots & y_K & \ddots & \vdots \\ \vdots & \ddots & y_1 & y_2 & \cdots & y_K & 0 \\ 0 & \cdots & 0 & y_1 & y_2 & \cdots & y_K \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{q_\alpha} \end{bmatrix} \tag{5.3}$$

- Since all the elements of $\mathbf{b}$ are subject to error in our case, we have $\beta = \Delta\mathbf{b}$, with $q_\beta = L$.
- In Hankel structured $[\mathbf{A}\ \mathbf{b}]$, the elements of $\mathbf{b}$ are linearly related with the elements of $\mathbf{A}$. It follows that β is also linearly related with α such that

$$\beta = \mathbf{P}_\beta \beta_0 + \mathbf{P}_\alpha \alpha,$$

where $\mathbf{P}_\beta = [0, \cdots, 0, 1]^T$, , $\mathbf{P}_\alpha = \begin{bmatrix} \mathbf{0}_{(L-1)\times K} & \mathbf{I}_{(L-1)\times(L-1)} \\ \mathbf{0}_{1\times K} & \mathbf{0}_{1\times(L-1)} \end{bmatrix}$, and $\beta_0 \in \mathbb{C}^{1\times 1}$.

- In error minimization, α and β_0 should be weighted by diagonal weighting matrices $\mathbf{D}_\alpha$ and $\mathbf{D}_\beta$, respectively, that account for the repetition of the elements of α and β_0 in $[\Delta\mathbf{A}\ \ \Delta\mathbf{b}]$. For our Hankel structured matrix $[\Delta\mathbf{A}\ \ \Delta\mathbf{b}]$, $\begin{bmatrix} \mathbf{D}_\alpha^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta^2 \end{bmatrix} = \mathrm{diag}(1, 2, \cdots, q, \cdots, q, \cdots, 2, 1)$, $q = \min(L, K+1)$, and the object function in (5.1) can be written as

$$f(\alpha, \beta_0) = \| \begin{bmatrix} \mathbf{D}_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta \end{bmatrix} \begin{bmatrix} \alpha \\ \beta_0 \end{bmatrix} \|_2^2 = \alpha^H \mathbf{D}_\alpha^2 \alpha + \beta_0^H \mathbf{D}_\beta^2 \beta_0. \tag{5.4}$$

- The modified residual vector in the nonlinear constraint of the optimization problem in (5.1) is a function of $\alpha, \beta, \mathbf{y}$ and can be written as

$$\begin{aligned} \mathbf{r}(\alpha, \beta, \mathbf{y}) &= (\mathbf{b} + \Delta\mathbf{b}) - (\mathbf{A} + \Delta\mathbf{A})\mathbf{y} \\ &= \mathbf{b} + \beta - \mathbf{A}\mathbf{y} - \check{\mathbf{Y}}\alpha \end{aligned}$$

which is nonlinear in the parameters $\alpha, \mathbf{y}$ due to the last term on the right-hand side. The nonlinearity can also be seen from

$$\begin{aligned} \mathbf{r} & (\alpha + \Delta\alpha, \beta + \Delta\beta, \mathbf{y} + \Delta\mathbf{y}) \\ &= \mathbf{b} + (\beta + \Delta\beta) - \mathbf{A}(\mathbf{y} + \Delta\mathbf{y}) - (\check{\mathbf{Y}} + \Delta\check{\mathbf{Y}})(\alpha + \Delta\alpha) \\ &= (\mathbf{b} + \beta - \mathbf{A}\mathbf{y} - \check{\mathbf{Y}}\alpha) + \Delta\beta - \mathbf{A}\Delta\mathbf{y} - \check{\mathbf{Y}}\Delta\alpha - \Delta\check{\mathbf{Y}}\alpha \\ &\quad -\Delta\check{\mathbf{Y}}\Delta\alpha \\ &= \mathbf{r}(\alpha, \beta, \mathbf{y}) - \check{\mathbf{Y}}\Delta\alpha + \Delta\beta - (\mathbf{A} + \Delta\mathbf{A})\Delta\mathbf{y} - \Delta\check{\mathbf{Y}}\Delta\alpha \\ &= \mathbf{r}(\alpha, \beta_0, \mathbf{y}) - (\check{\mathbf{Y}} - \mathbf{P}_\alpha)\Delta\alpha + \mathbf{P}_\beta\Delta\beta_0 - (\mathbf{A} + \Delta\mathbf{A})\Delta\mathbf{y} \\ &\quad -\Delta\check{\mathbf{Y}}\Delta\alpha \end{aligned} \tag{5.5}$$

Several solution methods to solve our problem (5.1) with the constraints given in (5.5) have been presented up till now. The accuracy and convergence rate of the STLN algorithm depends on which method is used to solve this optimization problem. In the sequel, two methods labeled as STLN Algorithm-1 and STLN Algorithm-2, respectively, are introduced and outlined.

STLN Algorithm-1

STLN Algorithm-1 is the weighting method described in [69][70][71]. It corresponds to the penalty function method in optimization theory [73][74]. Let $\mathbf{z} = \begin{bmatrix} \alpha \\ \beta_0 \\ \mathbf{y} \end{bmatrix}$ and define a penalty function by weighting the constraint violations $\mathbf{r}(\mathbf{z})^H\mathbf{r}(\mathbf{z})$, and adding it to the objective function in (5.4).

$$\begin{aligned} P(\mathbf{z}) &= f(\mathbf{z}) + w^2\mathbf{r}(\mathbf{z})^H\mathbf{r}(\mathbf{z}) \\ &= \alpha^H\mathbf{D}_\alpha^2\alpha + \beta_0^H\mathbf{D}_\beta^2\beta_0 + w^2\mathbf{r}(\alpha, \beta_0, \mathbf{y})^H\mathbf{r}(\alpha, \beta_0, \mathbf{y}) \end{aligned} \tag{5.6}$$

The constrained optimization problem is then reduced to an unconstrained optimization problem:

$$\min_{\mathbf{z}} P(\mathbf{z}) \quad \text{or} \quad \min_{\alpha,\beta_0,\mathbf{y}} \left\| \begin{bmatrix} w\mathbf{r}(\alpha,\beta_0,\mathbf{y}) \\ \mathbf{D}_\alpha \alpha \\ \mathbf{D}_\beta \beta_0 \end{bmatrix} \right\|_2^2 \tag{5.7}$$

w is a non-negative weight, which has to be taken large for the unconstrained and the constrained problem to be equivalent. It can be shown that, under mild conditions, the solution of (5.7) converges to the solution of (5.1) when w approaches infinity [74]. However, w cannot be taken too large due to the numerical problems, since the condition number of the Hessian matrix of the penalty function $P(\mathbf{z})$ increases as w becomes larger, with singularity occurring in the limit. The problems in choosing a suitable weight w makes automatic implementation of this algorithm difficult.

In the following outline of STLN Algorithm-1, $\mathbf{r}(\alpha,\beta_0,\mathbf{y})$ is linearized by omitting the second order terms $-\Delta\check{\mathbf{Y}}\Delta\alpha$ in (5.5):

$$\begin{aligned} \mathbf{r} \quad & (\mathbf{z}+\Delta\mathbf{z}) \\ = \quad & \mathbf{r}(\alpha+\Delta\alpha,\beta_0+\Delta\beta_0,\mathbf{y}+\Delta\mathbf{y}) \\ \approx \quad & \mathbf{r}(\alpha,\beta_0,\mathbf{y}) - (\check{\mathbf{Y}}-\mathbf{P}_\alpha)\Delta\alpha + \mathbf{P}_\beta\Delta\beta_0 - (\mathbf{A}+\Delta\mathbf{A})\Delta\mathbf{y} \\ = \quad & \mathbf{r}(\mathbf{z}) + [\mathbf{P}_\alpha - \check{\mathbf{Y}} \quad \mathbf{P}_\beta \quad -(\mathbf{A}+\Delta\mathbf{A})]\,\Delta\mathbf{z} \end{aligned} \tag{5.8}$$

so that our problem transforms to a sequence of LS minimization subproblems.

Algorithm 5

Outline of the STLN Algorithm-1 for Hankel structured $[\mathbf{A}_{L\times K} \;\; \mathbf{b}_{L\times 1}]$

Given. $\mathbf{A}, \mathbf{b}, \mathbf{D}_\alpha, \mathbf{D}_\beta$, *Hankel structure of* $[\mathbf{A}\;\mathbf{b}]$, *tolerance* ϵ.

Step 1: Initialization. Choose a large number w. *Set* $\alpha = \mathbf{0}$, $\beta_0 = \mathbf{0}$, $\Delta\mathbf{A} = \mathbf{0}$, *compute the LS solution* $\mathbf{y}$ *from* $\min_{\mathbf{y}} \|\mathbf{A}\mathbf{y} - \mathbf{b}\|_2$, $\check{\mathbf{Y}}$ *from* $\mathbf{y}$ *according to (5.3), and* $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{y}$.

Step 2: Iteration. Repeat

(a) Solve the following LS problem

$$\min_{\Delta\alpha,\Delta\beta_0,\Delta\mathbf{y}} \left\| \begin{bmatrix} w(\mathbf{P}_\alpha - \check{\mathbf{Y}}) & w\mathbf{P}_\beta & -w(\mathbf{A}+\Delta\mathbf{A}) \\ \mathbf{D}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0} \end{bmatrix} \right.$$

$$\left. \begin{bmatrix} \Delta\alpha \\ \Delta\beta_0 \\ \Delta\mathbf{y} \end{bmatrix} + \begin{bmatrix} w\mathbf{r} \\ \mathbf{D}_\alpha\alpha \\ \mathbf{D}_\beta\beta_0 \end{bmatrix} \right\|_2^2$$

(b) $\alpha := \alpha + \Delta\alpha, \beta_0 := \beta_0 + \Delta\beta_0, \mathbf{y} := \mathbf{y} + \Delta\mathbf{y}$

(c) Construct $\Delta\mathbf{A}$ *from* α, $\Delta\mathbf{b}$ *from* α *and* β_0, $\check{\mathbf{Y}}$ *from* $\mathbf{y}$. *Compute* $\mathbf{r} = (\mathbf{b}+\Delta\mathbf{b}) - (\mathbf{A}+\Delta\mathbf{A})\mathbf{y}$.

Until $(\|\Delta\alpha\|, \|\Delta\beta_0\|, \|\Delta\mathbf{y}\|, \|\mathbf{r}\| \le \epsilon)$ (5.9)

STLN Algorithm-2

The STLN Algorithm-2 used in [45] is different from the STLN Algorithm-1 only in Step 2. It is also based on linearizing the nonlinear constraints of our optimization problem (5.1) as done in (5.8). Unlike the weighting method in STLN Algorithm-1, however, which reduces the equality constrained problem to an unconstrained one, STLN Algorithm-2 solves the resulting sequence of LS subproblems with linear equality constraints (LSE) of the following form using the generalized SVD (GSVD) [75] :

$$\min_{\Delta\mathbf{z}} \left\|\hat{\mathbf{A}}\Delta\mathbf{z} - \hat{\mathbf{b}}\right\|_2^2 \quad \text{subject to} \quad \hat{\mathbf{B}}\Delta\mathbf{z} - \hat{\mathbf{d}} = \mathbf{0} \tag{5.10}$$

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{D}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0} \end{bmatrix} \in \mathbb{C}^{\hat{m}\times\hat{n}}, \; \Delta\mathbf{z} = \begin{bmatrix} \Delta\alpha \\ \Delta\beta_0 \\ \Delta\mathbf{y} \end{bmatrix} \in \mathbb{C}^{\hat{n}\times 1},$$

$$\hat{\mathbf{B}} = \left[\mathbf{P}_\alpha - \check{\mathbf{Y}} \quad \mathbf{P}_\beta \quad -(\mathbf{A}+\Delta\mathbf{A})\right] \in \mathbb{C}^{\hat{p}\times\hat{n}},$$

$$\hat{\mathbf{b}} = -\begin{bmatrix} \mathbf{D}_\alpha\alpha \\ \mathbf{D}_\beta\beta_0 \end{bmatrix} \in \mathbb{C}^{\hat{m}\times 1}, \; \hat{\mathbf{d}} = -\mathbf{r} \in \mathbb{C}^{\hat{p}\times 1},$$

$$\hat{m} = N = L+K, \; \hat{n} = L+2K, \; \hat{p} = L, \;\text{ and } \hat{p} < \hat{m} < \hat{n}.$$

The basic idea is to perform a GSVD of $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$, such that

$$\hat{\mathbf{A}} = \mathbf{U}_A\mathbf{D}_A\mathbf{Z}_{AB}, \quad \hat{\mathbf{B}} = \mathbf{U}_B\mathbf{D}_B\mathbf{Z}_{AB} \tag{5.11}$$

where $\mathbf{U}_A \in \mathbb{C}^{\hat{m}\times\hat{m}}$ and $\mathbf{U}_B \in \mathbb{C}^{\hat{p}\times\hat{p}}$ are unitary, $\mathbf{Z}_{AB} \in \mathbb{C}^{\hat{n}\times\hat{n}}$ is a full-rank matrix with the same singular values as those of $\begin{bmatrix}\hat{\mathbf{A}}\\ \hat{\mathbf{B}}\end{bmatrix}$. $\mathbf{D}_A = [\mathbf{P}_A, \ \mathbf{0}] \in \mathbb{R}^{\hat{m}\times\hat{n}}$, $\mathbf{P}_A = \text{diag}(\rho_{A1}, \rho_{A2}, \ldots, \rho_{A\hat{m}})$, $1 \geq \rho_{A1} \geq \rho_{A2} \geq \ldots \geq \rho_{A\hat{m}} > 0$, and $\mathbf{D}_B = [\mathbf{0}, \ \mathbf{P}_B] \in \mathbb{R}^{\hat{p}\times\hat{n}}$, $\mathbf{P}_B = \text{diag}(\rho_{B1}, \rho_{B2}, \ldots, \rho_{B\hat{p}})$. $0 < \rho_{B1} \leq \rho_{B2} \leq \ldots \leq \rho_{B\hat{p}} < 1$. In this way, the LSE problem of (5.10) is simplified to:

$$\min_{\Delta\mathbf{z}} \ \|\mathbf{D}_A\Delta\check{\mathbf{z}} - \check{\mathbf{b}}\|_2^2 \quad \text{subject to} \quad \mathbf{D}_B\Delta\check{\mathbf{z}} = \check{\mathbf{d}} \tag{5.12}$$

where $\Delta\check{\mathbf{z}} = \mathbf{Z}_{AB}\Delta\mathbf{z}$, $\check{\mathbf{b}} = \mathbf{U}_A^H\hat{\mathbf{b}}$, and $\check{\mathbf{d}} = \mathbf{U}_B^H\hat{\mathbf{d}}$.

The special diagonal structure of $\mathbf{D}_A$ and $\mathbf{D}_B$ enables an easy computation of the elements $\Delta\check{z}_i$, $i = 1, 2, \ldots, \hat{n}$ of $\Delta\check{\mathbf{z}}$:

$$\Delta\check{z}_i = \begin{cases} \check{b}_i/\rho_{Ai}, & i = 1, 2, \ldots, (\hat{n}-\hat{p}) \\ \check{d}_{(i+\hat{p}-\hat{n})}/\rho_{B(i+\hat{p}-\hat{n})}, & i = \hat{n}-\hat{p}+1, \hat{n}-\hat{p}+2, \ldots, \hat{n} \end{cases} \tag{5.13}$$

Once $\Delta\check{\mathbf{z}}$ is obtained, $\Delta\mathbf{z}$ can be found from $\Delta\mathbf{z} = \mathbf{Z}_{AB}^{-1}\Delta\check{\mathbf{z}}$.

Algorithm 6

Outline of the STLN Algorithm-2 for Hankel structured $[\mathbf{A}_{L\times K} \ \mathbf{b}_{L\times 1}]$

Given. $\mathbf{A}, \mathbf{b}, \mathbf{D}_\alpha, \mathbf{D}_\beta$, *Hankel structure of* $[\mathbf{A}\ \mathbf{b}]$, *tolerance* ϵ.

Step 1: Initialization. *Set* $\alpha = \mathbf{0}$, $\beta_0 = 0$, $\Delta\mathbf{A} = \mathbf{0}$, *compute the LS solution* $\mathbf{y}$ *from* $\min_{\mathbf{y}}\|\mathbf{A}\mathbf{y} - \mathbf{b}\|_2$, $\check{\mathbf{Y}}$ *from* $\mathbf{y}$ *according to (5.3), and* $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{y}$.

Step 2: Iteration. *Repeat*

(a) $\hat{\mathbf{A}} := \begin{bmatrix}\mathbf{D}_\alpha & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0}\end{bmatrix}$, $\Delta\mathbf{z} := \begin{bmatrix}\Delta\alpha\\ \Delta\beta_0\\ \Delta\mathbf{y}\end{bmatrix}$, $\hat{\mathbf{b}} := -\begin{bmatrix}\mathbf{D}_\alpha\alpha\\ \mathbf{D}_\beta\beta_0\end{bmatrix}$,

$\hat{\mathbf{B}} := \begin{bmatrix}(\mathbf{P}_\alpha - \check{\mathbf{Y}}) & \mathbf{P}_\beta & -(\mathbf{A} + \Delta\mathbf{A})\end{bmatrix}$, $\hat{\mathbf{d}} = -\mathbf{r}$.

(b) Use the GSVD method to solve the following LSE problem

$$\min_{\Delta\mathbf{z}} \ \left\|\hat{\mathbf{A}}\Delta\mathbf{z} - \hat{\mathbf{b}}\right\|_2^2 \quad \textit{subject to} \quad \hat{\mathbf{B}}\Delta\mathbf{z} - \hat{\mathbf{d}} = 0$$

(c) $\alpha := \alpha + \Delta\alpha, \beta_0 := \beta_0 + \Delta\beta_0, \mathbf{y} := \mathbf{y} + \Delta\mathbf{y}$

(d) Construct $\Delta\mathbf{A}$ *from* α, $\Delta\mathbf{b}$ *from* α *and* β_0, $\check{\mathbf{Y}}$ *from* $\mathbf{y}$. *Compute* $\mathbf{r} = (\mathbf{b} + \Delta\mathbf{b}) - (\mathbf{A} + \Delta\mathbf{A})\mathbf{y}$.

Until $(\|\Delta\alpha\|, \|\Delta\beta_0\|, \|\Delta\mathbf{y}\|, \|\mathbf{r}\| \leq \epsilon)$ (5.14)

STLN for structured $[\mathbf{A}_{L\times K} \ \mathbf{B}_{L\times(M-K)}]$

The key idea of generalizing the one right-hand-side STLN problem to a multiple-right-hand-side one is to vectorize $\mathbf{A}_{L\times K}\mathbf{Y}_{K\times(M-K)} \approx \mathbf{B}_{L\times(M-K)}$ with $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_{(M-K)}], \mathbf{y}_i \in \mathbb{C}^{K\times 1}$ and $\mathbf{B} = [\mathbf{b}_1, \cdots, \mathbf{b}_{(M-K)}], \mathbf{b}_i \in \mathbb{C}^{L\times 1}$:

$$\begin{bmatrix}\mathbf{A} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \mathbf{A} & \cdots & \mathbf{0}\\ \vdots & & \ddots & \vdots\\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{A}\end{bmatrix}\begin{bmatrix}\mathbf{y}_1\\ \mathbf{y}_2\\ \vdots\\ \mathbf{y}_{(M-K)}\end{bmatrix} \approx \begin{bmatrix}\mathbf{b}_1\\ \mathbf{b}_2\\ \vdots\\ \mathbf{b}_{(M-K)}\end{bmatrix}$$

$$\text{or} \quad \vec{\mathbf{A}}\vec{\mathbf{y}} \approx \vec{\mathbf{b}} \tag{5.15}$$

As detailed in [71], the multiple-right-hand-side STLN problem is then reduced to a one-right-hand-side STLN problem formulated below

$$\min_{\alpha,\beta,\vec{\mathbf{y}}} \ \left\|\begin{bmatrix}\mathbf{D}_\alpha\alpha\\ \mathbf{D}_\beta\beta\end{bmatrix}\right\|_2^2, \quad \text{subject to} \quad \vec{\mathbf{r}}(\alpha,\beta,\vec{\mathbf{y}}) = \mathbf{0}$$

where $\vec{\mathbf{r}}(\alpha,\beta,\vec{\mathbf{y}}) = (\vec{\mathbf{b}} + \Delta\vec{\mathbf{b}}) - (\vec{\mathbf{A}} + \Delta\vec{\mathbf{A}})\vec{\mathbf{y}}$ is the modified residual vector of the set of vectorized linear equations. Hence, the same algorithms described above can be applied.

5.2.2 The STLS algorithm applied to complex Hankel data matrices

The problem of approximating a matrix $\mathbf{X}$, consisting of data x_n, by a reduced-rank matrix $\hat{\mathbf{X}}$, consisting of data $\hat{x}_n$ and with the same structure as $\mathbf{X}$, can also be solved using the following theorem.

Theorem 4 *For a complex data matrix* $\mathbf{X}$, *the solution of the STLS optimization problem (5.2) can be obtained from the triplet* $(\mathbf{u}, \tau, \mathbf{v})$ *of the Riemannian SVD defined below:*

Find the triplet $(\mathbf{u}, \tau, \mathbf{v})$ *corresponding to the minimal* τ *that satisfies*

$$\begin{cases} \mathbf{X}\mathbf{v} = \mathbf{D}_v\mathbf{u}\tau & \mathbf{u}^H\mathbf{D}_v\mathbf{u} = 1 \\ \mathbf{X}^H\mathbf{u} = \mathbf{D}_u\mathbf{v}\tau & \mathbf{v}^H\mathbf{D}_u\mathbf{v} = 1 \end{cases} \tag{5.16}$$

where

- *The elements of the desired rank-reduced matrix* $\hat{\mathbf{X}}$ *are obtained from* $\hat{x}_n = x_n - (\mathbf{u}^H\mathbf{X}_n\mathbf{v})^*\tau, n = 0, 1, \ldots, N-1$, *if* N *fixed matrices* $\mathbf{X}_n \in \mathbb{R}^{L\times M}$ *are chosen to represent the structure of the data matrix such that* $\mathbf{X} = \Sigma_{n=0}^{N-1} x_n\mathbf{X}_n$ *and* $\hat{\mathbf{X}} = \Sigma_{n=0}^{N-1}\hat{x}_n\mathbf{X}_n$.
- *The vector* $\mathbf{y}$ *is given as* $\mathbf{y} = \mathbf{v}/\|\mathbf{v}\|$.
- $\mathbf{D}_u$ *and* $\mathbf{D}_v$ *are positive or non-negative definite matrices defined via*

$$\sum_{n=0}^{N-1}\frac{1}{w_n}\mathbf{X}_n^H(\mathbf{u}^H\mathbf{X}_n\mathbf{v})\mathbf{u} = \mathbf{D}_u\mathbf{v} \text{ and } \sum_{n=0}^{N-1}\frac{1}{w_n}\mathbf{X}_n(\mathbf{u}^H\mathbf{X}_n\mathbf{v})^*\mathbf{v} = \mathbf{D}_v\mathbf{u}$$

respectively.

- *The modified data vector is orthogonal to the correction vector in the sense that* $\hat{\mathbf{x}}^H\mathbf{W}(\mathbf{x} - \hat{\mathbf{x}})$.

This theorem gives a solution for any affinely structured complex data matrix. When the structure is Hankel, say, $\mathbf{X}$ is defined as (2.1), the STLS solution can simply be obtained from

$$\hat{\mathbf{x}} = \mathbf{x} - (\mathbf{u}\otimes\mathbf{v}^*)\ \tau$$
$$\mathbf{D}_u = \check{\mathbf{T}}_u\mathbf{W}^{-1}\check{\mathbf{T}}_u^H \quad \text{and} \quad \mathbf{D}_v = \check{\mathbf{T}}_v\mathbf{W}^{-1}\check{\mathbf{T}}_v^H \tag{5.17}$$

where $\otimes$ denotes convolution, $\check{\mathbf{T}}_u$ and $\check{\mathbf{T}}_v$ are Toeplitz matrices consisting of the elements of $\mathbf{u}$ and $\mathbf{v}$, respectively:

$$\check{\mathbf{T}}_u = \begin{bmatrix} u_1 & u_2 & \cdots & u_L & & 0 \\ & \ddots & & & \ddots & \\ 0 & & u_1 & u_2 & \cdots & u_L \end{bmatrix}$$

The above solutions for a complex data matrix with an affine structure, as well as a special case of Hankel structure, are actually direct extensions of the solutions for real data matrix given in [64].

The STLS algorithm, presented in [64], is based on inverse iteration. For given $\mathbf{D}_u$ and $\mathbf{D}_v$, we perform one step of inverse iteration, using the QR decomposition of $\mathbf{X}$, in order to obtain new estimates of $\mathbf{u}$ and $\mathbf{v}$. Then we update $\mathbf{D}_u$ and $\mathbf{D}_v$ for a new iteration.

Let $\mathbf{X} = [\mathbf{Q}_1 \quad \mathbf{Q}_2]\begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix} = \mathbf{Q}_1\mathbf{R}$, $\mathbf{Q}_1 \in \mathbb{C}^{L\times M}$, $\mathbf{Q}_2 \in \mathbb{C}^{L\times(L-M)}$, $\mathbf{R} \in \mathbb{C}^{M\times M}$. Then we can decompose $\mathbf{u}$ as

$$\mathbf{u} = [\mathbf{Q}_1 \quad \mathbf{Q}_2]\begin{bmatrix}\check{\mathbf{u}}_{Q1}\\ \check{\mathbf{u}}_{Q2}\end{bmatrix} = \mathbf{Q}_1\check{\mathbf{u}}_{Q1} + \mathbf{Q}_2\check{\mathbf{u}}_{Q2}$$

for certain $\check{\mathbf{u}}_{Q1} \in \mathbb{C}^{M\times 1}$, $\check{\mathbf{u}}_{Q2} \in \mathbb{C}^{(L-M)\times 1}$. From $\mathbf{X}^H\mathbf{u} = \mathbf{D}_u\mathbf{v}\tau$, $\mathbf{Q}_2^H(\mathbf{D}_v\mathbf{u}\tau) = \mathbf{Q}_2^H(\mathbf{X}\mathbf{v}) = \mathbf{0}$, $\mathbf{Q}_1^H(\mathbf{D}_v\mathbf{u}\tau - \mathbf{X}\mathbf{v}) = \mathbf{0}$, we find

$$\begin{bmatrix} \mathbf{R}^H & \mathbf{0} & \mathbf{0} \\ \mathbf{Q}_2^H\mathbf{D}_v\mathbf{Q}_1 & \mathbf{Q}_2^H\mathbf{D}_v\mathbf{Q}_2 & \mathbf{0} \\ \mathbf{Q}_1^H\mathbf{D}_v\mathbf{Q}_1 & \mathbf{Q}_1^H\mathbf{D}_v\mathbf{Q}_2 & -\mathbf{R} \end{bmatrix}\begin{bmatrix}\check{\mathbf{u}}_{Q1}\\ \check{\mathbf{u}}_{Q2}\\ \mathbf{v}/\tau\end{bmatrix} = \begin{bmatrix}\mathbf{D}_u\mathbf{v}\tau\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}$$

Solving the above system of linear equations with a block triangular system matrix iteratively results in the following STLS algorithm. Only Hankel structured data matrices are treated here, where $\mathbf{D}_u$ and $\mathbf{D}_v$ are explicitly defined by (5.17). For these structures, the STLS algorithm reduces as follows.

Algorithm 7

Outline of the STLS algorithm for Hankel structured $[\mathbf{A}_{L\times K}\ \mathbf{b}_{L\times 1}]$

Given. $\mathbf{A}, \mathbf{b}, \mathbf{W}$, *structure on* $\mathbf{X}_{L\times M} = [\mathbf{A}_{L\times K}\ \mathbf{b}_{L\times 1}]$, *tolerance* ϵ.

Step 1: Initialization. Choose $\mathbf{u}$ *and* $\mathbf{v}$ *(say* $\mathbf{u} := \mathbf{u}_L$, $\mathbf{v} := \mathbf{v}_M$ *from the SVD of* $\mathbf{X}$*), construct* $\mathbf{D}_v$ *from* $\mathbf{v}$ *according to (5.17), normalize* $\mathbf{u}$ *via* $\mathbf{u} := \mathbf{u}/\sqrt{\mathbf{u}^H\mathbf{D}_v\mathbf{u}}$, *construct* $\mathbf{D}_u$ *from* $\mathbf{u}$ *according to (5.17), and normalize* $\mathbf{v}$ *via* $\mathbf{v} := \mathbf{v}/\sqrt{\mathbf{v}^H\mathbf{D}_u\mathbf{v}}$.

Step 2: Iteration. Repeat

(a) $\check{\mathbf{u}}_{Q1} := \mathbf{R}^{-H}\mathbf{D}_u\mathbf{v}$, $\check{\mathbf{u}}_{Q2} := -(\mathbf{Q}_2^H\mathbf{D}_v\mathbf{Q}_2)^{-1}(\mathbf{Q}_2^H\mathbf{D}_v\mathbf{Q}_1)\check{\mathbf{u}}_{Q1}$,
$\mathbf{u} := \mathbf{Q}_1\check{\mathbf{u}}_{Q1} + \mathbf{Q}_2\check{\mathbf{u}}_{Q2}$, $\mathbf{u} := \mathbf{u}/\sqrt{\mathbf{u}^H\mathbf{D}_v\mathbf{u}}$,
construct $\mathbf{D}_u$ *from* $\mathbf{u}$.

(b) $\mathbf{v} := \mathbf{R}^{-1}\mathbf{Q}_1^H\mathbf{D}_v\mathbf{u}$, $\tau := 1/\sqrt{\mathbf{v}^H\mathbf{D}_u\mathbf{v}}$, $\mathbf{v} := \mathbf{v}\tau$,
construct $\mathbf{D}_v$ *from* $\mathbf{v}$.

Until $(\|\Delta \mathbf{v}\| \leq \epsilon)$

Step 3: Computation of $\hat{x}$ and $\hat{\mathbf{X}}$. Compute $\hat{x}_n$ from $\hat{x}_n = x_n - (\mathbf{u}^H \mathbf{X}_n \mathbf{v})^* \tau$, *and construct $\hat{\mathbf{X}}$ from $\hat{x}_n$.*

5.2.3 Discussion

In contrast to STLS, the STLN algorithm can handle multiple-right-hand-side problems, $\mathbf{AY} \approx \mathbf{B}$, and allows for other norms than $\| \ \|_2$, e.g., $\| \ \|_1$, $\| \ \|_\infty$ (see [69] for details). The following comparisons and discussion in this subsection are made for single-right-hand-side problems, $\mathbf{Ay} \approx \mathbf{b}$.

When $\mathbf{W} = \begin{bmatrix} \mathbf{D}_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta \end{bmatrix}^2$, STLN and STLS actually minimize the same object function but in different ways:

- STLN forces $\mathbf{b} + \Delta\mathbf{b}$ to lie in the column space of $\mathbf{A} + \Delta\mathbf{A}$, while STLS assures rank deficiency of $[\mathbf{A} + \Delta\mathbf{A} \ \ \mathbf{b} + \Delta\mathbf{b}]$.
- STLN imposes structure on $[\Delta\mathbf{A} \ \ \Delta\mathbf{b}]$, while STLS imposes structure on $[\mathbf{A} + \Delta\mathbf{A} \ \ \mathbf{b} + \Delta\mathbf{b}]$.
- STLN uses a constrained optimization approach, while STLS uses a Riemannian SVD approach.

Once convergence is reached, it is proven below that the optimality conditions for the optimization problem (5.1) are satisfied by the two STLN algorithms.

STLN Algorithm-1

In the STLN Algorithm-1, the optimality condition for a local optimum of the penalty function $P(\mathbf{z})$ in (5.6) is the vanishing of the gradients :

$$\frac{\partial P(\mathbf{z})}{\partial \mathbf{z}} = \begin{bmatrix} \frac{\partial P(\mathbf{z})}{\partial \check{\mathbf{P}}_\alpha} \\ \frac{\partial P(\mathbf{z})}{\partial \beta_0} \\ \frac{\partial P(\mathbf{z})}{\partial \mathbf{y}} \end{bmatrix} = \begin{bmatrix} w^2(\mathbf{P}_\alpha - \check{\mathbf{Y}})^T \mathbf{r}^* + \mathbf{D}_\alpha^2 \alpha^* \\ w^2(\mathbf{P}_\beta)^T \mathbf{r}^* + \mathbf{D}_\beta^2 \beta^* \\ -w^2(\mathbf{A} + \Delta\mathbf{A})^T \mathbf{r}^* \end{bmatrix} = \mathbf{0}$$

This is equivalent to the convergence of the LS subproblems in Step 2(a) of STLN Algorithm-1, since the LS solution $\Delta\mathbf{z} = [\Delta\alpha^T, \Delta\beta_0^T, \Delta\mathbf{y}]^T$ in Step 2(a)

satisfies

$$(\mathbf{M}^H\mathbf{M})^{-1} \begin{bmatrix} \Delta\alpha \\ \Delta\beta_0 \\ \Delta\mathbf{y} \end{bmatrix} = -\mathbf{M}^H \begin{bmatrix} w\mathbf{r} \\ \mathbf{D}_\alpha \alpha \\ \mathbf{D}_\beta \beta_0 \end{bmatrix}$$

$$= -\begin{bmatrix} w^2(\mathbf{P}_\alpha - \check{\mathbf{Y}})^H \mathbf{r} + \mathbf{D}_\alpha^2 \alpha \\ w^2(\mathbf{P}_\beta)^H \mathbf{r} + \mathbf{D}_\beta^2 \beta \\ -w^2(\mathbf{A} + \Delta\mathbf{A})^H \mathbf{r} \end{bmatrix} = -\begin{bmatrix} \frac{\partial P(\mathbf{z})}{\partial \check{\mathbf{P}}_\alpha} \\ \frac{\partial P(\mathbf{z})}{\partial \beta_0} \\ \frac{\partial P(\mathbf{z})}{\partial \mathbf{y}} \end{bmatrix}^*$$

where

$$\mathbf{M} = \begin{bmatrix} w(\mathbf{P}_\alpha - \check{\mathbf{Y}}) & w\mathbf{P}_\beta & -w(\mathbf{A} + \Delta\mathbf{A}) \\ \mathbf{D}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0} \end{bmatrix} \tag{5.18}$$

As long as matrix $\mathbf{M}$ has full rank, $\Delta\mathbf{z}$ has a unique solution, which will be zero if, and only if, the right-hand-side gradients are zero.

STLN Algorithm-2

For our problem formulated as (5.10) in STLN Algorithm-2, the optimality condition for a local minimum is the vanishing of the gradients of the Lagrange function $\mathcal{L}(\mathbf{z}) = f(\mathbf{z}) + \lambda^T \mathbf{r}(\mathbf{z})$:

$$\frac{\partial \mathcal{L}(\mathbf{z})}{\partial \mathbf{z}} = 0 \quad \Rightarrow \quad \hat{\mathbf{A}}^T \hat{\mathbf{b}}^* + \hat{\mathbf{B}}^T \lambda = \mathbf{0}$$

$$\frac{\partial \mathcal{L}(\mathbf{z})}{\partial \lambda} = 0 \quad \Rightarrow \quad \hat{\mathbf{B}} \Delta\mathbf{z} - \hat{\mathbf{d}} = \mathbf{0}$$

When the solution $\Delta\check{\mathbf{z}} = \mathbf{0}$ and hence $\Delta\mathbf{z} = \mathbf{0}$ is obtained using the GSVD in Step 2(b) of STLN Algorithm-2, the above two equalities of optimality conditions are satisfied as shown below.

Referring(5.11) and introducing $\mathbf{U}_A = [\mathbf{U}_{A1}, \ \mathbf{U}_{A2}] = [\mathbf{u}_{A1}, \mathbf{u}_{A2}, \ldots, \mathbf{u}_{A\hat{m}}]$, $\mathbf{U}_{A1} \in \mathbb{C}^{\hat{m} \times (\hat{n} - \hat{p})}$, $\mathbf{U}_{A2} \in \mathbb{C}^{\hat{m} \times (\hat{m} + \hat{p} - \hat{n})}$, $\mathbf{u}_{Ai} \in \mathbb{C}^{\hat{m} \times 1}, i = 1, 2, \ldots, \hat{m}$, $\mathbf{U}_B = [\mathbf{u}_{B1}, \mathbf{u}_{B2}, \ldots, \mathbf{u}_{B\hat{p}}]$, $\mathbf{u}_{Bi} \in \mathbb{C}^{\hat{p} \times 1}, i = 1, 2, \ldots, \hat{p}$, and $\mathbf{P}_A = \begin{bmatrix} \mathbf{P}_{A1} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{A2} \end{bmatrix}$, with $\mathbf{P}_{A1} = \text{diag}\,(\rho_{A1}, \rho_{A2}, \ldots, \rho_{A(\hat{n}-\hat{p})})$, $\mathbf{P}_{A2} = \text{diag}\,(\rho_{A(\hat{n}-\hat{p}+1)}, \rho_{A(\hat{n}-\hat{p}+2)}, \ldots, \rho_{A\hat{m}})$. Then the solution of $\Delta\check{\mathbf{z}}$ in (5.13) can be written as

$$\Delta\check{z}_i = \begin{cases} \mathbf{u}_{Ai}^H \hat{\mathbf{b}} / \rho_{Ai}, & i = 1, 2, \ldots, (\hat{n} - \hat{p}) \\ \mathbf{u}_{B(i+\hat{p}-\hat{n})}^H \hat{\mathbf{d}} / \rho_{B(i+\hat{p}-\hat{n})}, & i = \hat{n} - \hat{p} + 1, \hat{n} - \hat{p} + 2, \ldots, \hat{n} \end{cases}$$

This means that

$$\Delta\check{\mathbf{z}} = \begin{bmatrix} \mathbf{P}_{A1}^{-1}\mathbf{U}_{A1}^H\hat{\mathbf{b}} \\ \mathbf{P}_B^{-1}\mathbf{U}_B^H\hat{\mathbf{d}} \end{bmatrix}$$

It follows immediately that

$$\text{if } \Delta\check{\mathbf{z}} = \mathbf{0} \text{ then } \Rightarrow \quad \mathbf{U}_{A1}^H\hat{\mathbf{b}} = \mathbf{0}, \quad \hat{\mathbf{d}} = \mathbf{0}, \text{and } \hat{\mathbf{B}}\Delta\mathbf{z} - \hat{\mathbf{d}} = \mathbf{0} \qquad (5.19)$$

Thus the second equality is proven. In the following, we show that the first equality with the complex Lagrange multipliers vector, $\lambda = -(\hat{\mathbf{b}}^H\mathbf{U}_A\mathbf{D}_A\mathbf{D}_B^\dagger\mathbf{U}_B^H)^T$, is satisfied for $\Delta\check{\mathbf{z}} = \mathbf{0}$:

$$\begin{aligned}
\hat{\mathbf{A}}^T\hat{\mathbf{b}}^* \;+\; & \hat{\mathbf{B}}^T\lambda = \mathbf{Z}_{AB}^T(\mathbf{D}_A^T\mathbf{U}_A^T\hat{\mathbf{b}}^* + \mathbf{D}_B^T\mathbf{U}_B^T\lambda) \\
= \; & \mathbf{Z}_{AB}^T(\mathbf{I}_{\hat{n}\times\hat{n}} - (\mathbf{D}_B^\dagger\mathbf{D}_B)^T)\mathbf{D}_A^T\mathbf{U}_A^T\hat{\mathbf{b}}^* \\
= \; & \mathbf{Z}_{AB}^T\begin{bmatrix} \mathbf{I}_{(\hat{n}-\hat{p})\times(\hat{n}-\hat{p})} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}\begin{bmatrix} \mathbf{P}_{A1} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{A2} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\begin{bmatrix} \mathbf{U}_{A1}^T \\ \mathbf{U}_{A2}^T \end{bmatrix}\hat{\mathbf{b}}^* \\
= \; & \mathbf{Z}_{AB}^T\begin{bmatrix} \mathbf{P}_{A1}\mathbf{U}_{A1}^T\hat{\mathbf{b}}^* \\ \mathbf{0}_{\hat{p}\times 1} \end{bmatrix}
\end{aligned}$$

where † denotes the pseudo-inverse and $\mathbf{D}_B^\dagger = \begin{bmatrix} \mathbf{0} \\ \mathbf{P}_B^{-1} \end{bmatrix}$. Since $\mathbf{U}_{A1}^T\hat{\mathbf{b}}^* = (\mathbf{U}_{A1}^H\hat{\mathbf{b}})^* = \mathbf{0}$ from (5.19), $\frac{\partial\mathcal{L}(\mathbf{z})}{\partial\mathbf{z}} = 0$ is satisfied for the Lagrange multiplier vector $\lambda = -(\hat{\mathbf{b}}^H\mathbf{U}_A\mathbf{D}_A\mathbf{D}_B^\dagger\mathbf{U}_B^H)^T$.

However, the optimization problem (5.1) is not convex. There is no guarantee that these two algorithms converge to a global minimum. In fact, they will converge to the closest local minimum when the residual $\mathbf{r}(\mathbf{z})$ is sufficiently small. In the rest of this thesis, we will mainly use STLN Algorithm-2, because STLN Algorithm-1 is non-automatic (in the choice of w) and less accurate, though faster, than STLN Algorithm-2 due to numerical problems (ill-conditioning when w is large, see Example 2 of Section 5.4.1) [76].

5.3 New parameter estimator HTLN

If we apply the STLN technique to our noise-corrupted data matrix $[\mathbf{A}\ \mathbf{B}]$ in (2.1), we end up with a rank-K Hankel matrix approximation $[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]$ and a solution $\mathbf{Y}$. From $[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]$, a "cleaned-up" signal is obtained, given by the first column and last row of the matrix $[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]$. Obviously, such a "cleaned-up" (though may be deviated) signal can be written

as an exact sum of K exponentially damped sinusoids, or an impulse response of order K. This enables several ways of simplification in the subsequent parameter estimator when "cleaned-up" data are used, as is the case of the MV or Cadzow enhancement when convergence is reached.

Refer to p. 74 in Chapter 4. When the STLN enhancement is fully convergent, one way of increasing efficiency is to reduce the number of data samples used in the pole-estimation algorithm to as small as $2K-1$ without affecting the resulting estimates. In particular, we can simply arrange the $2K-1$ "cleaned-up" data into a matrix $\mathbf{X}$ of size $K \times K$, and compute nonlinear parameters from eigenvalues of the solution $\tilde{\mathbf{Z}}$ in $\mathbf{X}^\dagger = \mathbf{X}_\downarrow\tilde{\mathbf{Z}}$, without performing the SVD.

However, the enhancement is typically *not fully* convergent due to the limit of numerical precision, and the most efficient and accurate estimator, HTLS, is suggested for the enhanced data. Such an STLN-HTLS estimator, unlike cMV-HTLS or cCA-HTLS, can also be simplified and improved in efficiency by using the computed STLN solution $\mathbf{Y}$. This implies that we do not need to run the HTLS algorithm from the beginning after STLN enhancement. In this way, STLN-HTLS reduces to HTLN (stands for Hankel Total Least Norm), as shown below.

Denote the SVD of $\mathbf{X}$ by $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^H, \mathbf{V} = [\mathbf{V}_s, \mathbf{V}_n] = [\mathbf{V}_K, \overline{\mathbf{V}}_K]$, where the subscripts $_s$ and $_n$ denote the signal and null space of $\mathbf{X}$, respectively. Consider the QR decomposition of the STLN solution $\begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix} = [\mathbf{Q}_n\ \ \mathbf{Q}_s]\begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_n\mathbf{R}$, which satisfies $\mathbf{X}\begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix} = \mathbf{0}$. Then, it is easy to see that $\mathbf{Q}_n$ and $\mathbf{V}_n$ span the same null space, while $\mathbf{Q}_s$ and $\mathbf{V}_s$ span the same signal space of $\mathbf{X}$. Therefore, the SVD of $\mathbf{X}$ in the HTLS estimator can be replaced by the QR decomposition of $\begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix}$, to find the orthonormal signal subspace basis with the shift-invariance property.

Algorithm 8

Outline of the HTLN algorithm

Given. A model-order estimate K, and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L\times M$ Hankel matrix $[\mathbf{A}_{L\times K}\ \ \mathbf{B}_{L\times(M-K)}]$ as in (2.1).

Step 1. Apply the STLN algorithm to $[\mathbf{A}\ \mathbf{B}]$ in order to find the Hankel cor-

rection matrix $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$ *and STLN solution* $\mathbf{Y}$ *such that*

$$[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]\begin{bmatrix}\mathbf{Y}\\ -\mathbf{I}\end{bmatrix}=\mathbf{0}$$

Step 2. Compute a QR decomposition of

$$\begin{bmatrix}\mathbf{Y}\\ -\mathbf{I}\end{bmatrix}=[\mathbf{Q}_n\ \mathbf{Q}_s]\begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix}$$

where $[\mathbf{Q}_n\ \mathbf{Q}_s]$ *is unitary, and* $\mathbf{R}$ *is upper triangular.*

Step 3. Compute the TLS solution $\mathbf{Z}^{(Q)}$ *of the following equation:*

$$\mathbf{Q}_s^{*\uparrow}\approx\mathbf{Q}_{s\downarrow}^{*}\mathbf{Z}^{(Q)}$$

and retrieve the eigenvalues z_k *of* $\mathbf{Z}^{(Q)}$*. The* f_k, d_k *are obtained straightforwardly from* z_k*, using the relation* $z_k=e^{(-d_k+j2\pi f_k)/f_s}$.

Step 4. Apply the linear-parameter-estimation part (Step 3) of the HTLS Algorithm 1, to the original data x_n, $n=0,1,\cdots,N-1$, *to find* a_k *and* φ_k, $k=1,\cdots,K$.

The improved efficiency of HTLN, compared to STLN-HTLS, stems from the use of the QR decomposition of $\begin{bmatrix}\mathbf{Y}\\ -\mathbf{I}\end{bmatrix}$, instead of the SVD of $[\mathbf{A}\ \mathbf{B}]$, for finding the orthonormal signal subspace basis. This is more accurate and faster than solving the set of equation $\mathbf{X}^{\uparrow}=\mathbf{X}_{\downarrow}\tilde{\mathbf{Z}}$ mentioned earlier, where $\mathbf{X}_{K\times K}$ is composed of $2K-1$ enhanced data.

Actually, an even more efficient way of finding the poles z_k after application of the STLN algorithm is to use the first column of the STLN solution $\mathbf{Y}$ as LP filter coefficients so that the z_k are simply obtained by rooting the corresponding LP polynomial of order K. This works well when the matrix $[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]$ is perfectly of rank K. When STLN is not fully convergent, e.g., when the tolerance ϵ in the stop criterion of the STLN iterations in (5.9) or (5.14) is not small enough (say, $\epsilon<10^{-8}$ in fitting *Signal-1*), such a rooting method is numerically not as reliable as HTLN. Moreover, the time saving is not significant compared to the computation time of STLN. Therefore we focus on the HTLN algorithm when estimating parameters in this chapter.

5.4 Numerical comparison of different algorithms

5.4.1 Optimality of the STLN, STLS solutions

In this subsection, we use two examples, for which the global minimum can be found analytically, and STLN, STLS converge to the optimal solutions while Cadzow's method does not. These examples arise in the problems of approximating a given data sequence by the impulse response of a finite-dimensional Linear Time-Invariant (LTI) system of a given order, such that the weighted sum of squared residuals is minimized, i.e.,

Approximate a given data sequence $x_n\in\mathbb{C}$ *by* $\hat{x}_n\in\mathbb{C}$, $n=0,1,\ldots,N-1$ *so as to minimize:*

$$\sum_{n=0}^{N-1}|x_n-\hat{x}_n|^2w_n\quad\text{subject to}\quad\hat{\mathbf{X}}_{L\times M}\begin{bmatrix}\mathbf{Y}_{K\times(M-K)}\\ -\mathbf{I}_{(M-K)\times(M-K)}\end{bmatrix}=\mathbf{0}\qquad(5.20)$$

where $\hat{\mathbf{X}}$ is an $L\times M$ Hankel matrix, $L\geq M$, constructed from the elements $\hat{x}_n$, and w_n are appropriate weights. The constraint imposing that $\hat{\mathbf{X}}$ has at most rank K ensures that $\hat{x}_n$ is the impulse response of a finite-dimensional linear system of order K at most. If the sequence x_n is itself the impulse response of a higher dimensional system, our problem corresponds to model reduction. If the sequence x_n is a given data sequence which is not an impulse response, but for instance, a noise-corrupted one, one might consider this problem as a noisy realization problem, or a signal enhancement problem. If additionally all $w_n=1$, we get model reduction in the L_2 norm.

Suppose the weights w_n account for the repetition of elements $\hat{x}_n$ in the Hankel matrix $\hat{\mathbf{X}}$, then

$$\begin{aligned}\mathbf{W}&=\operatorname{diag}(w_0,w_1,\ldots,w_{N-1})\\ &=\begin{bmatrix}\mathbf{D}_\alpha^2&\mathbf{0}\\ \mathbf{0}&\mathbf{D}_\beta^2\end{bmatrix}=\operatorname{diag}(1,2,\ldots,M,\ldots,M,\ldots,2,1)\end{aligned}$$

and (5.20) reduces to a STLN optimization problem with $(M-K)$ right-hand-sides as we have discussed earlier. If $M=K+1$, the problem can also be solved by the STLS algorithm presented in the previous section.

As proven in [64], there is one special case for which the global optimum of (5.20) can be found analytically, i.e.

Approximate a given data sequence $x_n\in\mathbb{C}$ *by a first order LTI impulse response, which can be parameterized as* $\hat{x}_n=\gamma\eta^n\in\mathbb{C}$, $n=0,1,\ldots,N-1$.

Since a Hankel matrix $\hat{\mathbf{X}}_{L\times M}$, composed of the elements $\hat{x}_n=\gamma\eta^n$, is always a rank-1 matrix, we obtain a minimization problem without the rank

deficiency constraint:

$$\min_{\gamma,\eta} \sum_{n=0}^{N-1} |x_n - \gamma\eta^n|^2 w_n = \min_{\gamma,\eta} f(\gamma,\eta) \tag{5.21}$$

The global minimum of (5.21) can be obtained by setting the derivatives with respect to γ and η to zero:

$$\frac{\partial f}{\partial \gamma} = 0 \quad \Rightarrow \quad \sum_{n=0}^{N-1} w_n(x_n - \gamma\eta^n)(-\eta^n) = 0,$$

$$\frac{\partial f}{\partial \eta} = 0 \quad \Rightarrow \quad \sum_{n=1}^{N-1} w_n(x_n - \gamma\eta^n)(-n\eta^{n-1}) = 0.$$

Eliminating γ from the first equation and substituting it in the second, we get the polynomial in η:

$$\left(\sum_{n=1}^{N-1} n w_n x_n \eta^{n-1}\right)\left(\sum_{n=0}^{N-1} w_n \eta^{2n}\right) - \left(\sum_{n=0}^{N-1} w_n x_n \eta^n\right)\left(\sum_{n=1}^{N-1} n w_n \eta^{2n-1}\right) = 0 \tag{5.22}$$

This is a polynomial of degree $3(N-1)-2$ in η. One has to select the root which gives the minimum in (5.21).

Example 1

Consider the same data sequence $\mathbf{x} = [6\ 5\ 4\ 3\ 2\ 1]^T$, as given in [64], which can be modeled as the impulse response of the following second order LTI system:

$$s_{n+1} = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} s_n \quad \text{with initial state} \quad s_0 = \begin{bmatrix} 6 \\ 1 \end{bmatrix}$$

$$x_n = [1\ 0]\, s_n$$

We arrange the data $x_n, n = 0, 1, \ldots, N-1$ in the $L \times M = 5 \times 2$ Hankel matrix $\mathbf{X}$, and find the best rank-1 structure-preserving approximation $\hat{\mathbf{X}}$ to $\mathbf{X}$

$$\mathbf{X} = \begin{bmatrix} x_0 & x_1 \\ x_1 & x_2 \\ x_2 & x_3 \\ x_3 & x_4 \\ x_4 & x_5 \end{bmatrix} = \begin{bmatrix} 6 & 5 \\ 5 & 4 \\ 4 & 3 \\ 3 & 2 \\ 2 & 1 \end{bmatrix}, \quad \hat{\mathbf{X}} = \begin{bmatrix} \hat{x}_0 & \hat{x}_1 \\ \hat{x}_1 & \hat{x}_2 \\ \hat{x}_2 & \hat{x}_3 \\ \hat{x}_3 & \hat{x}_4 \\ \hat{x}_4 & \hat{x}_5 \end{bmatrix}$$

method	$w_n = 1,2,2,2,2,1$		
	# iterations	# flops	η
Rooting polynomial			0.7629230150743218
STLN Algorithm-1	14	108358	0.7629230150743225
STLN Algorithm-2	14	472887	0.7629230150743224
STLS	9	17202	0.7629230150743225
Cadzow	87	34365	0.7627806087028032
method	$w_n = 1,1,1,1,1,1$		
	# iterations	# flops	η
Rooting polynomial			0.7602263542172322
STLN Algorithm-1	15	116159	0.7602263542172321
STLN Algorithm-2	14	417775	0.7602263542172322
STLS	8	15309	0.7602263542172323

Table 5.1: Example 1: number of iterations, number of flops and values of η at convergence. The tolerance is $\epsilon = 10^{-15}$ for all the methods. The best weight, after several tests, is used in STLN Algorithm-1, which is $w = 10^{10}$ in this case.

by solving the corresponding STLN problem with weighting, $w_n = 1,2,2,2,2,1$, and without weighting, $w_n = 1,1,1,1,1,1$, for $n = 0,1,2,3,4,5$. The results of computation are listed in Table 5.1. The true values of η which give the minimum in (5.21) are obtained by finding the roots of (5.22) with $L = 5, K = 1, M = 2$ using MATLAB version 4.2a. The same minima are also obtained by the STLN Algorithm-1, STLN Algorithm-2 and the STLS algorithm, up to 14 digits, from each ratio $\hat{x}_{n+1}/\hat{x}_n \quad n = 0,1,\ldots,4$. This shows that the algorithms of STLN and STLS converge to the optimal solution in this example, though both STLN Algorithms, especially STLN Algorithm-2, needs large number of flops. On the other hand, the suboptimal methods such as Cadzow's algorithm does not converge to the true solution.

Example 2

Consider the following fourth order LTI system:

$$s_{n+1} = \begin{bmatrix} 0.4 & 0 & 0 & 0 \\ 0 & 0.3 & 0 & 0 \\ 0 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} s_n \quad \text{with initial state} \quad s_0 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

method	$w_n = 1,2,3,4,4,3,2,1$		
	# iterations	# flops	η
Rooting polynomial			0.2602566142908348
STLN Algorithm-1[a]	58	1867699	0.2602566142908066
STLN Algorithm-2	10	1434083	0.2602566142908349
Cadzow	82	116166	0.2602455147443435
methods	$w_n = 1,1,1,1,1,1,1$		
	# iterations	# flops	η
Rooting polynomial			0.2568639874063329
STLN Algorithm-1[b]	103	3320939	0.2568639874063330
STLN Algorithm-2	8	1064917	0.2568639874063330

[a] The best weight is $w = 10^6$. A larger w causes numerical problems.
[b] The best weight is $w = 10^7$. A larger w causes numerical problems.

Table 5.2: Example 2: number of iterations, number of flops and values of η at convergence. The tolerance is $\epsilon = 10^{-15}$ for both algorithms.

$$x_n = [1\ 1\ 1\ 1]\, s_n$$

Arrange the first 8 samples of the impulse response, $\mathbf{x} = [4\ \ 1\ \ 0.3$ $0.1\ \ 0.0354\ \ 0.013\ \ 0.00489\ \ 0.00187]^T$, in an $L \times M = 5 \times 4$ Hankel matrix $\mathbf{X}$ as done in Example 1. Table 5.2 shows the results of computing the best rank-1 structure-preserving approximation $\hat{\mathbf{X}}$ to $\mathbf{X}$ using different methods. Again, STLN Algorithm-2 reaches the known optimal solution for η up to at least 14 digits, while Cadzow's approach converges to suboptimal solutions. Since this problem has $M - K = 3$ right-hand-sides, the STLS method, which only reduce the rank of $\mathbf{X}$ by one, can not be considered here.

Note that when w is large and the number of right-hand-sides increases, there tend to be more ill-conditioning or non-convergent problems with STLN Algorithm-1. Typically, ill-conditioning occurs when $\mathbf{M}$ in (5.18) becomes rank deficient. A lower weight, w, can reduce such problems, but it leads to inaccurate results. Therefore, *we stick to STLN Algorithm-2 in the rest of this chapter.*

Similar results have been obtained for various other sequences x_n and other weights w_n for the problem (5.21).

5.4.2 Simulation results

In this section, several enhancement algorithms, Cadzow's, MV, STLS, STLN, and the corresponding enhanced parameter estimators CA-HTLS, MV-HTLS, STLS-HTLS, and HTLN are compared in terms of resolution, accuracy, and computational efficiency, based on computer simulations.

Simulation procedure

The complex signal, *Signal-1*, with $N = 25$ data points is processed by different algorithms. The maximum number of iterations of the iterative enhancement methods is set to 100. The tolerance ϵ in the stop criterion of iterative methods is $\epsilon = 10^{-10}$. If not specified, the iterative methods converge within 100 iterations. Different techniques such as CA, MV, STLS, and STLN, which solve the incompatible set $\mathbf{AY} \approx \mathbf{B}$ by adding different Hankel correction matrices $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$ to the noisy, full-rank Hankel matrix $[\mathbf{A}\ \ \mathbf{B}]$ so that the set $(\mathbf{A} + \Delta\mathbf{A})\mathbf{Y} = (\mathbf{B} + \Delta\mathbf{B})$ is exactly solvable, are first applied to signal enhancement and compared in Frobenius norm $\|[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]\|_F$ averaged over 200 noise realizations. Finally, the performance of different enhanced estimation methods CA-HTLS, MV-HTLS, STLS-HTLS, and HTLN are studied, in terms of respectively resolution and parameter accuracy, by comparing the number of failures and the RMSE. The size $L \times M$ of the Hankel data matrix used in Cadzow's enhancement procedure and the HTLS estimator is 13×13, since a square matrix produces the best results for HTLS and Cadzow's method takes the same dimension as that of the applied estimator. The size $L \times M$ used in the MV, STLS and STLN enhancement procedures is 23×3, unless specified otherwise. STLN Algorithm-2 is used in STLN and HTLN methods.

Enhancement results

Fig. 5.1 shows the spectrum of a noisy ($\sigma_\nu = 0.1257$ or SNR=15dB) simulated *signal-1*, the signal enhanced by STLN, and its residue. Here, $N = 81$ is chosen for both STLN enhancement and FFT spectrum so that a nice resolution in the spectrum is achieved. Clearly, the STLN-enhanced signal looks smooth and "clean". The signal spectra enhanced by other methods such as CA, MV and STLS are similar and hence not shown. The difference between the considered enhancement methods is not clear from their residue spectra.

A better comparison is obtained by comparing the corrections applied by each method to the data matrices or data vectors when solving $\mathbf{AY} \approx \mathbf{B}$. Table 5.3 compares the corrections applied by the enhancement methods studied, as

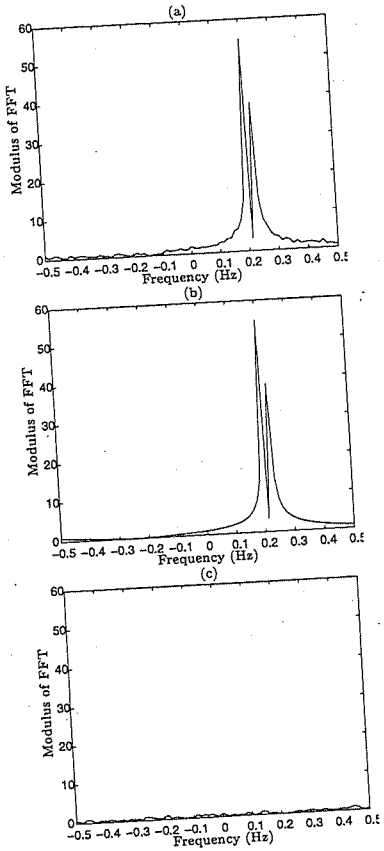


Figure 5.1: $N = 81$, SNR=15dB, $\sigma_\nu = 0.1257$, $M = K + 1$. (a) Original noisy signal spectrum; (b) STLN-enhanced signal spectrum; (c) residual spectrum.

SNR(dB)	method	M	# iteration[a]	corr_M	corr_v
25	TRUE[b]	3		$4.627e-01$	$2.781e-01$
	Cadzow	13	32	$6.693e-01$	$2.628e-01$
	MV[c]	3	100	$4.087e-01$	$2.476e-01$
	STLS	3	14	$4.272e-01$	$2.581e-01$
	STLN	3	10	$4.272e-01$	$2.581e-01$
	LS	3		$4.297e-01$	
	TLS	3		$2.650e-01$	
25	TRUE	4		$5.218e-01$	$2.792e-01$
	MV	4	100	$4.758e-01$	$2.563e-01$
	STLN	4	17	$4.815e-01$	$2.593e-01$
	LS	4		$6.837e-01$	
	TLS	4		$3.651e-01$	

[a] Averaged number of iterations over convergent cases.
[b] True correction or simulated noise.
[c] In all 200 runs, the MV method does not converge within 100 iterations.

Table 5.3: Expected (averaged over 200 runs) Frobenius norm, corr_M and corr_v, of correction matrices and correction vectors given by the first column and the last row of Hankel data matrices.

well as those of other techniques such as LS and TLS that solves $\mathbf{AY} \approx \mathbf{B}$. In all runs, the MV method did not converge within 100 iterations. All other iterative methods converge within 100 iterations, if not indicated otherwise. In Table 5.3, corr_M is the expected Frobenius norm of the correction matrix $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$. If $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$ is Hankel, then corr_v represents the expected Frobenius norm of the corresponding correction vector given by the first column and the last row of $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$. Note that the corrections of the TLS method are always minimal, which can be easily proven. The corrections applied by STLN and STLS are about the same, larger than the correction applied by TLS which does not take the Hankel structure into account but lower than the true correction (or simulated noise) and the LS correction. Although MV has lower corrections, it does not successfully reduce the rank of the Hankel matrix $[\mathbf{A}\ \ \mathbf{B}]$ to $K = 2$ since it fails to converge within 100 iterations. Hence the correction applied by STLN and STLS is the lowest among the methods that preserve the Hankel structure and reduce the rank to K.

Parameter estimation results

All enhancement methods give very smooth signals when convergence occurs. However, as pointed out in Chapter 4, the convergence of the MV and Cadzow's algorithms is redundant when followed by an SVD-based estimator, such as HTLS, since the first iteration is the most decisive one. Therefore, the MV and Cadzow's method perform only one iteration when used to enhance the parameter estimates in the following simulation.

In STLS enhancement, the matrix size can only be chosen as $L \times M = 23 \times 3$ here for *Signal-1* due to the restriction of $M = K + 1$. In MV and STLN enhancement, more choices of the matrix size is possible, as long as $M \geq K + 1 = 3$. However, simulation results show that the MV enhancement gives comparable resolution and accuracy for *Signal-1* when M is equal to or a bit larger than $K+1$. However, the STLN enhancement performs best for this signal at $M = K + 1$. The reason is perhaps the larger number of constraints $\mathbf{r}(\mathbf{z})$ than the number of the unknowns $\mathbf{z}$ in (5.5) when $M - K$ is larger than 1 for this signal. Therefore, we will use the same size, $L \times M = 23 \times 3$, in the STLS, STLN and MV enhancement.

Table 5.4 (a) and (b) list the results obtained with the different methods. At SNR's as high as 25 dB, no failure in convergence or failure in resolving the two peaks has been observed. HTLN and STLS-HTLS produce almost the same results. These methods obtain lower RMSE values and hence are more accurate than HTLS, CA-HTLS and MV-HTLS. At SNR=20 dB, the iterative methods HTLN and STLS-HTLS still converge well, typically within 30 iterations. But there are some failures for all the methods. The iterative methods have fewer failures than the non-iterative methods HTLS, CA-HTLS and MV-HTLS, and show lower RMSE results. When the SNR is lowered to 15 dB, there are several runs in which HTLN and STLS-HTLS fail to converge within 100 iterations. Among the runs in which convergence occurred, there are still failures. At this SNR, MV-HTLS can better resolve the two interfering peaks since it produces the least number of failures. The RMSE of MV-HTLS and the iterative methods are comparable, all of which are lower than HTLS and CA-HTLS. Looking at the bias and standard deviation, we can see that HTLN and STLS-HTLS reduce the bias for d_1 with a factor 2 to 4 compared to HTLS at all the SNR's. In addition, HTLN and STLS-HTLS have lower standard deviation for both f_1 and d_1 at almost all the SNR's.

Computational efficiency

Although HTLN and STLS-HTLS are more accurate than other methods,

SNR	method	nc[a]	fail.	RMSE f_1	Bias±std. f_1
25dB	HTLS		0	0.00239	-0.00022 ± 0.00239
	CA-HTLS		0	0.00228	-0.00014 ± 0.00228
	MV-HTLS		0	0.00220	-0.00005 ± 0.00221
	STLS-HTLS	0	0	0.00208	-0.00013 ± 0.00208
	HTLN	0	0	0.00208	-0.00013 ± 0.00208
20dB	HTLS		11	0.00392	-0.00050 ± 0.00389
	CA-HTLS		9	0.00374	-0.00033 ± 0.00374
	MV-HTLS		11	0.00352	-0.00028 ± 0.00352
	STLS-HTLS	0	3	0.00353	-0.00060 ± 0.00348
	HTLN	0	3	0.00355	-0.00056 ± 0.00352
15dB	HTLS		96	0.00506	-0.00027 ± 0.00508
	CA-HTLS		62	0.00491	-0.00047 ± 0.00491
	MV-HTLS		46	0.00464	-0.00033 ± 0.00464
	STLS-HTLS	12	71	0.00427	-0.00062 ± 0.00425
	HTLN	2	86	0.00453	-0.00043 ± 0.00453

[a] Total number of runs, out of 200, in which the iterative method fails to converge after 100 iterations.

Table 5.4: (a): Number of non-convergence and failures, RMSE, bias and standard deviation of f_1.

they are less efficient as can be seen from the number of flops in Table 5.5. Therefore, one must trade off between the efficiency and accuracy requirement when choosing among optimal and suboptimal, iterative and non-iterative methods. The computational cost of these methods gets larger as the number of data samples N is larger or the SNR is lower. Moreover, the memory requirement gets larger when N is increased. This hampers the practical application of these methods in NMR spectroscopy.

5.5 Conclusion

In this chapter, the STLN and STLS algorithms are studied and applied to signal enhancement and parameter estimation of complex exponentially damped sinusoids. As a matter of fact, STLN and STLS can be applied in many other applications, wherever TLS is applied to structured problems and ML estimates are desired.

SNR	method	RMSE d_1	Bias±std. d_1
25dB	HTLS	0.01489	-0.00214 ± 0.01477
	CA-HTLS	0.01431	-0.00154 ± 0.01426
	MV-HTLS	0.01360	0.00022 ± 0.01363
	STLS-HTLS	0.01252	-0.00055 ± 0.01254
	HTLN	0.01252	-0.00055 ± 0.01254
20dB	HTLS	0.02714	0.00086 ± 0.02720
	CA-HTLS	0.02877	-0.00053 ± 0.02884
	MV-HTLS	0.02357	0.00390 ± 0.02330
	STLS-HTLS	0.02420	0.00045 ± 0.02426
	HTLN	0.02414	0.00046 ± 0.02419
15dB	HTLS	0.05980	-0.00411 ± 0.05995
	CA-HTLS	0.05517	-0.00239 ± 0.05532
	MV-HTLS	0.04255	0.00469 ± 0.04243
	STLS-HTLS	0.04157	0.00000 ± 0.04175
	HTLN	0.04550	0.00185 ± 0.04567

Table 5.4: (cont.) (b): RMSE, bias and standard deviation of d_1.

These two algorithms are closely related in that for single-right-hand-side problems, $\mathbf{Ay} \approx \mathbf{b}$, they minimize the same object function but in different ways

- STLN imposes structure on $[\Delta\mathbf{A} \;\; \Delta\mathbf{b}]$, while STLS imposes structure on $[\mathbf{A} + \Delta\mathbf{A} \;\; \mathbf{b} + \Delta\mathbf{b}]$.
- STLN uses a constrained optimization approach, while STLS transforms the problem to the computation of a Riemannian SVD via an inverse iteration based algorithm.
- Both STLN and STLS achieve optimal solutions.
- The results of enhancement by STLN are similar to those of STLS, both remove corruption noise effectively and improve resolution and parameter accuracy, especially, the standard deviations of the estimated parameters.

Comparison of the STLN and STLS algorithms with Cadzow's algorithms shows that the STLN and STLS algorithms can provide optimal solutions while Cadzow's algorithms can not. This partly explains the better accuracy performance in our parameter estimation, especially the lower bias and standard

methods	SNR=25dB	SNR=20dB	SNR=15dB
HTLS	1.19e+05	1.21e+05	1.23e+05
CA-HTLS	2.23e+05	2.25e+05	2.27e+05
MV-HTLS	1.24e+05	1.24e+05	1.25e+05
STLS-HTLS	8.43e+06	1.38e+07	2.60e+07
HTLN	4.86e+07	6.55e+07	1.15e+08

Table 5.5: Number of flops required by different methods using Matlab averaged over 200 runs at SNR=25dB, 20dB, and 15dB.

deviation of the estimated parameters by HTLN and STLS-HTLS. However, similar to Cadzow's algorithms, STLN and STLS also introduce colored noise and the obtained K exponentials are not the exact noise-free signals but deviated ones.

A common problem with STLN and STLS is that they are computationally expensive. A more efficient STLN algorithm has been proposed in [77] based on the Karush-Kuhn-Tucker equations, which is merely Newton's method applied to the Lagrange function of our optimization problem with nonlinear constraints. It is implemented using the sequential quadratic programming approach and line search [73]. In fact, STLN Algorithm2 is a special case of this new algorithm when the Hessian of constraints is not taken into account, in which case the object functions and constraints of the two algorithms are the same. By including the second order constraint terms, which is important for obtaining superlinear convergence rate for nonlinear constraints [73], the new method achieves better convergence rate at similar accuracy. Nevertheless, this is still not sufficient for processing real-world NMR spectroscopic signals, which have typically large number of data samples and low SNR's. Further work needs to be done in decreasing the computational cost of the STLN and STLS algorithms, which is crucial for real-world applications.

Chapter 6

Incorporation of prior knowledge of known signal poles

Up till now, we have presented several prefiltered and enhanced parameter estimation methods, which improve the resolution and parameter accuracy to different degree. These new methods are based on the *techniques* of filtering, minimum variance estimation, STLN and STLS, etc. Intuitively, if some *prior knowledge* is available and incorporated, the resolution and accuracy of the estimates can be further improved. For example, in NMR spectroscopy and other applications, some of the signal poles might be known *a priori*. The formulation of such prior knowledge and its incorporation into the TLS-FLP method, briefly mentioned in Section 2.2 of Chapter 2, has been described by Dowling et al. in [5][79], where the known signal poles are incorporated via linear constraints. The resulting method, using the linearly constrained TLS (LCTLS) technique and called LCTLS-FLP, produces better results than the TLS-FLP method, especially when the known signal poles are close to the unknown ones in the spectrum. In this chapter, we incorporate the same prior knowledge into more accurate estimators, such as TLS-BLP, HTLS and HTLN, and show that the resulting methods, LCTLS-BLP, HTLS(PK) and HTLN(PK), outperform the LCTLS-FLP method.

The rest of this chapter is arranged as follows. Section 1 reviews briefly some of the above-mentioned parameter estimators. Section 2 describes the corresponding prior-knowledge-incorporating estimators. In Section 3, results

of computer simulations using different estimators are presented. Conclusions are given in Section 4. More analysis and applications of the most appropriate estimator, HTLS(PK), will be given in the next chapter.

6.1 Brief review of some parameter estimators

Before incorporating prior knowledge of known signal poles into parameter estimators like TLS-FLP, TLS-BLP, HTLS and HTLN, we first review the theory of these estimation methods for clarity of exposition. Since the HTLS and HTLN algorithms have been described in Chapter 2 and 5 respectively, we only introduce the TLS-FLP and TLS-BLP algorithms, i.e., the forward and backward linear prediction methods based on the TLS technique.

Assume that an $L \times M$ Hankel matrix $\mathbf{X}$, defined in (2.1), consisting of N uniformly sampled data points $x_n, n = 0, 1, \ldots, N-1$, and a model-order estimate K are given.

TLS-FLP [20][25]

Let $[\mathbf{X}^{(FLP)}_{L\times(M-1)} \quad \mathbf{x}^{(FLP)}_{L\times 1}] = \mathbf{X}, \qquad L \geq K, M \geq K+1, N = L+M-1$

where $\mathbf{X}^{(FLP)}$ consists of the first $M-1$ columns of $\mathbf{X}$ and $\mathbf{x}^{(FLP)}$ is the last column of $\mathbf{X}$.

If the data are noise-free, then the following forward prediction equations are satisfied:

$$\mathbf{X}^{(FLP)}\mathbf{b}^{(FLP)} = \mathbf{x}^{(FLP)} \quad \text{or} \quad \mathbf{X}\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} = \mathbf{0}$$

where $\mathbf{b}^{(FLP)} = [b^{(FLP)}_{M-1}, b^{(FLP)}_{M-2}, \ldots, b^{(FLP)}_1]^T$ contains the forward prediction coefficients, and $z_k, k = 1, 2, \cdots, K$ are the K signal zeros (see below) out of the $M-1$ zeros of the prediction-error filter polynomial

$$B^{(FLP)}(z) = 1 - b^{(FLP)}_1 z^{-1} - \cdots - b^{(FLP)}_{M-1} z^{-(M-1)} \tag{6.1}$$

For noisy data, the exact solution $\mathbf{b}^{(FLP)}$ to $\mathbf{X}^{(FLP)}\mathbf{b}^{(FLP)} = \mathbf{x}^{(FLP)}$ does not exist and must be estimated. The computation of $\mathbf{b}^{(FLP)}$ is based on the minimum norm TLS solution (see e.g. [24] p. 59) of the set $\mathbf{X}\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} \approx \mathbf{0}$

and is derived from the SVD of $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$ in the following way:

$$\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} = \frac{-\overline{\mathbf{V}}_K \overline{\mathbf{V}}_K^H \mathbf{e}_M}{\mathbf{e}_M^T \overline{\mathbf{V}}_K \overline{\mathbf{V}}_K^H \mathbf{e}_M} \tag{6.2}$$

where $\overline{\mathbf{V}}_K$ is the basis of the orthogonal complement of the space spanned by the first K right singular vectors $\mathbf{V}_K$ as defined in Chapter 2, i.e., $\mathbf{V} = [\mathbf{V}_K \ \ \overline{\mathbf{V}}_K]$, and $\mathbf{e}_M = [0, 0, \ldots, 0, 1]^T \in \mathbb{R}^{M\times 1}$. Note that if $K < M - K$ it is more efficient to work with $\mathbf{V}_K$ via the replacement $\overline{\mathbf{V}}_K \overline{\mathbf{V}}_K^H = \mathbf{I} - \mathbf{V}_K \mathbf{V}_K^H$.

Once the $M-1$ poles are determined from the $M-1$ zeros of the forward prediction error filter polynomial (6.1), the K true poles are separated from the remaining $M-1-K$ extraneous poles. A popular approach for pole selection is to choose K poles based on their location with respect to the unit circle. For example, one can choose the K poles closest to the unit circle if it is known that the poles are undamped or the K poles with smallest moduli if it is known that the poles are damped. However, a more reliable way is to classify poles as true or extraneous based on the energy of the corresponding mode, as used in our computer simulation in Section 6.3.

TLS-BLP [20][21][25]

$$\text{Let} \quad [\mathbf{x}_{L\times 1}^{(BLP)} \ \ \mathbf{X}_{L\times(M-1)}^{(BLP)}] = \mathbf{X}^*, \qquad L \geq K, M \geq K+1, N = L+M-1 \tag{6.3}$$

where $\mathbf{x}_{L\times 1}^{(BLP)}$ is the first column of $\mathbf{X}^*$ and $\mathbf{X}_{L\times(M-1)}^{(BLP)}$ consists of the last $M-1$ columns of $\mathbf{X}^*$.

If the data are noise-free, then the following backward prediction equations are satisfied:

$$\mathbf{X}^{(BLP)}\mathbf{b}^{(BLP)} = \mathbf{x}^{(BLP)} \quad \text{or} \quad [\mathbf{X}^{(BLP)} \ \ \mathbf{x}^{(BLP)}] \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} = \mathbf{0}$$

where $\mathbf{b}^{(BLP)} = [b_1^{(BLP)}, b_2^{(BLP)}, \ldots, b_{M-1}^{(BLP)}]^T$ contains the backward prediction coefficients, and $(z_k^*)^{-1} = \exp[d_k \Delta t + j(2\pi f_k \Delta t)]$, $k = 1, 2, \cdots, K$ are the K signal zeros out of the $M-1$ zeros of the prediction-error filter polynomial

$$B^{(BLP)}(z) = 1 - b_1^{(BLP)} z^{-1} - \cdots - b_{M-1}^{(BLP)} z^{-(M-1)} \tag{6.4}$$

The inverse in $(z_k^*)^{-1}$ comes from the reversion of prediction coefficients and the conjugation $*$ in $(z_k^*)^{-1}$ comes from the conjugation of data. Note that

$(z_k^*)^{-1} = \exp[d_k \Delta t + j(2\pi f_k \Delta t)]$ is different from z_k only in the sign of the damping factor d_k.

For noisy data, the computation of $\mathbf{b}^{(BLP)}$ is similar to that of $\mathbf{b}^{(FLP)}$ in TLS-FLP, that is, by taking the minimum norm TLS solution of the set $[\mathbf{X}^{(BLP)} \ \ \mathbf{x}^{(BLP)}] \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} \approx \mathbf{0}$.

Once the $M-1$ poles are determined from the $M-1$ zeros of the backward prediction error filter polynomial (6.4), the poles with negative damping factors are eliminated and the true poles are separated from the extraneous poles by selecting the K peaks with the largest peak energy.

6.2 Prior-knowledge-incorporating estimators

In this section, the incorporation of known signal poles into the TLS-FLP method, known as the LCTLS-FLP method, is first reviewed [79]. Incorporating the same prior knowledge into the other parameter estimators described in Section 2 results in new and better estimators, LCTLS-BLP, HTLS(PK) and HTLN(PK), which are presented thereafter. The abbreviation of PK stands for prior knowledge of known signal poles. Note that the LCTLS-FLP method will be derived in a different way, compared to [5][79], in order to motivate the HTLS(PK) method.

Without loss of generality, we index the p known poles as the first p poles, that is, $z_k, k = 1, \ldots, p$ are known and $z_k, k = p+1, \ldots, K$ are unknown.

LCTLS-FLP [5]

Since $z_k, k = 1, \ldots, p$ are *a priori* exactly known signal poles, they are also the zeros of the forward prediction-error filter polynomial in (6.1), which means

$$z^{M-1} B^{(FLP)}(z)|_{z=z_k} = 0, \quad k = 1, \ldots, p$$

Stacking these p equations together results in a set of p linear constraints which expresses the assumed prior knowledge in terms of known signal poles and forward prediction coefficients:

$$\begin{bmatrix} 1 & z_1 & \cdots & z_1^{M-2} \\ 1 & z_2 & \cdots & z_2^{M-2} \\ \vdots & & & \vdots \\ 1 & z_p & \cdots & z_p^{M-2} \end{bmatrix} \begin{bmatrix} b_{M-1}^{(FLP)} \\ b_{M-2}^{(FLP)} \\ \vdots \\ b_1^{(FLP)} \end{bmatrix} = \begin{bmatrix} z_1^{M-1} \\ z_2^{M-1} \\ \vdots \\ z_p^{M-1} \end{bmatrix}$$

or

$$\mathbf{T}_p^T \begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} = \mathbf{0} \tag{6.5}$$

where $\mathbf{T}_p \in \mathbb{C}^{M\times p}$ is a Vandermonde matrix as defined in (2.7) but composed of only the first p known signal poles.

The incorporation of this prior knowledge into the TLS-FLP method is accomplished by using the above equations as linear constraints. Therefore, the TLS solution $\mathbf{b}^{(FLP)}$ of $\mathbf{X}^{(FLP)}\mathbf{b}^{(FLP)} \approx \mathbf{x}^{(FLP)}$ in the TLS-FLP algorithm is replaced by the linearly constrained TLS, or LCTLS, solution of:

$$\begin{aligned} \mathbf{X}\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} &\approx \mathbf{0} \\ \text{subject to } \mathbf{T}_p^T\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} &= \mathbf{0} \end{aligned}$$

resulting in the LCTLS-FLP algorithm.

The outline of the LCTLS-FLP algorithm is as follows. First perform a QR decomposition of $\mathbf{T}_p = [\mathbf{Q}_1 \;\; \mathbf{Q}_2]\begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}$, where $\mathbf{R} \in \mathbb{C}^{p\times p}$ is upper triangular, to find an orthonormal basis $\mathbf{Q}_2 \in \mathbb{C}^{M\times(M-p)}$ of the null space of $\mathbf{T}_p^H$, such that $\mathbf{T}_p^H\mathbf{Q}_2 = \mathbf{0}$. Let $\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix}^T = \gamma\mathbf{Q}_2^*\hat{\mathbf{b}}^{(FLP)}$, where γ is a nonzero scaling factor that forces the last element to be -1 and $\hat{\mathbf{b}}^{(FLP)}$ is a column vector of dimension $M-p$. Then the constraints hold if and only if $[\mathbf{b}^{(BLP)T} \;\; -1]^T = \gamma\hat{\mathbf{Q}}_2\hat{\mathbf{b}}^{(BLP)}$ for some $\hat{\mathbf{b}}^{(BLP)} \in \mathbb{C}^{(M-p)\times 1}$. In this way, the LCTLS problem is reduced to a classical TLS problem: $(\mathbf{X}\mathbf{Q}_2^*)\hat{\mathbf{b}}^{(FLP)} \approx \mathbf{0}$. Once the TLS solution $\hat{\mathbf{b}}^{(FLP)}$ is computed, the forward prediction coefficients $\mathbf{b}^{(FLP)}$ are obtained, needed for the rooting of polynomial (6.1).

The TLS solution of $(\mathbf{X}\mathbf{Q}_2^*)\hat{\mathbf{b}}^{(FLP)} \approx \mathbf{0}$ can be computed from the SVD. Since $\mathbf{X}\mathbf{Q}_2^*$ has rank $K-p$ if noise-free data are used (refer to Eq. (2.7)):

$$\begin{aligned} \mathbf{X}\mathbf{Q}_2^* &= [\mathbf{S}_p \;\; \mathbf{S}_{K-p}]\begin{bmatrix} \mathbf{C}_p & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_{K-p} \end{bmatrix}[\mathbf{T}_p \;\; \mathbf{T}_{K-p}]^T\mathbf{Q}_2^* \\ &= \mathbf{S}_{K-p}\mathbf{C}_{K-p}\mathbf{T}_{K-p}^T\mathbf{Q}_2^* \end{aligned} \tag{6.6}$$

the null space of $(\mathbf{X}\mathbf{Q}_2^*)$ has rank $(M-p)-(K-p) = M-K$. Typically, the rank of the null space is $M-K > 1$ due to the fact that the LP method is preferably used at M much larger than K, e.g., around $M \approx \frac{3}{4}N$ [21]. Hence for noisy data, taking the SVD of $\tilde{\mathbf{X}} = (\mathbf{X}\mathbf{Q}_2^*) = \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^H$, the solution $\hat{\mathbf{b}}^{(FLP)}$

can be computed as the minimum norm TLS solution

$$\hat{\mathbf{b}}^{(FLP)} = \frac{-\overline{\tilde{\mathbf{V}}}_{K-p}\overline{\tilde{\mathbf{V}}}_{K-p}^H\mathbf{e}_{M-p}}{\mathbf{e}_{M-p}^T\overline{\tilde{\mathbf{V}}}_{K-p}\overline{\tilde{\mathbf{V}}}_{K-p}^H\mathbf{e}_{M-p}} \tag{6.7}$$

where $\overline{\tilde{\mathbf{V}}}_{K-p}$ is the basis of the orthogonal complement of the space spanned by the first $K-p$ right singular vectors $\tilde{\mathbf{V}}_{K-p}$, i.e., $\tilde{\mathbf{V}} = [\tilde{\mathbf{V}}_{K-p} \;\; \overline{\tilde{\mathbf{V}}}_{K-p}]$, and $\mathbf{e}_{M-p} = [0,0,\ldots,0,1]^T \in \mathbb{R}^{(M-p)\times 1}$. If $K-p < M-K$, it is more efficient to work with $\tilde{\mathbf{V}}_{K-p}$ via the replacement $\overline{\tilde{\mathbf{V}}}_{K-p}\overline{\tilde{\mathbf{V}}}_{K-p}^H = \mathbf{I} - \tilde{\mathbf{V}}_{K-p}\tilde{\mathbf{V}}_{K-p}^H$.

Algorithm 9

Outline of the LCTLS-FLP algorithm

Given. *A model-order estimate K, p known signal poles arranged in an $M\times p$ Vandermonde matrix $\mathbf{T}_p$ as in (6.5), and N uniformly sampled data points x_n, $n = 0,\cdots,N-1$, arranged in an $L\times M$ Hankel matrix $\mathbf{X}$ as in (2.1).*

Step 1. *Perform a QR decomposition of $\mathbf{T}_p = [\mathbf{Q}_1 \;\; \mathbf{Q}_2][\mathbf{R}^T \;\; \mathbf{0}]^T$, where $[\mathbf{Q}_1 \;\; \mathbf{Q}_2]$ is unitary, $\mathbf{Q}_1 \in \mathbb{C}^{M\times p}$, $\mathbf{Q}_2 \in \mathbb{C}^{M\times(M-p)}$ and $\mathbf{R} \in \mathbb{C}^{p\times p}$ is upper triangular.*

Step 2. *Compute the minimum norm TLS solution $\hat{\mathbf{b}}^{(FLP)}$ of*

$$(\mathbf{X}\mathbf{Q}_2^*)\hat{\mathbf{b}}^{(FLP)} \approx \mathbf{0}$$

via the SVD of $\mathbf{X}\mathbf{Q}_2^*$.

Step 3. *Compute the vector of forward prediction coefficients $\mathbf{b}^{(FLP)}$ from $[\mathbf{b}^{(FLP)} \;\; -1]^T = \gamma\mathbf{Q}_2^*\hat{\mathbf{b}}^{(FLP)}$ by calculating the appropriate scaling factor γ.*

Step 4. *Root the polynomial (6.1) using the coefficients $\mathbf{b}^{(FLP)}$ in order to find the $M-1$ poles, and extract the signal poles $z_k, k = 1,2,\ldots,K$ with the largest peak energy. The nonlinear parameters f_k, d_k, $k = 1,\cdots,K$ are found straight-forwardly using the relation $z_k = e^{(-d_k+j2\pi f_k)/f_s}$.*

Step 5. *Apply the linear-parameter-estimation part, Step 3, of the HTLS Algorithm 1 to the original data x_n, $n = 0,\cdots,N-1$. using the extracted K signal poles, to find the corresponding a_k and φ_k, $k = 1,\cdots,K$.*

LCTLS-BLP

The LCTLS-BLP algorithm is developed similarly to the LCTLS-FLP algorithm. The exactly known signal poles $z_k, k = 1, \ldots, p$ are the zeros of the backward-prediction-error filter polynomial in (6.4), which means:

$$B^{(BLP)}(z)|_{z=(z_k^*)^{-1}} = 0, \quad k = 1, \ldots, p$$

Stacking these p equations together results in a set of p linear constraints which expresses the assumed prior knowledge in terms of known signal poles and backward prediction coefficients:

$$\begin{bmatrix} z_1^* & z_1^{*2} & \cdots & z_1^{*(M-1)} \\ z_2^* & z_2^{*2} & \cdots & z_2^{*(M-1)} \\ \vdots & & & \vdots \\ z_p^* & z_p^{*2} & \cdots & z_p^{*(M-1)} \end{bmatrix} \begin{bmatrix} b_1^{(BLP)} \\ b_2^{(BLP)} \\ \vdots \\ b_{M-1}^{(BLP)} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

or

$$\hat{\mathbf{T}}_p^H \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} = \mathbf{0} \tag{6.8}$$

where $\hat{\mathbf{T}}_p$ permutes the first row of $\mathbf{T}_p$ in (6.5) to the bottom row.

This prior knowledge is incorporated into the TLS-BLP method by using the above equations as constraints. This is done by replacing the TLS solution $\mathbf{b}^{(BLP)}$ of $\mathbf{X}^{(BLP)}\mathbf{b}^{(BLP)} \approx \mathbf{x}^{(BLP)}$ in the TLS-BLP algorithm by the LCTLS solution of:

$$\begin{bmatrix} \mathbf{X}^{(BLP)} & \mathbf{x}^{(BLP)} \end{bmatrix} \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} \approx \mathbf{0} \tag{6.9}$$

$$\text{subject to } \hat{\mathbf{T}}_p^H \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} = \mathbf{0} \tag{6.10}$$

resulting in the LCTLS-BLP algorithm.

Algorithm 10

Outline of the LCTLS-BLP algorithm

Given. A model-order estimate K, p known signal poles arranged in an $M \times p$ matrix $\hat{\mathbf{T}}_p$ as in (6.8), and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel matrix $[\mathbf{x}^{(BLP)}\ \mathbf{X}^{(BLP)}]$ as in (6.3) and permuted to $[\mathbf{X}^{(BLP)}\ \mathbf{x}^{(BLP)}]$.

Step 1. Perform a QR decomposition of $\hat{\mathbf{T}}_p = [\hat{\mathbf{Q}}_1\ \hat{\mathbf{Q}}_2][\hat{\mathbf{R}}^T\ \mathbf{0}]^T$, where $[\hat{\mathbf{Q}}_1\ \hat{\mathbf{Q}}_2]$ is unitary, $\hat{\mathbf{Q}}_1 \in \mathbb{C}^{M \times p}$, $\hat{\mathbf{Q}}_2 \in \mathbb{C}^{M \times (M-p)}$ and $\hat{\mathbf{R}} \in \mathbb{C}^{p \times p}$ is upper triangular.

Step 2. Compute the minimum norm TLS solution $\hat{\mathbf{b}}^{(BLP)}$ of

$$([\mathbf{X}^{(BLP)}\ \mathbf{x}^{(BLP)}]\hat{\mathbf{Q}}_2)\hat{\mathbf{b}}^{(BLP)} \approx \mathbf{0}$$

via the SVD of $([\mathbf{X}^{(BLP)}\ \mathbf{x}^{(BLP)}]\hat{\mathbf{Q}}_2)$.

Step 3. Compute the vector of backward prediction coefficients $\mathbf{b}^{(BLP)}$ from $[\mathbf{b}^{(BLP)}\ -1]^T = \gamma\hat{\mathbf{Q}}_2\hat{\mathbf{b}}^{(BLP)}$ by calculating the appropriate scaling factor γ.

Step 4. Root the polynomial (6.4) using the coefficients $\mathbf{b}^{(BLP)}$ in order to find the $M-1$ poles, and extract the signal poles z_k with positive damping factors d_k, $k = 1, \cdots, K$. The nonlinear parameters f_k, d_k, $k = 1, \cdots, K$ are found straight-forwardly using the relation $z_k = e^{(-d_k + j2\pi f_k)/f_s}$.

Step 5. Apply the linear-parameter-estimation part, Step 3, of the HTLS Algorithm 1 to the original data x_n, $n = 0, \cdots, N-1$ using the extracted K signal poles, to find the corresponding a_k and φ_k, $k = 1, \cdots, K$.

HTLS(PK)

For noise-free data, we can remove the known part in the Vandermonde decomposition (2.7) of the original Hankel data matrix $\mathbf{X}$ by projecting the rows of $\mathbf{X}$ onto the orthogonal complement of the column space of $\mathbf{T}_p$, i.e.,

$$\begin{aligned} \tilde{\mathbf{X}} &= \mathbf{X}\mathbf{Q}_2^* \\ &= (\mathbf{S}_p\mathbf{C}_p\mathbf{T}_p^T + \mathbf{S}_{K-p}\mathbf{C}_{K-p}\mathbf{T}_{K-p}^T)\mathbf{Q}_2^* \\ &= \mathbf{S}_{K-p}\mathbf{C}_{K-p}\mathbf{T}_{K-p}^T\mathbf{Q}_2^* \end{aligned}$$

as shown in (6.6). The multiplication of $\mathbf{X}$ with $\mathbf{Q}_2^*$ from the right destroys the shift-invariant property in the row space but keeps the shift-invariant property in the column space, that is,

$$\mathbf{S}_{K-p}^{\uparrow} = \mathbf{S}_{K-p\downarrow}\mathbf{Z}_{K-p}$$

is still satisfied, where $\mathbf{Z}_{K-p}$ is a diagonal matrix composed of the $K-p$ unknown signal poles. On the other hand, we have $\tilde{\mathbf{X}} = \tilde{\mathbf{U}}_{K-p}\tilde{\mathbf{\Sigma}}_{K-p}\tilde{\mathbf{V}}_{K-p}^H$,

where $\tilde{\mathbf{U}}_{K-p}$ contains the first $K-p$ left singular vectors of $\tilde{\mathbf{X}}$. Hence,

$$\tilde{\mathbf{U}}^{\uparrow}_{K-p} = \tilde{\mathbf{U}}_{(K-p)\downarrow}\mathbf{Z}^{(\tilde{U})}_{K-p}$$

where $\mathbf{Z}^{(\tilde{U})}_{K-p}$ is equal to $\mathbf{Z}_{K-p}$ up to a similarity transform. Therefore, the $K-p$ unknown poles are the eigenvalues of $\mathbf{Z}^{(\tilde{U})}_{K-p}$.

Algorithm 11

Outline of the HTLS(PK) algorithm

Given. A model-order estimate K, p known signal poles arranged in an $M \times p$ Vandermonde matrix $\mathbf{T}_p$ as in (6.5), and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel matrix $\mathbf{X}$ as in (2.1).

Step 1: (QRD). Perform a QR decomposition of $\mathbf{T}_p = [\mathbf{Q}_1 \;\; \mathbf{Q}_2][\mathbf{R}^T \;\; \mathbf{0}]^T$, where $[\mathbf{Q}_1 \;\; \mathbf{Q}_2]$ is unitary, $\mathbf{Q}_1 \in \mathbb{C}^{M\times p}$, $\mathbf{Q}_2 \in \mathbb{C}^{M\times(M-p)}$ and $\mathbf{R} \in \mathbb{C}^{p\times p}$ is upper triangular.

Step 2: (SVD). Compute a truncated SVD of $\mathbf{X}\mathbf{Q}_2^* = \tilde{\mathbf{U}}_{K-p}\tilde{\mathbf{\Sigma}}_{K-p}\tilde{\mathbf{V}}^H_{K-p}$.

Step 3: (TLS). Compute the TLS solution $\mathbf{Z}^{(\tilde{U})}_{K-p}$ of $\tilde{\mathbf{U}}^{\uparrow}_{K-p} \approx \tilde{\mathbf{U}}_{(K-p)\downarrow}\mathbf{Z}^{(\tilde{U})}_{K-p}$ and extract the signal poles $z_k, k = p+1, p+2, \ldots, K$ from the eigenvalues of $\mathbf{Z}^{(\tilde{U})}_{K-p}$ to find f_k and d_k, $k = p+1, p+2, \cdots, K$.

Step 4: (LS). Apply the linear-parameter-estimation part, Step 3, of the HTLS Algorithm 1 to the original data x_n, $n = 0, \cdots, N-1$, using the known and estimated signal poles $z_k, k = 1, 2, \ldots, p, p+1, \ldots, K$ to find a_k and φ_k, $k = 1, 2, \cdots, K$.

An alternative way is to find an orthogonal basis $\mathbf{Q}_2$ from the QR decomposition of $\mathbf{S}_p$ in (6.6), i.e., $\mathbf{S}_p = [\mathbf{Q}_1 \;\; \mathbf{Q}_2]\begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix}$. Then the known poles are removed via left multiplication of $\mathbf{Q}_2^H$ with $\mathbf{X}$, i.e., $\tilde{\mathbf{X}} = \mathbf{Q}_2^H\mathbf{X}$. In this case, the shift-invariant property in the row space

$$\tilde{\mathbf{V}}^{*\uparrow}_{K-p} = \tilde{\mathbf{V}}^{*}_{(K-p)\downarrow}\mathbf{Z}^{(\tilde{V})}_{K-p}$$

can be used, and the unknown signal poles are the eigenvalues of $\mathbf{Z}^{(\tilde{V})}_{K-p}$.

HTLN(PK)

Prior knowledge is incorporated by adding constraints to the STLN problem $\mathbf{AY} \approx \mathbf{B}$, i.e.,

$$\begin{aligned} [\mathbf{A} \;\; \mathbf{B}]\begin{bmatrix}\mathbf{Y}\\ -\mathbf{I}\end{bmatrix} &\approx \mathbf{0} \\ \text{subject to} \quad \mathbf{T}_p^T\begin{bmatrix}\mathbf{y}_1\\ -1\end{bmatrix} &= \mathbf{0} \end{aligned} \tag{6.11}$$

where $\mathbf{y}_1$ is the forward prediction coefficient vector, or, the first column of $\mathbf{Y}$.

When $M = K+1$, i.e., $\mathbf{B}$ has only one column, $\mathbf{b}$, the constrained problem is simply

$$\begin{bmatrix}\mathbf{A} \;\; \mathbf{b}\\ \mathbf{T}_p^T\end{bmatrix}\begin{bmatrix}\mathbf{y}\\ -1\end{bmatrix} \approx \mathbf{0}$$

where $[\mathbf{A} \;\; \mathbf{b}]$ is affected by errors and has a Hankel structure, while $\mathbf{T}_p$ is error-free and fixed. This can be solved using the STLN algorithm so that the correction matrix applied to $[\mathbf{A} \;\; \mathbf{b}]$ is Hankel and the correction matrix applied to $\mathbf{T}_p^T$ is $\mathbf{0}$.

Recall that in Chapter 5 the two STLN algorithms for solving Hankel structured problems are described. However, the data matrix here is not purely Hankel structured. An easy solution is to add the constraints $\mathbf{T}_p^T\begin{bmatrix}\mathbf{y}\\ -1\end{bmatrix} = \mathbf{0}$ to the constraints $\mathbf{r}(\mathbf{z}+\Delta\mathbf{z}) = \mathbf{0}$ or $\hat{\mathbf{B}}\Delta\mathbf{z} - \hat{\mathbf{d}} = \mathbf{0}$ in (5.10) so that the STLN Algorithm-2 can be used after minor modifications of Step 2(a) as shown below:

Modified STLN Algorithm-2

Apply STLN Algorithm-2 in which Step 2(a) is modified as follows:

Step 2 (a).

$$\hat{\mathbf{A}} := \begin{bmatrix}\mathbf{D}_\alpha & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0}\end{bmatrix}, \hat{\mathbf{B}} := \begin{bmatrix}\mathbf{P}_\alpha - \check{\mathbf{Y}} & \mathbf{P}_\beta & -(\mathbf{A}+\Delta\mathbf{A})\\ \mathbf{0} & \mathbf{0} & \mathbf{P}_p\end{bmatrix},$$

$$\Delta\mathbf{z} := \begin{bmatrix}\Delta\alpha\\ \Delta\beta_0\\ \Delta\mathbf{y}\end{bmatrix}, \hat{\mathbf{b}} := -\begin{bmatrix}\mathbf{D}_\alpha\alpha\\ \mathbf{D}_\beta\beta_0\end{bmatrix} \hat{\mathbf{d}} = \begin{bmatrix}-\mathbf{r}\\ \mathbf{p}_p\end{bmatrix}.$$

$\mathbf{P}_p$ and $\mathbf{p}_p$ are, respectively, the first $M-1$ columns and the last column of

$\mathbf{T}_p^T$, i.e., $\mathbf{T}_p^T = [\mathbf{P}_p, \mathbf{p}_p]$.

When $M > K+1$, the constrained problem can be written in the following form via the vectorization as done in (5.15):

$$\begin{bmatrix} \mathbf{A} & 0 & \cdots & 0 & \mathbf{b}_1 \\ 0 & \mathbf{A} & \cdots & 0 & \mathbf{b}_2 \\ \vdots & & \ddots & & \vdots \\ 0 & \cdots & 0 & \mathbf{A} & \mathbf{b}_{(M-K)} \\ \mathbf{P}_p & & \mathbf{0} & & \mathbf{p}_p \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_{(M-K)} \\ -1 \end{bmatrix} \approx \mathbf{0}$$

In this way, the problem can also be solved by the STLN algorithm which preserves the special structure of the data matrix.

Algorithm 12

Outline of the HTLN(PK) algorithm

Given. A model-order estimate K, p known signal poles arranged in an $M \times p$ Vandermonde matrix $\mathbf{T}_p$ as in (6.5), and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel matrix $\mathbf{X} = [\mathbf{A}_{L\times K} \;\; \mathbf{B}_{L\times(M-K)}]$.

Step 1. Apply the modified STLN Algorithm-2 to (6.11) in order to find the Hankel correction matrix $[\mathbf{E} \;\; \mathbf{F}]$ and STLN solution $\mathbf{Y}$ such that

$$[\mathbf{A}+\mathbf{E} \;\; \mathbf{B}+\mathbf{F}] \begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix} = \mathbf{0} \text{ and } \mathbf{T}_p^T \begin{bmatrix} \mathbf{y}_1 \\ -1 \end{bmatrix} = \mathbf{0}$$

Step 2. Compute a QR decomposition of

$$\begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix} = [\mathbf{Q}_n \;\; \mathbf{Q}_s] \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}$$

where $[\mathbf{Q}_n \;\; \mathbf{Q}_s]$ is unitary, $\mathbf{R}$ is upper triangular.

Step 3. Compute the TLS solution $\mathbf{Z}^{(Q)}$ of the following set of equations:

$$\mathbf{Q}_s^{*\uparrow} \approx \mathbf{Q}_{s\downarrow}^{*} \mathbf{Z}^{(Q)}$$

and retrieve the eigenvalues z_k of $\mathbf{Z}^{(Q)}$. The nonlinear parameters f_k, d_k, $k = 1, \cdots, K$ are found straight-forwardly using the relation $z_k = e^{(-d_k + j2\pi f_k)/f_s}$.

Step 4. Apply the linear-parameter-estimation part, Step 3, of the HTLS Algorithm 1 to the original data x_n, $n = 0, \cdots, N-1$ using the extracted signal poles, to find the corresponding a_k and φ_k, $k = 1, 2, \cdots, K$.

All the prior-knowledge-incorporating estimators, namely, LCTLS-FLP, LCTLS-BLP, HTLS(PK) and HTLN(PK) presented above, are computationally more efficient than their corresponding estimator TLS-FLP, TLS-BLP, HTLS and HTLN, respectively. This difference in computational efficiency is more pronounced as the number p of *a priori* known poles increases. This is because the former estimators only need to compute the SVD of smaller sized matrices, as compared to the latter estimators, and the SVD is computationally the most time-consuming part of each method. For example, the LCTLS-FLP method is more efficient than the TLS-FLP method since the SVD in LCTLS-FLP is performed on the smaller matrix $\mathbf{X}\mathbf{Q}_2^*$ which has p columns fewer than the matrix $\mathbf{X}$ as used in the TLS-FLP method. The savings for the LCTLS-BLP and HTLS methods are similar. As for HTLN(PK), the constraints expressing prior knowledge reduce the number of iterations required for convergence.

6.3 Simulation results

We study and compare the performance of different estimation algorithms, HTLS, LCTLS-FLP, LCTLS-BLP, HTLS(PK) and HTLN(PK), in terms of respectively resolution and parameter accuracy. In order to obtain good parameter estimates, the number of columns M of the data matrix used in the LP, HTLS and HTLN estimators is respectively $M \approx \frac{2}{3}N$, $M \approx \frac{1}{2}N$ and $M = K+1$, unless specified otherwise. The extraction of the signal peaks in the LP methods is done by selecting the K peaks with the largest peak energy, which is defined as $\sum_{n=0}^{N-1} a_k^2 |z_k|^{2n} = a_k^2(1-|z_k|^{2N})/(1-|z_k|^2)$ for the k-th peak [80]. The maximum number of iterations in the iterative method HTLN is set to 100. The tolerance ϵ in the stop criterion of the STLN iterations (5.14) for HTLN is $\epsilon = 10^{-10}$. If not specified, all the HTLN runs are convergent within 100 iterations.

Example 1: *Signal-1*

In each algorithm, $N = 25$ data samples are taken, and the parameters $f_1 = 0.2$ Hz and $d_1 = 0.01$ rad/s of the first peak of *Signal-1* are assumed to be known *a priori*.

method	Bias±standard deviation	
	f_2 (Hz)	d_2 (rad/s)
HTLS	-0.0001 ± 0.0058	-0.0049 ± 0.0689
LCTLS-FLP	-0.0001 ± 0.0025	0.0060 ± 0.0147
LCTLS-BLP	0.0002 ± 0.0026	0.0040 ± 0.0130
HTLS(PK)	0.0000 ± 0.0020	0.0013 ± 0.0124
HTLN(PK)	0.0000 ± 0.0018	0.0011 ± 0.0120
method	Bias±standard deviation	
	a_2 (a.u.)	ψ_2 (°)
HTLS	0.1704 ± 0.7293	-2.2126 ± 46.7598
LCTLS-FLP	0.0944 ± 0.2254	-2.2586 ± 11.6442
LCTLS-BLP	0.0464 ± 0.1730	-2.2695 ± 9.9867
HTLS(PK)	0.0195 ± 0.1532	-0.7168 ± 8.1088
HTLN(PK)	0.0147 ± 0.1491	-0.8432 ± 7.6712

Table 6.1: Example 1: Bias ± standard deviation estimates of the parameters of the unknown peak at SNR=15dB, $N = 25$. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $M = 13, 17, 3$ is used respectively for the HTLS method, HTLS(PK) method, LP-based methods and the HTLN(PK) method.

Fig. 6.1(a) shows the failure rate of each method. Since the two peaks of this simulated signal are so close that their spectra overlap, all the methods incorporating prior knowledge of f_1, d_1 of the first peak have much fewer failures than the HTLS method. The LCTLS-FLP and LCTLS-BLP methods are comparable in resolution in this example and reduce the failure rate of HTLS with a factor 4. The HTLS(PK) and HTLN(PK) methods in turn reduce the failure rate of LCTLS-FLP by a factor of at least 10, since both methods have only one failure at SNR= 10dB ($\sigma_\nu \approx 0.2236$) and no failures at higher SNR's.

In each of the accuracy plots, Fig. 6.1(b) and (c), two full lines are added: the lower and upper one representing respectively the CR lower bounds with and without prior knowledge of f_1 and d_1 [81]. It can be clearly seen that the HTLS method produces the largest RMSE values which closely follow the

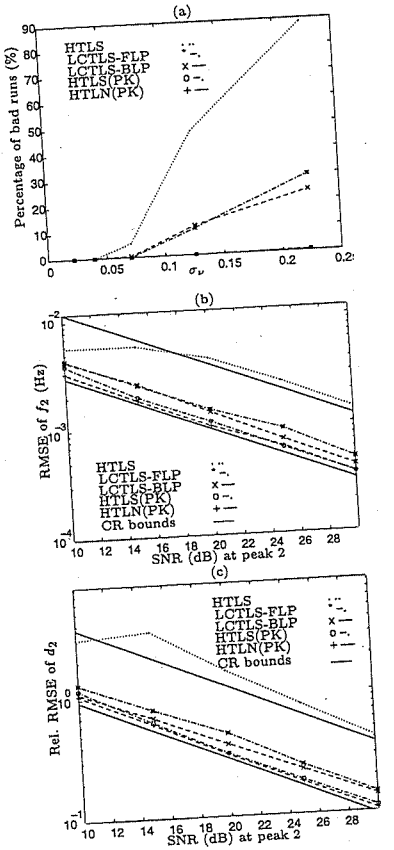


Figure 6.1: Example 1: (a) Failure rate versus σ_ν. (b) RMSE of $f_2 = 0.22$ Hz and (c) Relative RMSE of $d_2 = 0.02$ rad/s versus the SNR at this peak. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$. $M = 13, 17, 3$ is used respectively for the HTLS method, HTLS(PK) method, LP-based methods and the HTLN(PK) method. HTLN(PK) has 2 non-convergent cases within 100 iterations of the STLN Algorithm-2 at SNR=10dB.

upper full line. The prior-knowledge-incorporating method, LCTLS-FLP, reduces the RMSE values of the HTLS frequency and damping factor estimates with a factor 3 to 4. The accuracy of these LCTLS-FLP estimates is further improved with a factor 1.2, 1.5 and 1.6, respectively, by using the LCTLS-BLP, HTLS(PK) and HTLN(PK) methods. Among these, only the HTLS(PK) method and the HTLN(PK) method approach the lower full line. A more careful examination indicates that the HTLN(PK) method achieves the lowest RMSE values, and especially, the lowest standard deviations as listed in Table 6.1.

methods	SNR=25dB	SNR=20dB	SNR=15dB
HTLS	1.19e+05	1.21e+05	1.23e+05
LCTLS-FLP	9.06e+05	9.06e+05	9.00e+05
LCTLS-BLP	4.29e+05	4.29e+05	4.30e+05
HTLS(PK)	1.19e+05	1.19e+05	1.20e+05
HTLN(PK)	3.45e+07	4.07e+07	5.08e+07

Table 6.2: Example 1: Number of flops required by different methods using Matlab averaged over 200 runs. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$. $M = 13, 17, 3$ is used respectively for the HTLS method, the HTLS(PK) method, LP-based methods and the HTLN(PK) method.

Table 6.2 compares the computational efficiency of the proposed algorithms in terms of the number of flops averaged over 200 runs required by each method when implemented using Matlab. Obviously, the iterative method, HTLN(PK), requires the largest number of flops compared to the other SVD-based methods while HTLS(PK) needs the smallest number of flops. Observe that the improvement in computational efficiency of HTLS(PK) compared to HTLS is not large because only prior knowledge of *one* peak is incorporated. The number of flops required by the HTLN(PK) method increases when the SNR is lowered, because the number of required iterations up to convergence increases. In all examples we studied, the improvement in accuracy of the HTLN(PK) estimates compared to that of HTLS(PK) estimates is not large enough to justify the large increase in computational load, especially when the number of data samples is large, and therefore the HTLN(PK) method is not considered in the following study.

Fig. 6.2 shows the failure rate of each method versus varying number of columns M of the data matrix. In LP methods, $M - 1$ is actually the prediction length. It is clear from this failure rate plot that the prior-knowledge-

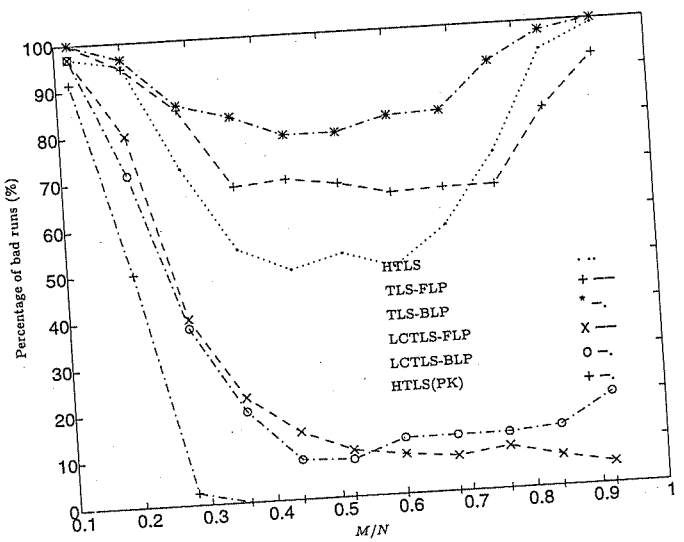


Figure 6.2: Example 1: Failure rate of each method versus M/N of the data matrix. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$, SNR=15dB or $\sigma_\nu = 0.1257$.

incorporating methods, LCTLS-FLP, LCTLS-BLP and HTLS(PK), give a wider minimum than the corresponding no-prior-knowledge-incorporating methods, TLS-FLP, TLS-BLP and HTLS, respectively. The same observation holds for the accuracy plots, which are not included. Hence the prior-knowledge-incorporating methods are not sensitive to the choice of M in this example. In particular, the region of constant failure rate and accuracy is as wide as $0.35 \leq M/N \leq 0.9$ for HTLS(PK) in contrast to $0.4 \leq M/N \leq 0.6$ for HTLS. In the following study, we will use $M \approx \frac{2}{3}N$ for the LCTLS-FLP and LCTLS-BLP methods, and $M \approx \frac{1}{2}N$ for the HTLS and HTLS(PK) methods, unless otherwise specified.

If, instead, the pole of the second peak is known *a priori*, the results are very similar and therefore omitted here.

Example 2: *Signal-2*

In each algorithm, $N = 128$ data samples are taken, and the parameters $f_4 = 353$ Hz and $d_4 = 117$ rad/s of the fourth peak of *Signal-2* are assumed to be known *a priori*.

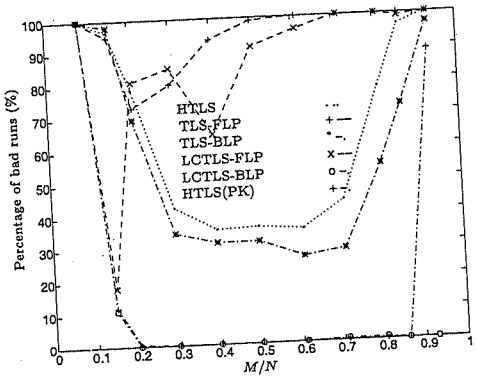


Figure 6.3: Example 2: Failure rate of each method versus M/N of the data matrix. $f_4 = 353$ Hz, $d_4 = 117$ rad/s are known *a priori*. $N = 128$, $\sigma_\nu = 2$.

If we vary the number of columns M used in each method as done for *Signal-1*, the results shown in Fig. 6.3 are similar to Fig. 6.2, except that the TLS-FLP and LCTLS-FLP methods have sharp minima around $M = 20$, instead of $M \approx \frac{2}{3}N \approx 85$. The reason is the following. When the length of the prediction filter is very large, say, around 85, and the noise is wildly fluctuating, some noise peaks are fitted as extraneous peaks with large peak energy. Thus the extraction of $K = 5$ signal poles from 85 fitted poles is rather difficult. Therefore, $M = 20$ will be used in the LCTLS-FLP method for this signal. However, $M \approx \frac{2}{3}N$ remains unchanged in the LCTLS-BLP method, since it computes all the extraneous peaks with negative damping factors, which are easily identified and eliminated.

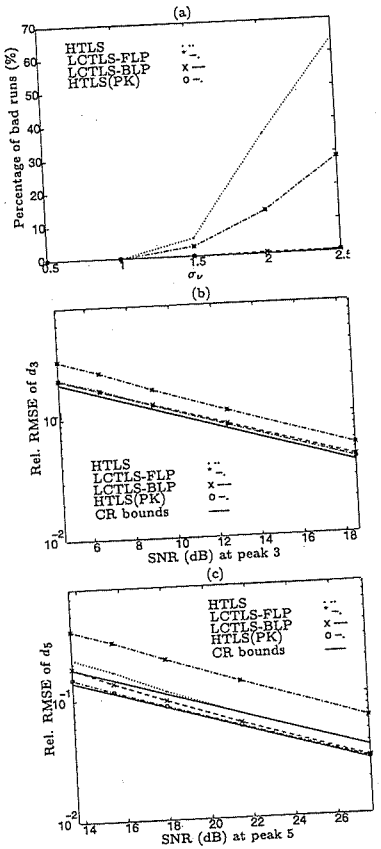


Figure 6.4: Example 2: (a) Failure rate versus σ_ν. Relative RMSE of the estimates of (b) $d_3 = 197$ rad/s and (c) $d_5 = 808$ rad/s versus the peak SNR. $f_4 = 353$ Hz, $d_4 = 117$ rad/s are known *a priori*. $N = 128$. $M = 64, 20, 85$ is used respectively for the HTLS, HTLS(PK), LCTLS-FLP and LCTLS-BLP methods.

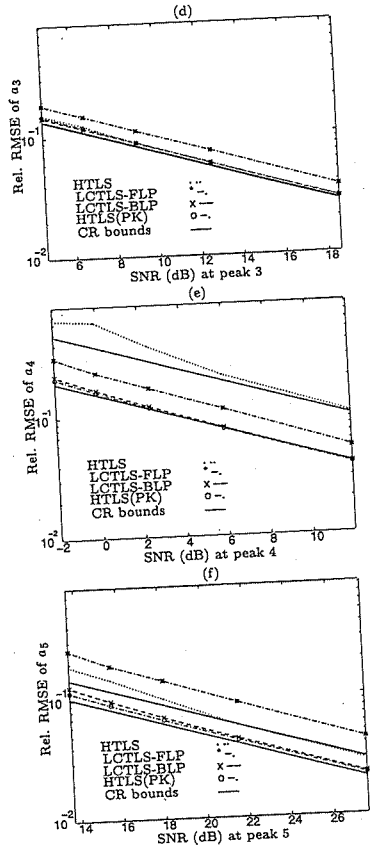


Figure 6.4: (cont.) Example 2: Relative RMSE of (d) $a_3 = 6.0$ a.u., (e) $a_4 = 2.8$ a.u., and (f) $a_5 = 17.0$ a.u. versus the peak SNR.

Fig. 6.4(a) gives the plot of the failure rate of the HTLS, LCTLS-FLP, LCTLS-BLP and HTLS(PK) methods versus the noise level σ_ν, which is similar to that of Example 1 (Fig. 6.1(a)), except that the LCTLS-BLP and HTLS(PK) methods are now comparable, with hardly any failure. However, the accuracy plots as shown in Fig. 6.4(b)$\sim$(f) are different from those of Example 1. Although $M = 20$ is used in LCTLS-FLP, the computed RMSE values are still too high due to un-eliminated extraneous peaks. The best accuracy is obtained by the HTLS(PK) and LCTLS-BLP methods, both approaching the CR lower bounds with prior knowledge. Actually, the accuracy of the HTLS(PK) estimates is slightly better than that of the LCTLS-BLP estimates.

Note that the gap between the CR lower bounds with and without prior knowledge is largest for a_4, obvious for a_5 and d_5, but hardly observable for a_3 and d_3. This is because peak 4 is the peak with known poles, peak 3 is far away from this known peak, but peak 5 is overlapping with it. The gap between the two CR bounds depends mainly on this difference in frequency. Hence, the incorporation of prior knowledge of known signal poles only improves resolution and parameter accuracy when the corresponding peaks are close to or even overlapping with the unknown peaks.

6.4 Conclusion

In parameter estimation of closely-spaced sinusoids, prior knowledge of some known signal poles can be incorporated, as done in the LCTLS-FLP algorithm proposed by Dowling et al. and based on forward prediction and total least squares. In this chapter, three better algorithms are presented, namely LCTLS-BLP, HTLS(PK) and HTLN(PK), derived respectively from the TLS-BLP, HTLS, and HTLN algorithms. These methods incorporate the same prior knowledge but clearly outperform the LCTLS-FLP method in resolution and parameter accuracy. Computer simulations indicate that the new methods, especially HTLS(PK) and HTLN(PK), improve the LCTLS-FLP method significantly, with a factor of at least 40 in resolution and 1.2 in accuracy. The accuracy of the HTLN(PK) method is slightly better than that of the HTLS(PK) method since it gives rise to the lowest RMSE values, and especially, the lowest standard deviations of the estimated parameters. However, HTLS(PK) is shown to be computationally the most efficient method. The improvement in accuracy of the HTLN(PK) estimates compared to that of HTLS(PK) estimates is not large enough to justify the large increase in computational load, especially when the number of data samples is large. Therefore, the HTLS(PK)

method is strongly recommended for exponential data fitting in the presence of known signal components. More analysis of the HTLS(PK) method and its application to NMR spectroscopy is given in the next chapter.

Chapter 7

More analysis of HTLS(PK) with application to NMR

The HTLS(PK) estimation method has been presented in the previous chapter and shown to be superior to other prior-knowledge-incorporating estimation techniques. In this chapter, we interpret the incorporation of prior knowledge in the HTLS(PK) method in terms of a prefiltering operation discussed in Chapter 3, which enables an intuitive analysis in the frequency domain. Subsequently, more results of quantification of simulated signals using HTLS(PK) are presented, including a performance comparison of HTLS(PK) and prefiltered HTLS, enhancement of the HTLS(PK) method, a stepwise efficiency comparison of HTLS(PK) and HTLS, and a sensitivity study of HTLS(PK) with respect to the accuracy of the given poles. Finally, the HTLS(PK) method is applied to a real-world NMR signal and compared with the HTLS estimator.

7.1 HTLS(PK) and prefiltering

Comparing $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{Q}_2^*$ performed in the HTLS(PK) method and $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}^T$ performed in the prefiltered HTLS method in Section 3.3, it is easy to see that $\mathbf{Q}_2^H$ is functioning as a filter matrix $\mathbf{H}$ discussed in Chapter 3. This enables us to examine the incorporation of prior knowledge in terms of a prefiltering operation.

A characteristic of this prefiltering operation is that it annihilates the p Vandermonde vectors, $\mathbf{t}_i, i = 1, 2, \ldots, p$, which consist of known signal poles $z_i, i = 1, 2 \ldots, p$ as defined in Definition 1. By annihilation, we mean that the filtered Vandermonde vectors $\mathbf{t}_i$ of p exponentials are all zeros, that is,

$$\mathbf{t}_i^T \mathbf{Q}_2^* = 0, \qquad i = 1, 2, \ldots, p$$

or

$$\mathbf{T}_p^T \mathbf{Q}_2^* = \mathbf{0}$$

As pointed out in [5], this implies a filtering operation via an FIR filter with p zeros at known signal poles. Let us examine such an FIR filtering in terms of a prefiltering operation discussed in Chapter 3. Suppose the z-transfer function of this FIR filter is

$$\begin{aligned} \mathsf{H}(\mathsf{z}) &= \mathsf{z}^{-p}(\mathsf{z} - z_1)(\mathsf{z} - z_2)\cdots(\mathsf{z} - z_p) \\ &= h_1 + h_2\mathsf{z}^{-1} + h_3\mathsf{z}^{-2} + \ldots + h_{p+1}\mathsf{z}^{-p} \end{aligned}$$

Let $q = p + 1$. We can perform a discrete convolution of the input data $x_1, x_2, \ldots, x_M, 0, \ldots, 0$ and the filter impulse response $h_1, h_2, \ldots, h_q, 0, \ldots, 0$ with zero-padding. As discussed in Section 3.2 of Chapter 3, several structures of the filter matrix are available. Here we take the form of filter matrix in the inner most box of (3.2) without distortion, i.e.

$$\mathbf{H}_{\mathrm{PK}} = \begin{bmatrix} h_q & h_{q-1} & \cdots & \cdots & h_1 & 0 & 0 \\ 0 & \ddots & \ddots & & & \ddots & 0 \\ 0 & 0 & h_q & h_{q-1} & \cdots & \cdots & h_1 \end{bmatrix} \in \mathbb{C}^{(M-q+1)\times M}$$

where $q = p + 1$ (7.1)

Such a filter matrix generates filtered output without distortion in the absence of noise and has the same dimension as $\mathbf{Q}_2^H$.

In this way, the filtered Vandermonde vectors, $\mathbf{t}_i, i = 1, 2, \ldots, p$, of the p exponentials via the filter matrix $\mathbf{H}_{\mathrm{PK}}$ are also zeros:

$$\mathbf{t}_i^T \mathbf{H}_{\mathrm{PK}}^T = [z_i^p \mathsf{H}(z_i),\ z_i^{p+1}\mathsf{H}(z_i),\ \cdots,\ z_i^{p+M-1}\mathsf{H}(z_i)] = \mathbf{0}$$
$$i = 1, 2, \ldots, p$$

or

$$\mathbf{T}_p^T \mathbf{H}_{\mathrm{PK}}^T = \mathbf{0}$$

This implies that the filter matrix $\mathbf{H}_{\mathrm{PK}}$ is playing a similar role to the filter matrix $\mathbf{Q}_2^H$ in that

- $\mathbf{H}_{\mathrm{PK}}$ and $\mathbf{Q}_2^H$ have the same dimension, $(M - p) \times M$, and are both full rank;
- $\mathbf{H}_{\mathrm{PK}}$ and $\mathbf{Q}_2^H$ are both orthogonal to the p-dimensional column space of $\mathbf{T}_p$;
- $\mathbf{H}_{\mathrm{PK}}$ *and* $\mathbf{Q}_2^H$ *have the same range, i.e.,* $\mathcal{R}(\mathbf{H}_{\mathrm{PK}}) = \mathcal{R}(\mathbf{Q}_2^H)$.

Therefore, the incorporation of prior knowledge through the filter matrix $\mathbf{Q}_2^H$ is actually an orthogonalized version of prefiltering through the FIR filter matrix $\mathbf{H}_{\mathrm{PK}}$. This relationship facilitates our interpretation of prior-knowledge-incorporation in the frequency domain.

Fig. 7.1 shows the modulus of frequency responses of the FIR filters designed to cancel the signal pole, z_1, of the two-peak signal, *Signal-1*, which has a rational z-transform

$$\mathsf{X}(\mathsf{z}) = \frac{c_1}{1 - z_1\mathsf{z}^{-1}} + \frac{c_2}{1 - z_2\mathsf{z}^{-2}}$$

Let $z_{1\mathrm{PK}} = e^{(-d_{1\mathrm{PK}} + j2\pi f_{1\mathrm{PK}})\Delta t}$ be the given prior knowledge. Then, the filter z-transfer function is

$$\mathsf{H}(\mathsf{z}) = 1 - z_{1\mathrm{PK}}\mathsf{z}^{-1}$$

When the given prior knowledge is exact, i.e., $z_{1\mathrm{PK}} = z_1$ as is the case of full curves in Fig. 7.1, the first pole of *Signal-1* is canceled or suppressed completely, and the problem of estimating poles of 2 exponentials is simplified to that of only 1 exponential after such prefiltering. Naturally, the decrease of the number of unknowns results in better resolution and accuracy, especially when the known and unknown exponentials are closely spaced, overlapping, and thus difficult to resolve.

When the given prior knowledge, $f_{1\mathrm{PK}}$ or $d_{1\mathrm{PK}}$ of $z_{1\mathrm{PK}}$, is perturbed, the designed filters are also perturbed as shown by the dashed lines in Fig. 7.1. By multiplying the frequency response of an FIR filter with the original signal spectrum, we can get the spectrum of a filtered signal. In Fig. 7.1 (a), $d_{1\mathrm{PK}}$ is fixed at d_1. The filter frequency response shifts due to a perturbation of the known frequency, $f_{1\mathrm{PK}} = f_1 + \Delta f$. This affects both filtered peaks, namely, their positions (frequencies) and shapes (damping factors). Intuitively, a positive Δf has a stronger influence than a negative Δf for the same absolute $|\Delta f|$, since $f_{1\mathrm{PK}}$ is closer to f_2 in the former case and further away from f_2 in the latter case when $f_1 < f_2$. In Fig. 7.1 (b), $f_{1\mathrm{PK}}$ is fixed at f_1 but $d_{1\mathrm{PK}}$ is perturbed. The series of the filter frequency responses corresponding to a perturbation of the known damping factor, $d_{1\mathrm{PK}} = d_1 + \Delta d$, is always centered

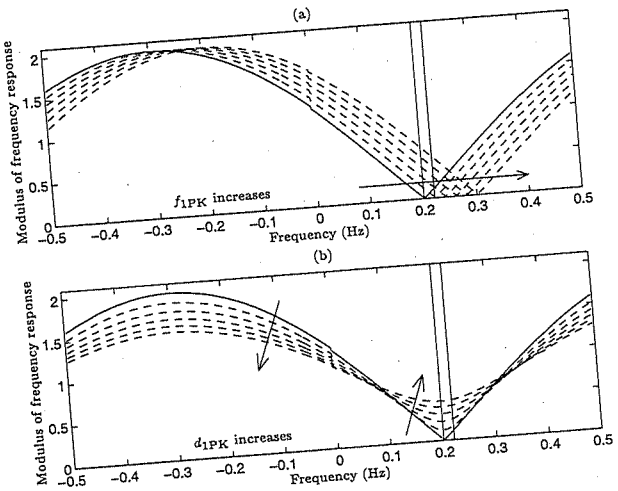


Figure 7.1: Frequency response of FIR filters H(z) for *Signal-1*, when the given prior knowledge is (a) $f_{1PK} = f_1 + \Delta f$, $d_{1PK} = d_1$, and (b) $f_{1PK} = f_1$, $d_{1PK} = d_1 + \Delta d$. The parallel vertical full lines denote the position of the two sinusoids in *Signal-1*. When Δf increases from 0 Hz till 0.1 Hz or Δd increases from 0 rad/s till $0.1 \times 2\pi$ rad/s, the filter frequency response shifts in the direction of arrows.

at $f_1 = 0.2$ Hz, the frequency of the first peak, which is supposed to be annihilated. The change in Δd determines the degree of suppression of the first peak. The larger Δd is, the less the first peak is suppressed. The un-suppressed part of the filtered first peak affects the parameter estimation of the second peak, especially when Δd is positive, instead of negative. This is the exponential effect in the damping factor. Take Fig. 7.1 (b) for example. As d_{1PK} changes from 0.01 rad/s to $0.01 + 0.1 \times 2\pi$ rad/s, the minimum modulus of the filter frequency response changes from 0.01 to 0.47. However, if d_{1PK} changes from 0 rad/s to 0.01 rad/s, this minimum only changes from 0 to 0.01, which is hardly visible. Of course, such frequency-domain analysis is rough. But it gives some

idea about the effect of perturbations in the known signal poles.

Though the prefiltering operation using $\mathbf{H}_{PK}$ of an FIR filter can be helpful in interpretation, it is less accurate in our parameter estimation than the HTLS(PK) method using the orthogonal filter matrix $\mathbf{Q}_2$. This is because, in HTLS(PK), $\mathbf{Q}_2^H$ has orthonormal rows due to $\mathbf{Q}_2^H\mathbf{Q}_2 = \mathbf{I}_{M-p}$. As a result, *a prefiltering operation* $\bar{\mathbf{X}} = \mathbf{X}\mathbf{Q}_2^*$ *does not change the original covariance matrix of white noise, which is, asymptotically, a scalar multiple of an identity matrix, in contrast to a prefiltering operation* $\bar{\mathbf{X}} = \mathbf{X}\mathbf{H}_{PK}^T$. This can be easily verified from

$$\mathbf{X} = \overset{\circ}{\mathbf{X}} + \mathbf{E},\ \overset{\circ}{\mathbf{X}}^H\mathbf{E} = 0,\ \mathbf{E}^H\mathbf{E} = \mu^2\mathbf{I}_M = L\nu^2\mathbf{I}_M$$

$$\bar{\mathbf{X}} = (\overset{\circ}{\mathbf{X}} + \mathbf{E})\mathbf{Q}_2^* = \overset{\circ}{\bar{\mathbf{X}}} + \bar{\mathbf{E}}$$

where $\overset{\circ}{\bar{\mathbf{X}}} = \overset{\circ}{\mathbf{X}}\mathbf{Q}_2^*$, $\bar{\mathbf{E}} = \mathbf{E}\mathbf{Q}_2^*$, and

$$\bar{\mathbf{X}}^H\bar{\mathbf{X}} = \mathbf{Q}_2^T\overset{\circ}{\mathbf{X}}^H\overset{\circ}{\mathbf{X}}\mathbf{Q}_2^* + \mu^2\mathbf{Q}_2^T\mathbf{Q}_2^* = \mathbf{Q}_2^T\overset{\circ}{\mathbf{X}}^H\overset{\circ}{\mathbf{X}}\mathbf{Q}_2^* + \mu^2\mathbf{I}_{M-p}$$

Thus the requirement for the HTLS estimator that the noise covariance matrix be a scalar multiple of identity matrix is satisfied after prefiltering with $\mathbf{Q}_2^H$. The benefit of the orthogonal filter matrix $\mathbf{Q}_2^H$ compared to the FIR filter matrix $\mathbf{H}_{PK}$ is also confirmed by numerical results in the next section.

It is to be noted that the prior-knowledge-incorporating methods, LCTLS-FLP and LCTLS-BLP, can also be interpreted as such a prefiltering operation in a similar way.

7.2 Quantification of simulated signals

In this section, we study the HTLS(PK) method more extensively, and compare it to the prefiltered HTLS and HTLS method. Again, we work with the second-order simulated signal, *Signal-1*, and the fifth-order simulated signal, *Signal-2*. For *Signal-1*, $N = 25$ data samples are used, and the parameters $f_1 = 0.2$ Hz and $d_1 = 0.01$ rad/s of the first peak are assumed to be known *a priori*. For *Signal-2*, $N = 128$ data samples are used, and the parameters $f_4 = 353$ Hz and $d_4 = 117$ rad/s of the fourth peak are assumed to be known *a priori*.

7.2.1 Comparison of HTLS(PK) and prefiltered HTLS

Fig. 7.2 compares HTLS(PK) with prefiltered HTLS using prior knowledge, z_1, of *Signal-1*. $\mathbf{H}_{PK}$-HTLS stands for the prefiltered HTLS method via $\bar{\mathbf{X}} =$



KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELECTROTECHNIEK
AFDELING ESAT
Kardinaal Mercierlaan 94 — 3001 Leuven (Heverlee)

SUBSPACE-BASED PARAMETER ESTIMATION OF EXPONENTIALLY DAMPED SINUSOIDS WITH APPLICATION TO NUCLEAR MAGNETIC RESONANCE SPECTROSCOPY DATA

Promotors:
Prof. Dr. Ir. S. Van Huffel
Prof. Dr. Ir. J. Vandewalle
Prof. Dr. P. Van Hecke

Thesis submitted to obtain the degree of doctor in applied sciences

by

Hua CHEN

May 1996



KATHOLIEKE UNIVERSITEIT LEUVEN
FACULTEIT TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELECTROTECHNIEK
AFDELING ESAT
Kardinaal Mercierlaan 94 — 3001 Leuven (Heverlee)

SUBSPACE-BASED PARAMETER ESTIMATION OF EXPONENTIALLY DAMPED SINUSOIDS WITH APPLICATION TO NUCLEAR MAGNETIC RESONANCE SPECTROSCOPY DATA

Jury:
Chairman: Prof. Dr. Ir. W. Dutré, vice-dean
Prof. Dr. Ir. S. Van Huffel, promotor
Prof. Dr. Ir. J. Vandewalle, promotor
Prof. Dr. P. Van Hecke, promotor
Dr. Ir. M. Moonen
Prof. Dr. Ir. D. van Ormondt
Prof. Dr. Ir. P. Van Dooren
Prof. Dr. Ir. P. Suetens

Thesis submitted to obtain the degree of doctor in applied sciences

by

Hua CHEN

UDC: 621.395.38(043) May 1996

© Katholieke Universiteit Leuven - Faculteit Toegepaste Wetenschappen
Arenbergkasteel, B-3001 Heverlee (Belgium)

Alle rechten voorbehouden. Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt worden door middel van druk, fotocopie, microfilm, elektronisch of op welke andere wijze ook zonder voorafgaandelijke schriftelijke toestemming van de uitgever.

All rights reserved. No part of this publication may be reproduced in any form by print, photoprint, microfilm or any other means without written permission from the publisher.

D/1996/7515/16

ISBN 90-5682-030-3

Acknowledgments

I would like to express my gratitude to my daily supervisor, Prof. S. Van Huffel, of the ESAT-SISTA group, for her invaluable guidance, insightful suggestions and enlightening discussions. I am heavily indebted to her for her painstaking reading and revising of all my papers, as well as the drafts of this thesis.

I would also like to thank Prof. J. Vandewalle, head of ESAT-SISTA, who provides me with the opportunity to study and work in the nice environment of ESAT-SISTA, K. U. Leuven. His constant encouragement and enthusiasm is very important to the accomplishment of this work.

My thanks also go to Prof. P. Van Hecke from the NMR lab of the Gasthuisberg hospital, for providing NMR data for analysis, suggesting the problem, and careful reading of my papers and thesis.

A special note of thanks goes to Prof. D. Van Ormondt from the Applied Physics laboratory of Delft University of Technology for his novel ideas, cheerful discussions, and valuable advice on fitting NMR data offered.

As well, Dr. M. Moonen is gratefully acknowledged for his critical review and constructive comments which made this thesis more readable.

I am truly grateful to all the staff and my fellow students, Belgian and Chinese, for their help and assistance in various way.

Last but not least, I would like to thank my husband, Jianbin, and our children, for their love, support, encouragement and tolerance over the last few years.

Hua Chen

Abstract

This dissertation treats the issue of subspace-based parameter estimation for data modeled as a sum of exponentially damped complex-valued exponentials in the context of NMR spectroscopy where the models are of the form

$$x_n = \sum_{k=1}^{K} a_k e^{j\varphi_k} e^{(-d_k + j2\pi f_k)n\Delta t} + e_n \qquad n = 0, \ldots, N-1$$

Firstly, aiming at emphasizing the sinusoids of interest and suppressing the unwanted sinusoids and noise, prefiltering methods are presented using FIR or IIR filter matrices prior to an SVD and subspace-based method. A theoretical analysis on special cases of one and two exponentially damped sinusoids is given, which gives the increment of the SNR obtained from prefiltering and reveals the relationship between the singular values/vectors of the prefiltered and original data matrices.

Secondly, an enhancement algorithm, based on the minimum variance estimation method, is presented for cleaning up noisy data before estimating signal parameters using a subspace-based method. This algorithm is superior to Cadzow's and non-enhanced algorithms in both resolution and accuracy performance. The convergence of these enhancement procedures is found to be unnecessary when followed by an SVD and subspace-based estimator such as HTLS, offering drastic reduction in the computational cost. The improvement in resolution and computational efficiency makes the minimum variance enhanced estimation method well suited to the quantitative time-domain analysis of NMR measurement data.

Thirdly, the structure-preserving total least norm technique, called STLN, and the structured total least squares technique for complex-valued Hankel data matrices are presented and discussed. Applying STLN as signal enhancement prior to the estimation approach of HTLS results in a more accurate estimator, HTLN, which removes corruption noise effectively and improves resolution and parameter accuracy, especially, the standard deviations of the parameters.

Finally, prior knowledge of some known signal poles is incorporated into parameter estimators like HTLS and HTLN, etc. The remarkable advantages of the new methods are confirmed through quantification of both simulated and *in vivo* NMR signals, especially when the known signal poles are close to the unknown ones in the spectrum. Furthermore, the incorporation of prior knowledge is interpreted in terms of prefiltering operation, which enables intuitive analysis in the frequency domain.

List of Abreviations

Abreviation	Meaning
SVD	singular value decomposition
LS	least squares
TLS	total least squares
LCTLS	linearly-constrained total least squares
NMR	nuclear magnetic resonance
FID	free induction decay
^{31}P	phosphorus with atomic mass 31
PDE	phospho-diesters
PME	phospho-monoesters
P_i	inorganic phosphate
NTP	nucleotide triphosphate
ATP	adenosine triphosphate
NAD	nicotine adenine dinucleotide
DFT	discrete Fourier transform
FFT	fast Fourier transform
SNR	signal-to-noise ratio
RMSE	root mean-squared error
CR	Cramér-Rao (bound)
HTLS	Hankel TLS (technique)
HSVD	Hankel SVD (technique)
VARPRO	variable projection (technique)
LP	linear prediction (technique)
FLP	forward linear prediction (technique)
BLP	backward linear prediction (technique)
MP	matrix pencil (technique)
FIR	finite impulse response
IIR	infinite impulse response
$HTLS_F$	HTLS prefiltering (technique)
MV	minimum variance (method)
CA	Cadzow's (method)
CTLS	constrained total least squares (technique)
STLS	structured total least squares (technique)
STLN	structured total least norm (technique)
HTLN	STLN-enhanced HTLS (technique)

Abreviation	Meaning
PK	prior knowledge
LTI	linear time-invariant
a.u.	arbitrary units

List of Important Notations

Notation	Meaning
j	$\sqrt{-1}$
x	scalar
$\lfloor x \rfloor$	truncate a scalar x to a closest smaller integer
$\mathbf{x}$	vector
$\mathbf{X}$	matrix
$\|\mathbf{X}\|$	unitarily invariant norm, say, $\| \|_F$ and $\| \|_2$, of matrix $\mathbf{X}$
$\mathbf{X}^*$	conjugate of matrix $\mathbf{X}$
$\mathbf{X}^T$	transpose of matrix $\mathbf{X}$
$\mathbf{X}^H$	Hermitian conjugate of matrix $\mathbf{X}$
$\mathbf{X}^\uparrow$	matrix $\mathbf{X}$ of which the top row is deleted
$\mathbf{X}_\downarrow$	matrix $\mathbf{X}$ of which the bottom row is deleted
$\mathcal{R}(\mathbf{X})$	range of matrix $\mathbf{X}$
$\mathcal{N}(\mathbf{X})$	null space of matrix $\mathbf{X}$
$\mathbb{R}^{L\times M}$	set of all $L \times M$ real-valued matrices
$\mathbb{C}^{L\times M}$	set of all $L \times M$ complex-valued matrices
$\overset{\circ}{x}_n$	the nth noise-free data sample
e_n	the nth noise sample
x_n	the nth noisy data sample such that $x_n = \overset{\circ}{x}_n + e_n$
$\tilde{x}_n$	the nth filtered data sample
K	number of exponentials in the data model
N	number of data samples
L	number of rows in matrix $\mathbf{X}$
M	number of columns in matrix $\mathbf{X}$
f_k	frequency of the kth exponential in Hz
d_k	damping factor of the kth exponential in rad/s
a_k	amplitude of the kth exponential in a.u.
$\varphi_k(\phi_k)$	phase of the kth exponential in radians (degrees, °)
c_k	complex-valued amplitude of the kth exponential
z_k	signal pole of the kth exponential
f_s	sampling frequency
Δt	sampling time interval
t_n	the nth time sample

Notation	Meaning
$\mathbf{U}$	unitary matrix of left singular vectors $\mathbf{u}_i$
$\mathbf{V}$	unitary matrix of right singular vectors $\mathbf{v}_i$
$\mathbf{\Sigma}$	diagonal matrix of singular values σ_i in decreasing order of magnitude
σ_ν	standard deviation of real or imaginary part of white noise
ν	standard deviation of white noise ($\nu = 2\sqrt{\sigma_\nu}$, for complex-valued white noise)
$\mathbf{H}$	filter matrix
$\mathbf{W}, \mathbf{D}$	weighting matrices
$\mathcal{X}(m)$	discrete Fourier transform of data vector $\mathbf{x}$
$\mathrm{X}(\mathrm{z})$	z-transform of data vector $\mathbf{x}$

Contents

Acknowledgments i

Abstract ii

Abreviations iv

Notations vi

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Parameter estimation in NMR spectroscopy . . . 3
- 1.3 Parameter estimation in other applications . . . 8
- 1.4 Thesis outline . . . 9

2 Subspace-based Parameter estimation 13
- 2.1 Parameter estimation techniques . . . 13
  - 2.1.1 Time-domain methods versus frequency-domain methods 14
  - 2.1.2 Noninteractive methods versus interactive methods . . . 15
- 2.2 Subspace-based parameter estimation . . . 16
  - 2.2.1 SVD and signal subspace estimation . . . 16
  - 2.2.2 HTLS: a subspace-based parameter estimation algorithm 20
- 2.3 Monte-Carlo simulation procedure . . . 23

3 Prefiltering techniques 25
- 3.1 Hankel matrix and filtering . . . 26
- 3.2 FIR and IIR filter matrices . . . 29
  - 3.2.1 FIR filter matrix: $\mathbf{H}_{\text{FIR}}$ . . . 29
  - 3.2.2 IIR filter matrix: $\mathbf{H}_{\text{IIR}}$ . . . 32
- 3.3 Prefiltered HTLS . . . 35

- 3.4 SVD relationships between the prefiltered and original data matrices . . . 37
  - 3.4.1 Definitions and implications . . . 37
  - 3.4.2 SVD expressions of $\mathbf{X}$ and $\bar{\mathbf{X}}$ in terms of signal parameters 40
  - 3.4.3 SVD relationships between $\mathbf{X}$ and $\bar{\mathbf{X}}$ . . . 41
- 3.5 SNR relationships between the prefiltered and original data matrices . . . 50
- 3.6 HTLS_F prefiltering . . . 52
- 3.7 Simulations . . . 54
- 3.8 Conclusions . . . 65

4 Minimum variance signal enhancement 67
- 4.1 Introduction . . . 68
- 4.2 Enhancement algorithms . . . 69
  - 4.2.1 Cadzow's algorithm . . . 70
  - 4.2.2 Minimum variance algorithm . . . 71
  - 4.2.3 Composite property mapping . . . 72
- 4.3 Computational complexity . . . 73
- 4.4 Simulation results . . . 76
- 4.5 Application to real-world NMR signals . . . 87
- 4.6 Conclusions . . . 90

5 Structured TLN and structured TLS in signal enhancement 92
- 5.1 Formulation of STLN and STLS . . . 93
- 5.2 STLN and STLS algorithms applied to complex Hankel/Toeplitz data matrices . . . 95
  - 5.2.1 The STLN algorithms applied to complex Hankel data matrices . . . 95
  - 5.2.2 The STLS algorithm applied to complex Hankel data matrices . . . 101
  - 5.2.3 Discussion . . . 104
- 5.3 New parameter estimator HTLN . . . 106
- 5.4 Numerical comparison of different algorithms . . . 109
  - 5.4.1 Optimality of the STLN, STLS solutions . . . 109
  - 5.4.2 Simulation results . . . 113
- 5.5 Conclusion . . . 117

6 Incorporation of prior knowledge of known signal poles 120
- 6.1 Brief review of some parameter estimators . . . 121
- 6.2 Prior-knowledge-incorporating estimators . . . 123

6.3 Simulation results 131
6.4 Conclusion 139

7 More analysis of HTLS(PK) with application to NMR 141
7.1 HTLS(PK) and prefiltering 141
7.2 Quantification of simulated signals 145
7.2.1 Comparison of HTLS(PK) and prefiltered HTLS 145
7.2.2 Enhancement of HTLS(PK) 147
7.2.3 Computational efficiency of HTLS(PK) 147
7.2.4 Sensitivity of HTLS(PK) w.r.t. the accuracy of given poles 149
7.3 Quantification of a real-world NMR signal 154
7.4 Conclusion 159

8 General Conclusions 160
8.1 Summary 160
8.2 Suggestions for further research 164

A Simulated signals 166

B Publications 176

List of tables 179

List of figures 182

Bibliography 187

Chapter 1

Introduction

In this opening chapter, we first motivate our study of subspace-based estimation techniques. The data model and its parameters are described thereafter. Finally we give a chapter-by-chapter outline of the dissertation.

1.1 Motivation

Parameter estimation of signals modeled as a sum of **exponentially damped sinusoids** embedded in noise is a kernel problem in many signal processing applications, including direction-of-arrival (DOA) estimation [1][2] in sensor systems such as radar, sonar, and electronic surveillance; high-resolution spectral analysis of time series in speech processing and biomedical signal processing [3][4]; the study of transient signals [5], etc. The problem is one of taking measurements and estimating the parameters that describe the system under study, where the functional form of the underlying signal is assumed to be known as a sum of exponentially damped sinusoids.

With the development of the singular value decomposition (SVD) [6] and other numerical techniques, tremendous research efforts have been put on developing **subspace-based parameter estimation** methods. Via the SVD of a properly structured noisy data matrix, a subdivision into a "signal" and a "noise" subspace is possible in subspace-based approaches. The class of SVD and subspace-based methods becomes more and more attractive in parameter estimation problems due to its efficiency and ease of operation (with little human involvement).

The study of subspace-based parameter estimation in this thesis is greatly motivated by the problem of **spectral estimation in Nuclear Magnetic**

Resonance (NMR) spectroscopy, where the measurement data are considered as impulse responses often modeled as complex-valued exponentials corrupted by additive white noise. The objective is to estimate from measurements a set of constant parameters upon which the impulse response modeled as a sum of exponentials depends. Although this model may be inadequate for signals containing unknown non-exponential components, the quantification of these signals can be accommodated by the SVD and subspace-based methods which decompose each non-exponential spectral component into a sum of exponentially damped sinusoids, in contrast to nonlinear least squares methods. Furthermore, the automation of subspace-based methods is especially important for the clinical use of NMR spectroscopy.

In real-world applications, the comprised exponentials are typically grouped together, very close to each other and difficult to resolve. Very often, the signal-to-noise ratio (SNR) is not sufficiently high due to the limitations of the experimental set-up. In some cases, the parameters have to be estimated from short data records. Yet high-resolution and accurate (low-bias, low-variance) estimates are desired in order to retrieve relevant information. Thus the aim of this thesis is to investigate parameter estimation techniques for the complex-valued exponential model mentioned above. The features of the different algorithms to be discussed cover theoretical foundations, computational efficiency of the implementations, and in particular, **resolution and accuracy of the estimated parameters**, which is of crucial importance to the information acquisition in many real-world applications.

One way to improve the resolution and accuracy of the SVD and subspace-based parameter estimation is to apply a "cleaning" process, or what we call **signal enhancement**, to remove noise, measurement distortion, or even theoretical mismatch present in the given data set, before extracting parameters. The enhancement is accomplished through the process of slightly modifying the measured signal so that the modified signal possesses certain attributes or properties and better represents the underlying signal than the original measured signal.

The existing SVD and subspace-based estimation methods do not accommodate prior knowledge and yields automatically all the parameters when fitting noisy data sets composed of K exponentially damped sinusoids. However, in some applications such as NMR spectroscopy, some **prior knowledge** might be available, or only some of the exponentials (fewer than K) are of interest. The resolution and accuracy can be further improved if prior knowledge is incorporated or interesting components are emphasized.

In our study, various parameter estimation algorithms are developed and tested using simulated signals consisting of exponentially damped sinusoids in additive white noise, as well as real-world NMR measurement data sets, to demonstrate their performance. It is to be noted that these estimation algorithms, although applied only to NMR spectroscopy in this dissertation, can also be applied, and are in fact of great value, to many other signal processing applications.

1.2 Parameter estimation in NMR spectroscopy

NMR spectroscopy is a routinely used method to monitor metabolic changes in living organisms in a fully non-invasive fashion. With NMR spectroscopy, a lot of biochemical information, such as the structure, chemical environment, mobility, concentration and reaction rates of the atoms and molecules under investigation, is available which is not accessible in any other way.

Basic principles of NMR [9][10][11]

The principle of NMR is based on a quantum mechanical property of an atomic nucleus, called spin, which denotes the spinning of the nucleus around its own axis. In the absence of an external magnetic field, the nuclei will be spinning in random orientations, with zero net magnetization of the whole sample. Once an external magnetic field is applied, the spinning state is different, associated with energy levels. By the uptake of an energy quantum, which is equal to the energy difference between two spinning states, the nucleus is excited specifically from its lower energy level to its higher level. The energy difference ΔE between the two spin states is directly proportional to the magnitude of the applied field [8][9]:

$$\Delta E = \gamma \hbar B_0 \tag{1.1}$$

with B_0 the magnitude of the applied magnetic filed $\vec{\mathbf{B}}_0$, $\hbar$ the Planck constant divided by 2π and γ the so-called gyromagnetic ratio.

However, the transition between the two spin states must be induced by an electromagnetic radiation at Larmor frequency ω_0, which is the precessing angular frequency of nuclei around the applied field:

$$\Delta E = \hbar \omega_0 \tag{1.2}$$

Equating (1.1) and (1.2), we have

$$\omega_0 = \gamma B_0$$

Hence for a given magnetic field strength B_0, nuclei of atoms in different molecular structures, experiencing different chemical environments, can be distinguished by their different Larmor frequencies.

The measurement of an NMR signal requires the perturbation of the system from its equilibrium. Suppose an oscillating magnetic field pulse (short and intense) is applied, with the frequency identical to the Larmor frequency of atoms under study. The nuclei will immediately get excited, absorbing a fraction of the pulse energy. Upon removal of the magnetic field, the system will return to its equilibrium state and the nuclei relax, emitting some energy in a form of a weak electromagnetic radiation which decays with time and is called "*Free Induction Decay*" or *FID*. These electromagnetic waves are detected, picked up, fed to a receiver and stored for further processing and analysis. The obtained measurement data carry highly specific information about the chemical and physical state of the nuclear surroundings in the sample:

- The resonance frequencies of the nuclei enable the distinction of the states of nuclei, since nuclei in different environments give rise to signals at different frequencies.
- The magnitude of the FID at time zero is proportional to the total number of spins, which is a direct measure for the concentration of the molecules under investigation.
- The decay of the measured NMR signal is characterized by the spin-spin relaxation time T_2, which is the time required for the transverse component of the magnetization vector to decrease to a fraction of its original value. This provides more information since different molecules, as well as the same molecules but in different tissues, exhibit different relaxation times due to different macromolecular environments.
- The splitting effect of spectral lines is a result of interaction between the neighboring nuclei. The parameters of the group exhibit special interrelations which reflect the nuclear interaction.

The finally acquired signal, however, will be perturbed by thermal noise from the sample and some noise from the electronic components.

An example of an FID signal, measured from the electromagnetic radiation, is displayed in Fig. 1.1. Clearly, the signal contains contributions from many different metabolites and is perturbed by noise.

Model function [11][12][13]

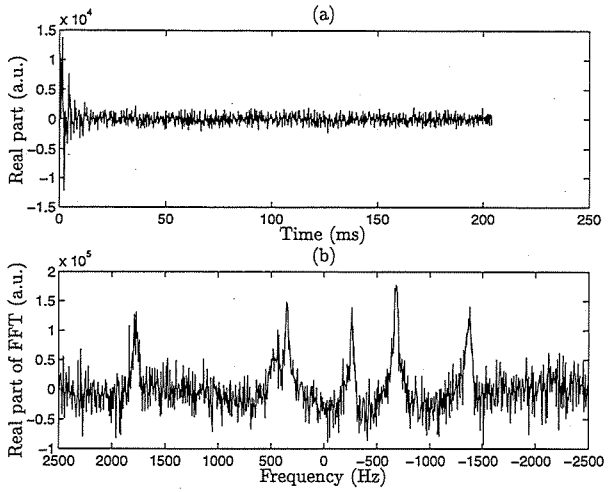


Figure 1.1: An example of an *FID* signal measured from an NMR experiment. (a) Time-domain or measurement-domain signal, (b) Signal spectrum. a.u. means arbitrary unit.

The measured FID signals are collected as complex-valued data since the magnetization is measured in two perpendicular directions. Most of the FID signals are of Lorentzian lineshape (exponential damping) and can be modeled as a sum of exponentially damped complex-valued exponentials as follows

$$x(t_n) = \sum_{k=1}^{K} a_k e^{j\varphi_k} e^{(-d_k + j2\pi f_k)t_n} + e(t_n) \qquad n = 0, \ldots, N-1 \quad (1.3)$$

where $j = \sqrt{-1}$, K is the number of exponentials in the spectrum, also called the model order, and $e(t_n)$ is complex white Gaussian noise. $t_n = t_0 + n\Delta t$ is the time lapse between the effective time origin and sample $x(t_n)$, where Δt is the time interval between observations (or $f_s = 1/\Delta t$ is the sampling frequency) and t_0 the spectrometer dead time. For simplicity, we always assume $t_0 = 0$

in the estimation algorithms, since the estimated parameters can always be extrapolated to the time origin once t_0 is found in practical NMR spectroscopy. The nonlinear parameters f_k and d_k represent the frequency and damping factors while the linear parameters a_k and φ_k are the amplitude and phase of the kth exponential. The physical meaning of these parameters in NMR spectroscopy can be described as follows:

- The frequency of a spectral component is characteristic of the particular metabolite.
- The damping factor of a spectral component corresponds to the inverse of the spin-spin relaxation time T_2 of the metabolite.
- The amplitude is the most important parameter since it is directly proportional to the concentration of the contributing nuclei, or the concentration of the metabolite under investigation.
- $\varphi_k = \varphi^{(0)} + \varphi_k^{(e)}$, in which $\varphi^{(0)}$ is the zero-order phase, whereas $\varphi_k^{(e)}$ represents extra degrees of freedom that may be required under certain experimental conditions (usually all $\varphi_k^{(e)}$ are zero).

To facilitate the analysis using signal processing tools, such as the z-transform, and the formulation and derivation of the estimation algorithms, the above data model (1.3) for $t_n = n\Delta t$ is written as

$$\begin{aligned} x_n &= \sum_{k=1}^{K} a_k e^{j\varphi_k} e^{(-d_k + j2\pi f_k)n\Delta t} + e_n \\ &= \sum_{k=1}^{K} c_k z_k^n + e_n \\ &= \overset{\circ}{x}_n + e_n \qquad\qquad n = 0, \ldots, N-1 \end{aligned} \tag{1.4}$$

where $\overset{\circ}{x}_n, n = 0, \ldots, N-1$ is the signal-only data, $c_k = a_k e^{j\varphi_k}$ is called the kth complex-valued amplitude and $z_k = e^{(-d_k + j2\pi f_k)\Delta t}$ the kth pole of the signal. Throughout this thesis, the signal poles are assumed distinct.

Procedure of parameter estimation

Fig. 1.2 illustrates a typical estimation procedure implemented in this thesis. The given data are time-domain measurements, $x_n, n = 0, 1, \ldots, N-1$, and the results are parameter estimates, $f_k, d_k, a_k, \varphi_k, k = 1, 2, \ldots, K$ defined

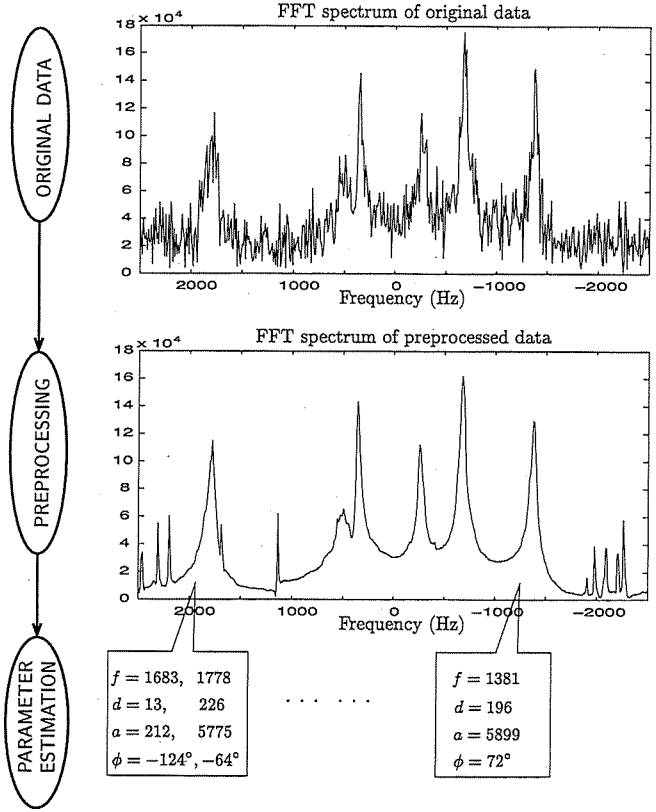


Figure 1.2: Typical parameter estimation procedure.

in (1.4). To visualize signal features, the FFT spectra are plotted, although the data are processed in the time domain in our study. The preprocessed spectrum is the FFT spectrum of the enhanced signal, using the minimum variance enhancement technique discussed in Chapter 4. In fact, the preprocessing step here can be other enhancement techniques or prefiltering techniques developed in this thesis. The estimated parameters are obtained using the HTLS estimation algorithm 1 given in the next Chapter, without incorporating prior knowledge.

1.3 Parameter estimation in other applications

Although our parameter estimation of a sum of exponentials is motivated by and treated in the context of NMR spectroscopy in this thesis, it is generally applicable to a wide variety of problems. For illustrative purposes, we cite the following two applications, where similar parameter estimation problems occur.

Identification of natural modes in transient signals [14][15][5]

Many electromagnetic scatterers and antennas may be modeled as linear time invariant systems represented in terms of their s-plane response. These systems are continuously spatially distributed and thus have an infinite number of modes or resonances. Yet there will be only a finite number of strong modes due to the band-limited nature of physical scatterers and anti-aliasing filters.

In the determination of the natural modes of an electromagnetic scatterer, the measured time series contain desired information, noise, and quite often transient components introduced by the excitation source or measuring equipment. An exponential parametric signal model is often taken for such an observed time series, using a finite and fixed model order:

$$x_n = \sum_{k=1}^{K} a_k e^{s_k n \Delta t} = \sum_{k=1}^{K} a_k z_k^n$$

where x_n, s_k, z_k are complex, and $z_1, z_2, \ldots, z_K$ contain both natural modes of interest and transient components. This is actually the same model function as (1.4), and accurate estimation of the natural modes in this model function is very useful in target identification.

Direction-of-arrival (DOA) estimation [1][2]

In the application of DOA estimation, data from an array of sensors are collected and the objective is to locate point sources assumed to be radiating energy that is detectable by the sensors. Based on a few reasonable assumptions, the radiation impinging on the array is in the form of a sum of plane waves. For simplicity, it is further assumed that the problem is planar and that all sources lie in a half-space, thus reducing the location parameter space to a single-dimensional subset of $\theta_k \in [0, \pi]$, where θ_k is the DOA of the kth source.

For narrowband signals, which are assumed to have the same known center frequency ω_0, the kth signal can be written as the real part of

$$s_k(t) \approx a_k(t) \exp[j(\omega_0 t + \varphi_k(t))] \approx s_k(t) \approx a_k \exp[j(\omega_0 t + \varphi_k)]$$

The approximation here is due to the slow variation of $a_k(t)$ and $\varphi_k(t)$ from narrowband assumption. In this way, the effect of a time delay τ on the received waveforms is simply a phase shift, i.e.,

$$s_k(t - \tau) = s_k(t) \exp(-j\omega_0 \tau)$$

The result is that $x_m(t)$, the complex-valued signal output of the kth sensor at time t, can be written as

$$x_m(t) = \sum_{k=1}^{K} H_m(\theta_k) s_k[t - \tau_m(\theta_k)] = \sum_{k=1}^{K} H_m(\theta_k) \exp[-j\omega_0 \tau_m(\theta_k)] s_k(t)$$

where $\tau_m(\theta_k)$ is the propagation delay between a reference point and the mth wavefront impinging on the array from direction θ_k, the term $H_m(\theta_k)$ is the corresponding sensor complex-valued response (gain and phase) at frequency ω_0, and K point sources are assumed.

Adding measurement noise $e(t)$ and sampling the signals uniformly at $t_n = n\Delta t$, the above model takes basically the same form as (1.3), although the meanings and functions of variables are different. As a matter of fact, the problem of DOA estimation is most frequently referred to in literatures on parameter estimation of a sum of exponentials using subspace-based techniques.

1.4 Thesis outline

In **Chapter 2**, we first give an overview of various parameter estimation techniques. Then, basic principles of SVD and subspace-based parameter estimation techniques, as well as the HTLS estimation algorithm, are reviewed in order to clarify the description of new algorithms based on HTLS in subsequent chapters.

Chapter 3 deals with the problem of frequency-selective estimation. As reviewed in **Chapter 2**, the HTLS estimation method is based on the TLS technique and SVD filtering, which filters out part of the noise by truncating the SVD of the Hankel data matrix to rank K and discarding the non-significant singular values and vectors. When such SVD filtering is not satisfactory at low SNR, or when only some (fewer than K) of the spectral components that reside in a selected frequency region are of interest, we need to further compress the noise and enhance the sinusoids of interest. To emphasize the sinusoids of interest and suppress the unwanted sinusoids and noise, prefiltering methods are presented in **Chapter 3** using FIR and IIR filter matrices prior to an SVD and subspace-based method, such as HTLS. Subsequently, a theoretical analysis of special cases of one and two exponentially damped sinusoids is given, which gives the increment of the SNR obtained from prefiltering and reveals the relationship between the singular values/vectors of the prefiltered and original data matrices. Furthermore, the FIR or IIR prefiltering methods are combined with the HTLS prefiltering technique, which is similar to the HSVD prefiltering technique proposed by Pijnappel et al. [32] for Kung's method. Finally, results of computer simulations are presented to confirm the theoretical predictions.

This chapter is based on [33], [34], [36] and [37].

In **Chapter 4**, we summarize and study a class of SVD-based signal enhancement algorithms, including Cadzow's enhancement [56][57][58] and the minimum variance enhancement [29][61], in the context of NMR spectroscopic signal processing. Based on SVD, these algorithms compute estimates of the signal-only data matrix and then restore the Hankel structure of the computed estimates to clean up noisy data before estimating signal parameters using a subspace-based method. The resolution performance and convergence behavior of these enhancement procedures is studied when followed by an SVD and subspace-based estimator such as HTLS. As demonstrated in this chapter, the minimum variance algorithm is superior to Cadzow's and non-enhanced algorithms in resolution and accuracy performance, as well as computational efficiency.

This chapter is based on [38], [41], [39], [40], and [42].

In **Chapter 5**, we study another class of iterative methods, STLN and STLS, for the problem of approximating Hankel structured rank-deficient data matrices. First, the solution and outline of the STLN and STLS algorithms is given for complex-valued data sets. Then, a new estimator, HTLN, is presented for our problem of exponential data fitting, which starts from the same Hankel data matrix but uses the STLN technique prior to the estimation approach of HTLS. The optimality of the STLN and STLS algorithms, as well as that of the minimum variance and Cadzow's algorithms, are studied in two examples for which the global minimum can be found explicitly. Finally, the enhancement algorithms, called Cadzow's (CA) method, minimum variance (MV), STLS, STLN, and the corresponding enhanced estimators CA-HTLS, MV-HTLS, STLS-HTLS, and HTLN are compared in terms of resolution and accuracy, as well as computational efficiency, based on computer simulations.

This chapter is based on [43], [44] and [45].

The previous chapters deal with different *techniques* of filtering out or cleaning up noise present in the obtained data. In **Chapter 6**, we incorporate *prior knowledge* (PK) to improve our parameter estimates. New methods, such as HTLS(PK), HTLN(PK), etc., that incorporate prior knowledge of some known signal poles into the parameter estimators HTLS and HTLN, etc., are presented. The benefits of the new methods compared to the old methods that allow or do not allow the incorporation of prior knowledge are confirmed through quantification of simulated NMR signals, especially when the known signal poles are close to the unknown ones in the spectrum.

This chapter is based on [46] and [47].

Chapter 7 gives a detailed analysis of the HTLS(PK) method and more quantification results using both simulated and *in vivo* NMR signals. By relating the HTLS(PK) method to the prefiltered HTLS method presented in **Chapter 3**, we provide a means to interpret the incorporation of prior knowledge in terms of a prefiltering operation, which facilitates the intuitive analysis in the frequency domain. The enhancement and efficiency of HTLS(PK), as well as the sensitivity of the HTLS(PK) method with respect to the accuracy of the given poles, are studied. This chapter is based on [48].

Finally, a brief summary of the main features of the new algorithms presented in this thesis, and some suggestions for future research, are given in **Chapter 8**.

To better visualize the structure of this thesis and the application field of the new estimation techniques developed in this thesis, we illustrate in Fig. 1.3 these subspace-based estimation techniques, as well as when and how to use them.

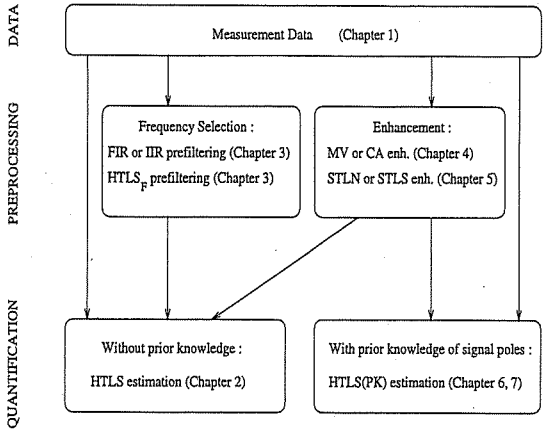


Figure 1.3: Relationships between the new techniques developed in this thesis from an application's point of view.

Chapter 2

Subspace-based Parameter estimation

In this chapter, an overview of various parameter estimation techniques is first given. Then, basic principles of SVD and subspace-based parameter estimation techniques are reviewed, followed by a description of the concise theory of a subspace-based parameter estimation method, called HTLS, with which all the new methods developed in this thesis are compared. Finally, general procedures of Monte-Carlo simulations used for evaluating different estimation methods are described.

2.1 Parameter estimation techniques

In the past years, tremendous effort has been put in developing parameter estimation techniques for the problem modeled in (1.4). This study aims at developing more accurate estimation methods using modern numerical techniques and based on the subspace separation. Before reviewing the existing subspace-based estimation algorithms, which is a noninteractive time-domain method, other methods like spectral estimation in the frequency domain and interactive non-linear data-fitting in the time domain are introduced for comparison.

2.1.1 Time-domain methods versus frequency-domain methods

Traditional methods for resolving closely-spaced sinusoids use the power spectrum or the discrete (fast) Fourier transform (DFT, FFT) of the measurement (time-domain) data sets. It is well known that these discrete transforms to the frequency domain limit the frequency resolution to the reciprocal of the observation interval length. The modern approach to overcoming this fundamental limit and achieving high resolution from a short data segment is by using prior knowledge of the model function formulation and fitting this model in the time domain. The problem of spectral estimation is then reduced to that of estimating the model parameters. In the following, we show more restrictions of frequency-domain methods and advantages of time-domain methods in the context of NMR spectroscopy.

For data modeled in (1.4), the FFT spectrum displays K peaks and the sharpness of these peaks is governed by the damping factor d_k. The larger d_k the wider the peak. As $d_k \to 0$ we approach the line spectrum of a non-damped sum of sinusoids. One of the frequency-domain methods in NMR spectroscopy consists of determining frequencies, amplitudes and damping factors from respectively the spectral peak position, area and width at half height.

Besides the problem of frequency resolution pointed out earlier, other problems arise when applying the FFT to measurement data sets in real-world applications. In NMR spectroscopy, for example, there is frequently an annoying baseline in the FFT spectrum, or a broad hump that can be modeled as a sum of noise peaks with large damping factors. The position of the baseline is critical for a correct quantification by means of a frequency-domain method, such as non-model function fitting (measurement of peak locations, cutting and weighing, triangulation, integration) and model function fitting [13]. In model function fitting, a mathematical lineshape function, for example, a sum of Lorentzian lines as in our model (1.3), is fitted to the spectral data. Due to the mathematically complicated form of the spectral model function, this is a very time-consuming procedure. A major problem arises if the data are incomplete at either end or the data set is truncated so that it does not completely cover the decay of the damping parameters. This leads to leakage of peaks into the sidelobes of the transform such that the FFT estimate of the spectrum deviates from the "true" spectrum. There are various filtering methods to reduce this effect, e.g., multiply the decaying FID by a bell-shaped window function. More methods have been devised to deconvolve artifacts originating from the FFT of data, such as linear prediction(LP)-based deconvolution, the maximum entropy method, etc. (see e.g. [3] and the references therein). However, the highest precision is usually attainable only if the ultimate step of the estimation of parameters is applied to the original data in the measurement domain.

In the last few years, there is an increasing demand to perform model function fitting in the measurement domain, or time domain, using signal processing tools. Generally, time-domain algorithms are less time consuming, in view of the mathematically straight-forward decay functions. But the main advantage of processing the data in the time domain over frequency domain is that absence of data points at either end of the data record need not complicate a quantification. This allows truncation of initial data points without affecting the signal parameters and avoids the requirement of baseline correction and deconvolution in practical applications. In addition, time-domain methods are evolving rapidly and resulting in superior precision by making use of results of numerical analysis and signal processing.

In view of the above features, our data quantification is carried out directly in the time domain in the present dissertation. FFT spectra are used here only for visualizing and interpreting time-domain data, as well as an aid for identifying particular areas of interest.

2.1.2 Noninteractive methods versus interactive methods

Time-domain methods can be divided into two classes: interactive and noninteractive methods.

Interactive methods are not restricted to the model function (1.3). But they require significant involvement of the user, such as choosing the proper model function, providing good starting values, gathering prior knowledge, and judging the convergence to the correct (global) minimum. Clearly, this may be rather expensive in terms of human effort. Note however that the imposition of prior knowledge usually has a beneficial effect on the stability of the convergence of an iterative process and on the resolution and accuracy of the estimated parameters. An example of this class is the VARPRO estimation method [16][17].

Noninteractive methods are always "black box" methods, that is, they require only minimal user involvement. The model function is restricted to (1.3) and minimal expert knowledge is required. Although this model may be inadequate for signals containing non-exponential components, in which case the quantification is mathematical rather than physical, the fit can most often be made quite well. This is achieved by decomposing each deviating non-exponential spectral component into a sum of exponentially damped sinusoids [18]. A price to pay for the relative ease of operation is that the computed parameter estimates are less accurate than those obtained with more elaborate

interactive methods, especially at low SNR's, since little prior knowledge can be included. Examples of noninteractive methods are SVD and subspace-based methods, and Evans-Fischl-based methods [19]. In this study we seek to improve the resolution and accuracy of subspace-based estimation algorithms of this class.

2.2 Subspace-based parameter estimation

By subspace-based parameter estimation, we mean that the noisy signal space is subdivided into a "signal" and a "noise" subspace using SVD or other numerical techniques. There are various subspace-based estimation methods, depending on how these subspaces are determined, and how the pole information is extracted from the subspace and its assumed properties. Some well-known estimation methods of this type are the linear prediction-based methods [20][21], the matrix pencil (MP) method [22] and Kung et al.'s state-space-based method [23] (which is called HSVD in the NMR literature [4]). All these methods start from a Hankel/Toeplitz data matrix and find the "signal" subspace by truncating the SVD of this matrix to rank K, the number of exponentials in the underlying signal. The accuracy of these methods is improved by making use of the total least squares (TLS) technique [24][6]. The forward and backward LP methods, based on TLS and called TLS-FLP and TLS-BLP, have been proposed in [25], while [26][27][28] describe how to use the TLS principle in Kung et al.'s state-space-based method. The latter method is called the Hankel TLS (HTLS) method in NMR spectroscopy.

Since HTLS manifests the best performance, including resolution and parameter accuracy, among these techniques [28], this study starts with the HTLS estimation method and tries to improve the resolution and accuracy of the HTLS parameter estimates via prefiltering, signal enhancement, incorporation of prior knowledge, etc. For clarity of exposition of the HTLS estimation method, as well as other subspace-based estimation techniques presented later, we reformulate the basic principles and the relationships between the SVD and signal subspace estimation [29][61][62], before introducing the HTLS estimation algorithm.

2.2.1 SVD and signal subspace estimation

This subsection is mainly based on De Moor's [29].

Assume that an $L \times M$ Hankel matrix

$$\mathbf{X} = \begin{bmatrix} x_0 & x_1 & \cdots & x_{M-1} \\ x_1 & x_2 & \cdots & \cdot \\ \vdots & \vdots & \vdots & \vdots \\ x_{L-1} & \cdot & \cdots & x_{N-1} \end{bmatrix} \quad L \geq M \geq K, N = L + M - 1 \qquad (2.1)$$

consisting of N uniformly sampled data points $x_n, n = 0, 1, \ldots, N-1$, and a model-order estimate K are given.

With the basic assumption of additive noise in (1.4), the data matrix $\mathbf{X}$ can be decomposed into two unknown matrices: an exact data matrix $\overset{\circ}{\mathbf{X}}$ and an additive perturbation matrix $\mathbf{E}$, so that

$$\mathbf{X} = \overset{\circ}{\mathbf{X}} + \mathbf{E}$$

where $\mathbf{X}, \overset{\circ}{\mathbf{X}}, \mathbf{E} \in \mathbb{C}^{L \times M}$, and typically, $\text{rank}(\mathbf{X}) = \text{rank}(\mathbf{E}) = \min\{L, M\}$, while $\text{rank}(\overset{\circ}{\mathbf{X}}) = K \leq \min\{L, M\}$. Computing the SVD of the exact rank-K matrix $\overset{\circ}{\mathbf{X}}$ yields:

$$\overset{\circ}{\mathbf{X}} = \overset{\circ}{\mathbf{U}} \overset{\circ}{\mathbf{\Sigma}} \overset{\circ}{\mathbf{V}}^H = [\overset{\circ}{\mathbf{U}}_1 \; \overset{\circ}{\mathbf{U}}_2] \begin{bmatrix} \overset{\circ}{\mathbf{\Sigma}}_1 & 0 \\ 0 & 0 \end{bmatrix} [\overset{\circ}{\mathbf{V}}_1 \; \overset{\circ}{\mathbf{V}}_2]^H \qquad (2.2)$$

where ^{H} denotes Hermitian conjugation or conjugate transposition, $\overset{\circ}{\mathbf{U}}_1 \in \mathbb{C}^{L \times K}$, $\overset{\circ}{\mathbf{U}}_2 \in \mathbb{C}^{L \times (L-K)}$, $\overset{\circ}{\mathbf{\Sigma}}_1 \in \mathbb{C}^{K \times K}$, $\overset{\circ}{\mathbf{V}}_1 \in \mathbb{C}^{M \times K}$, $\overset{\circ}{\mathbf{V}}_2 \in \mathbb{C}^{M \times (M-K)}$. In this way, the conjugated row space of $\overset{\circ}{\mathbf{X}}$ coincides with the column space of $\overset{\circ}{\mathbf{V}}_1$ [1], and the columns of $\overset{\circ}{\mathbf{V}}_2$ generate an orthogonal basis for the null space of $\overset{\circ}{\mathbf{X}}$ [2]:

$$\mathcal{R}(\overset{\circ}{\mathbf{X}}^H) = \mathcal{R}(\overset{\circ}{\mathbf{V}}_1), \quad \mathcal{N}(\overset{\circ}{\mathbf{X}}) = \mathcal{R}(\overset{\circ}{\mathbf{V}}_2)$$

Let $\check{\mathbf{U}}_1 \check{\mathbf{\Sigma}}_1 \check{\mathbf{V}}_1^H$ be the SVD of $(\overset{\circ}{\mathbf{X}} \overset{\circ}{\mathbf{V}}_1 + \mathbf{E} \; \overset{\circ}{\mathbf{V}}_1)$, and $\check{\mathbf{U}}_2 \check{\mathbf{\Sigma}}_2 \check{\mathbf{V}}_2^H$ be the SVD of $(\mathbf{E} \; \overset{\circ}{\mathbf{V}}_2)$. Then, under the following three conditions:

C1. The exact data matrix $\overset{\circ}{\mathbf{X}}$ should be orthogonal to the noise matrix $\mathbf{E}$ in the sense that $\overset{\circ}{\mathbf{X}}^H \mathbf{E} = \mathbf{0}$;

[1] $\mathcal{R}(\cdot)$ denotes the range of the matrix in brackets
[2] $\mathcal{N}(\cdot)$ denotes the null space of the matrix in brackets

C2. The matrices $\overset{\circ}{\mathbf{V}}_1$ and $\overset{\circ}{\mathbf{V}}_2$ must be orthogonal to each other in the inner product generated by the matrix $\mathbf{E}^H\mathbf{E}$: $\overset{\circ}{\mathbf{V}}_1^H\ \mathbf{E}^H\mathbf{E}\ \overset{\circ}{\mathbf{V}}_2 = \mathbf{0}$;

C3. The smallest singular value of $\breve{\mathbf{\Sigma}}_1$ must be larger than the largest singular value of $\breve{\mathbf{\Sigma}}_2$;

the rank K of $\overset{\circ}{\mathbf{X}}$ and the subspaces generated by $\overset{\circ}{\mathbf{V}}_1$ and $\overset{\circ}{\mathbf{V}}_2$ (the row and null space of $\overset{\circ}{\mathbf{X}}$) can be estimated consistently, while the original exact column space of $\overset{\circ}{\mathbf{X}}$, represented by $\overset{\circ}{\mathbf{U}}_1$, cannot be recovered from the SVD of $\mathbf{X}$, as shown below.

$$\begin{aligned}\mathbf{X} &= \overset{\circ}{\mathbf{X}} + \mathbf{E} \\ &= (\overset{\circ}{\mathbf{X}}\overset{\circ}{\mathbf{V}}_1 + \mathbf{E}\overset{\circ}{\mathbf{V}}_1)\overset{\circ}{\mathbf{V}}_1^H + (\mathbf{E}\overset{\circ}{\mathbf{V}}_2)\overset{\circ}{\mathbf{V}}_2^H \\ &= (\breve{\mathbf{U}}_1\breve{\mathbf{\Sigma}}_1\breve{\mathbf{V}}_1^H)\overset{\circ}{\mathbf{V}}_1^H + (\breve{\mathbf{U}}_2\breve{\mathbf{\Sigma}}_2\breve{\mathbf{V}}_2^H)\overset{\circ}{\mathbf{V}}_2^H \\ &= [\breve{\mathbf{U}}_1\ \breve{\mathbf{U}}_2]\begin{bmatrix}\breve{\mathbf{\Sigma}}_1 & \mathbf{0}\\ \mathbf{0} & \breve{\mathbf{\Sigma}}_2\end{bmatrix}\begin{bmatrix}\breve{\mathbf{V}}_1^H\overset{\circ}{\mathbf{V}}_1^H \\ \breve{\mathbf{V}}_2^H\overset{\circ}{\mathbf{V}}_2^H\end{bmatrix} \qquad (2.3)\end{aligned}$$

Suppose that the above three assumptions are all satisfied. Since $\mathcal{R}(\breve{\mathbf{U}}_1) = \mathcal{R}(\overset{\circ}{\mathbf{X}}\overset{\circ}{\mathbf{V}}_1 + \mathbf{E}\overset{\circ}{\mathbf{V}}_1)$ and $\mathcal{R}(\breve{\mathbf{U}}_2) = \mathcal{R}(\mathbf{E}\overset{\circ}{\mathbf{V}}_2)$, the assumptions C1 and C2 guarantee that $\overset{\circ}{\mathbf{V}}_1^H\ (\overset{\circ}{\mathbf{X}}^H + \mathbf{E}^H)\mathbf{E}\overset{\circ}{\mathbf{V}}_2 = \mathbf{0}$, i.e., $\breve{\mathbf{U}}_1^H\breve{\mathbf{U}}_2 = \mathbf{0}$. Considering the assumption C3, it is easy to see that Eq. (2.3) is an SVD of $\mathbf{X}$.

Further assume that $\mathbf{E}^H\mathbf{E}$ is a scalar multiple of the identity matrix

$$\mathbf{E}^H\mathbf{E} = \mu^2\mathbf{I}_M = L\nu^2\mathbf{I}_M \qquad (2.4)$$

where ν^2 is the variance of the independently and identically distributed zero-mean noise element in $\mathbf{E}$ asymptotically, as the number of measurements and hence $L \to \infty$. This is a special case of assumption C2. In this case, the SVD of $\mathbf{X}$ can be explicitly expressed in terms of the SVD of $\overset{\circ}{\mathbf{X}}$

$$\begin{aligned}\mathbf{X} &= \overset{\circ}{\mathbf{X}} + \mathbf{E} \\ &= \overset{\circ}{\mathbf{U}}_1\overset{\circ}{\mathbf{\Sigma}}_1\overset{\circ}{\mathbf{V}}_1^H + \mathbf{E}\overset{\circ}{\mathbf{V}}_1\overset{\circ}{\mathbf{V}}_1^H + \mathbf{E}\overset{\circ}{\mathbf{V}}_2\overset{\circ}{\mathbf{V}}_2^H \\ &= [(\overset{\circ}{\mathbf{U}}_1\overset{\circ}{\mathbf{\Sigma}}_1 + \mathbf{E}\overset{\circ}{\mathbf{V}}_1)(\overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K)^{-\frac{1}{2}}\quad \mathbf{E}\overset{\circ}{\mathbf{V}}_2\,\mu^{-1}] \\ &\quad \begin{bmatrix}\overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K & \mathbf{0} \\ \mathbf{0} & \mu\mathbf{I}_{M-K}\end{bmatrix}^{\frac{1}{2}}\begin{bmatrix}\overset{\circ}{\mathbf{V}}_1^H \\ \overset{\circ}{\mathbf{V}}_2^H\end{bmatrix}\end{aligned}$$

$$= [\mathbf{U}_1\ \mathbf{U}_2]\begin{bmatrix}\mathbf{\Sigma}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{\Sigma}_2\end{bmatrix}\begin{bmatrix}\mathbf{V}_1^H \\ \mathbf{V}_2^H\end{bmatrix} \quad \text{for } L \geq M \qquad (2.5)$$

Upon examination of these relationships, it can be concluded that under the assumptions we have made,

- The exact singular values of $\overset{\circ}{\mathbf{X}}$ can be calculated from
$$\overset{\circ}{\mathbf{\Sigma}}_1 = \sqrt{\mathbf{\Sigma}_1^2 - \mu^2\mathbf{I}_K}$$
- The row space of $\overset{\circ}{\mathbf{X}}$ can be recovered exactly, i.e., $\overset{\circ}{\mathbf{V}}_1 = \mathbf{V}_1$, $\overset{\circ}{\mathbf{V}}_2 = \mathbf{V}_2$
- The column space of $\overset{\circ}{\mathbf{X}}$ can not be recovered, but the canonical angles between the column space of $\overset{\circ}{\mathbf{X}}$, represented by $\mathcal{R}(\overset{\circ}{\mathbf{U}}_1)$, and the column space $\mathcal{R}(\mathbf{U}_1)$ generated by the first K left singular vectors of $\mathbf{X}$, can be computed. The cosine of these canonical angles are the diagonal elements of the diagonal matrix, denoted by $\mathbf{C}_c$ as follows:
$$\begin{aligned}\mathbf{C}_c &= \overset{\circ}{\mathbf{U}}_1^H\ \mathbf{U}_1 \\ &= \overset{\circ}{\mathbf{U}}_1^H\ (\overset{\circ}{\mathbf{U}}_1\overset{\circ}{\mathbf{\Sigma}}_1 + \mathbf{E}\overset{\circ}{\mathbf{V}}_1^H)(\overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K)^{-\frac{1}{2}} \\ &= \overset{\circ}{\mathbf{\Sigma}}_1\,(\overset{\circ}{\mathbf{\Sigma}}_1^2 + \mu^2\mathbf{I}_K)^{-\frac{1}{2}} \\ &= \overset{\circ}{\mathbf{\Sigma}}_1\,\mathbf{\Sigma}_1^{-1} \qquad (2.6)\end{aligned}$$
which characterizes the bias of the signal subspace estimate $\mathbf{X}$ and provides an insight as to how well the matrix $\mathbf{X}$ approximates the exact matrix $\overset{\circ}{\mathbf{X}}$.

Despite the fact that $\overset{\circ}{\mathbf{U}}_1$ cannot be consistently estimated from the SVD of $\mathbf{X}$, it is possible to find the estimate of $\overset{\circ}{\mathbf{X}}$, for example, in the least squares (LS) sense, from the SVD of $\mathbf{X}$ if all the three assumptions are satisfied and $\mathbf{E}^H\mathbf{E} = \mu^2\mathbf{I}$.

LS estimation of $\overset{\circ}{\mathbf{X}}$

Approximate $\mathbf{X}$ by a matrix $\mathbf{X}_{\mathrm{LS}}$ of rank K in the least squares sense:

$$\min_{\mathbf{X}_{\mathrm{LS}} \in \mathbb{C}^{L\times M}, \mathrm{rank}(\mathbf{X}_{\mathrm{LS}}) = K \leq \min\{L,M\}} \|\mathbf{X} - \mathbf{X}_{\mathrm{LS}}\|_F^2$$

Its classical solution is

$$\mathbf{X}_{\mathrm{LS}} = \mathbf{U}_1(\overset{\circ}{\Sigma}_1^2 + \mu^2\mathbf{I}_K)^{\frac{1}{2}}\overset{\circ}{\mathbf{V}}_1^H = \mathbf{U}_1\Sigma_1\mathbf{V}_1^H$$

which is simply a truncated SVD of $\mathbf{X}$ to rank K.

In the following presentation, we frequently use $\mathbf{U}_K\Sigma_K\mathbf{V}_K^H$ to denote such a rank-K truncation, where $\mathbf{U}_K = \mathbf{U}_1 \in \mathbb{C}^{L\times K}, \mathbf{V}_K = \mathbf{V}_1 \in \mathbb{C}^{M\times K}$ are truncated matrices containing the first K singular vectors of $\mathbf{U} \in \mathbb{C}^{L\times L}, \mathbf{V} \in \mathbb{C}^{M\times M}$, and $\Sigma_K = \Sigma_1 = \mathrm{diag}(\sigma_1, \cdots, \sigma_K) \in \mathbb{C}^{K\times K}$ is the truncated matrix of $\Sigma \in \mathbb{C}^{L\times M}$. Also, $\mathbf{U}_2$ and $\mathbf{V}_2$, the bases of orthogonal complements of $\mathcal{R}(\mathbf{U}_1)$ and $\mathcal{R}(\mathbf{V}_1)$, are represented by $\overline{\mathbf{U}}_K$ and $\overline{\mathbf{V}}_K$.

SVD truncation can be considered as filtering out part of the noise by discarding the nonprincipal singular values/vectors and finding the signal subspace basis $\mathbf{U}_K$ or $\mathbf{V}_K$. The success of SVD-based subspace methods critically depends on the assumptions C1, C2, C3, which in practice however are never satisfied exactly (except maybe for condition C3). A nice feature of the SVD however is its robustness with respect to (mild) violations of these conditions. If for instance $\| \overset{\circ}{\mathbf{X}}^H \mathbf{E}\|$ is small (where $\|.\|$ is any unitarily invariant norm), the SVD of $\mathbf{X}$ will still deliver good approximations. The smaller $\| \overset{\circ}{\mathbf{X}}^H \mathbf{E}\|$ gets, the better the approximations will be.

2.2.2 HTLS: a subspace-based parameter estimation algorithm

If the data are noise-free, then $\mathbf{X} = \overset{\circ}{\mathbf{X}} \in \mathbb{C}^{L\times M}$ in (2.1) has a Vandermonde decomposition of the following form:

$$\begin{aligned}\mathbf{X} = \overset{\circ}{\mathbf{X}} &= \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1^1 & z_2^1 & \cdots & z_K^1 \\ \vdots & & & \vdots \\ z_1^{L-1} & z_2^{L-1} & \cdots & z_K^{L-1} \end{bmatrix} \begin{bmatrix} c_1 & & & 0 \\ & c_2 & & \\ & & \ddots & \\ 0 & & & c_K \end{bmatrix} \\ &\quad \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1^1 & z_2^1 & \cdots & z_K^1 \\ \vdots & & & \vdots \\ z_1^{M-1} & z_2^{M-1} & \cdots & z_K^{M-1} \end{bmatrix}^T \\ &= \mathbf{S}_K\mathbf{C}_K\mathbf{T}_K^T \end{aligned} \tag{2.7}$$

where ^{T} denotes matrix transposition, $\mathbf{S}_K \in \mathbb{C}^{L\times K}$ and $\mathbf{T}_K \in \mathbb{C}^{M\times K}$ are Vandermonde matrices composed of the K signal poles, and $\mathbf{C}_K \in \mathbb{C}^{K\times K}$ is a diagonal matrix composed of K complex-valued amplitudes. One observation from the above Vandermonde decomposition is that the rank of such a signal-only Hankel matrix $\mathbf{X}$ is K according to the rank inequality [30] below

$$\begin{aligned} \mathrm{rank}(\mathbf{S}_K) &= \mathrm{rank}(\mathbf{C}_K) = \mathrm{rank}(\mathbf{T}_K) = K \\ \mathrm{rank}(\mathbf{S}_K\mathbf{C}_K\mathbf{T}_K^T) &\geq \mathrm{rank}(\mathbf{S}_K) + \mathrm{rank}(\mathbf{C}_K) + \mathrm{rank}(\mathbf{T}_K^T) - 2K \\ \mathrm{rank}(\mathbf{S}_K\mathbf{C}_K\mathbf{T}_K^T) &\leq \min\{\mathrm{rank}(\mathbf{S}_K), \mathrm{rank}(\mathbf{C}_K), \mathrm{rank}(\mathbf{T}_K^T)\} \end{aligned}$$

The rank-K property of $\mathbf{X}$ results in

$$\mathbf{X} \overset{\mathrm{SVD}}{=} \mathbf{U}\Sigma\mathbf{V}^H \overset{\mathrm{truncate}}{=} \mathbf{U}_K\Sigma_K\mathbf{V}_K^H \tag{2.8}$$

Another observation is that $\mathbf{S}_K, \mathbf{T}_K$ have the shift-invariant property, i.e.,

$$\mathbf{S}_K^{\uparrow} = \mathbf{S}_{K\downarrow}\mathbf{Z}_K, \quad \mathbf{T}_K^{\uparrow} = \mathbf{T}_{K\downarrow}\mathbf{Z}_K, \quad \mathbf{Z}_K = \mathrm{diag}(z_1, z_2, \ldots, z_K)$$

where the up (down) arrow stands for deleting the top (bottom) row, and $\mathbf{Z}_K \in \mathbb{C}^{K\times K}$ is a diagonal matrix composed of the K signal poles, $z_k, k = 1, \ldots, K$.

Equating the two decompositions (2.8) and (2.7), we have the shift-invariant property of $\mathbf{U}_K$ and $\mathbf{V}_K^*$ also:

$$\mathbf{U}_K^{\uparrow} = \mathbf{U}_{K\downarrow}\mathbf{Z}_K^{(U)}, \quad \mathbf{V}_K^{*\uparrow} = \mathbf{V}_{K\downarrow}^*\mathbf{Z}_K^{(V)}$$

where * denotes conjugation, $\mathbf{Z}_K^{(U)} \in \mathbb{C}^{K\times K}$ and $\mathbf{Z}_K^{(V)} \in \mathbb{C}^{K\times K}$ are similarity transforms of $\mathbf{Z}_K$.

For noisy data, the equation of SVD truncation in (2.8) does not hold and there is no exact solution $\mathbf{Z}_K^{(U)}, \mathbf{Z}_K^{(V)}$ of the shift-invariant property in $\mathbf{U}_K, \mathbf{V}_K$ anymore. However, based on the LS estimation of $\overset{\circ}{\mathbf{X}}$ derived in the previous subsection, we can approximate $\mathbf{X}$ by the truncated SVD of

$$\mathbf{X} \overset{\mathrm{SVD}}{=} \mathbf{U}\Sigma\mathbf{V}^H \overset{\mathrm{truncate}}{\approx} \mathbf{U}_K\Sigma_K\mathbf{V}_K^H$$

and compute the TLS solution $\mathbf{Z}_K^{(U)}$ or $\mathbf{Z}_K^{(V)}$ of the set of equations

$$\mathbf{U}_K^{\uparrow} \approx \mathbf{U}_{K\downarrow}\mathbf{Z}_K^{(U)}, \quad or \quad \mathbf{V}_K^{*\uparrow} \approx \mathbf{V}_{K\downarrow}^*\mathbf{Z}_K^{(V)}$$

Then, the estimates of signal-poles $z_k, k = 1, 2, \ldots, K$ can be computed as the eigenvalues of $\mathbf{Z}_K^{(U)}$ or $\mathbf{Z}_K^{(V)}$.

The linear parameters are found by filling in the obtained poles in the model function (1.4) and fitting these equations to the original data with the LS method, as described in [23][27] and shown in the following outline of the HTLS algorithm.

Algorithm 1

Outline of the HTLS algorithm

Given. *A model-order estimate K and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel matrix $\mathbf{X}$ as in (2.1).*

Step 1: SVD. *Compute a truncated SVD of $\mathbf{X} \approx \mathbf{U}_K \mathbf{\Sigma}_K \mathbf{V}_K^H$, where $\mathbf{U}_K$ ($\mathbf{V}_K$) are the first K left (right) singular vectors, and $\mathbf{\Sigma}_K$ is the diagonal matrix composed of the first K singular values.*

Step 2: TLS. *Compute the TLS solution $\mathbf{Z}_K^{(U)}$ of $\mathbf{U}_K^{\uparrow} \approx \mathbf{U}_{K\downarrow} \mathbf{Z}_K^{(U)}$, extract the signal poles $z_k, k = 1, 2, \ldots, K$ from the eigenvalues of $\mathbf{Z}_K^{(U)}$, and find the nonlinear parameters f_k, d_k, $k = 1, \cdots, K$ using the relation $z_k = e^{(-d_k + j2\pi f_k)/f_s}$.*

Step 3: (LS). *Compute the LS solution $\mathbf{c} = [c_1, c_2, ..., c_K]^T$ of the set*

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1^1 & z_2^1 & \cdots & z_K^1 \\ \vdots & & & \vdots \\ z_1^{N-1} & z_2^{N-1} & \cdots & z_K^{N-1} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_K \end{bmatrix} \approx \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix}$$

based on the SVD of the Vandermonde matrix at the left-hand-side, and then estimate the linear parameters a_k and φ_k from $c_k = a_k e^{j\varphi_k}, k = 1, 2, \ldots, K$.

The pole-estimation in step 2 can also be performed in the row space just by solving $\mathbf{V}_K^{*\uparrow} \approx \mathbf{V}_{K\downarrow}^{*} \mathbf{Z}_K^{(V)}$ instead of $\mathbf{U}_K^{\uparrow} \approx \mathbf{U}_{K\downarrow} \mathbf{Z}_K^{(U)}$ and retrieving the eigenvalues of $\mathbf{Z}_K^{(V)}$, instead of $\mathbf{Z}_K^{(U)}$.

It is to be noted that the HTLS algorithm performs well on $\mathbf{X} = \overset{\circ}{\mathbf{X}} + \mathbf{E}$ when $\overset{\circ}{\mathbf{X}}^H \mathbf{E} = \mathbf{0}$ is satisfied asymptotically, and $\mathbf{E}^H\mathbf{E}$ is a scalar multiple of identity matrix, or, the noise is asymptotically white. This is due to the assumptions C1, C2 and (2.4) in the previous subsection.

For differently sized data matrices $\mathbf{X}_{L \times M}$, with $L + M - 1 = N$, the HTLS algorithm gives rather accurate estimates at around $1/3 \leq M/N \leq 2/3$ [26]. But the best estimates are obtained at $M/N \approx 0.5$. Hence, $\mathbf{X}$ is suggested to be as square as possible, i.e., $L = M(+1)$, in the following study.

2.3 Monte-Carlo simulation procedure

In the next chapters, the performance of different estimation algorithms are compared and evaluated via extensive Monte-Carlo experiments on simulated signals. Here we describe the simulation procedure in general.

The objective is to resolve the sinusoids of interest in simulated signals and estimate their signal parameters using a limited number of noisy observations by means of different methods. Various scenarios are simulated in which SNR, number of data points, estimated order, matrix size, and other parameters are varied over a specified range. All computations are performed using Matlab [31].

The simulated signals studied in this thesis are described in Appendix A. Their FFT spectra are depicted in figures. Following NMR spectroscopy convention, these spectra are reversed, i.e., from positive frequency to negative frequency, and expressed mostly using the real part of the FFT, which gives sharper peaks and clearer view of phases. Each of these signals is represented by N data points, uniformly sampled at f_s Hz and modeled by the Kth order exponential function perturbed by white Gaussian noise given by (1.4). The corresponding signal parameters f_k, d_k, a_k and $\phi_k = \varphi_k * 180/\pi$ (i.e., ϕ_k expresses the phase φ_k in degrees) are listed in tables. The real and imaginary components of the Gaussian noise have standard deviation σ_ν.

In our Monte Carlo simulations, we compare the performance of different methods in terms of resolution and parameter accuracy. Unless specified otherwise, 200 noise trials are realized for each considered noise standard deviations σ_ν. For each case, the same (noisy) data are used for all the methods. The results are shown in tables and figures. Typically, when σ_ν are varied, the failure rates are plotted versus σ_ν, while the (relative) root mean-squared error (RMSE) values of the estimated parameters, together with the corresponding Cramér-Rao (CR) lower bounds, are plotted versus the peak SNR. Alternatively, these results, as well as the bias and standard deviation of the estimated parameters, may be listed in tables in some cases. The computation of the above notations, such as peak SNR, CR bound, failure, RMSE, bias and standard deviation, are explained in the following.

Peak SNR:

The SNR of peak k is expressed in decibels and given by $10\log_{10}(a_k^2/(2\sigma_\nu^2))$.

CR bound:

The CR lower bounds [22] on the statistical errors in the parameter estimates are used for accuracy evaluation. The values of CR bounds are derived from σ_ν, the number of data points used and the exact parameter values of the complex exponentially damped model specified in (1.4).

Failure or bad run:

A *failure*, or a bad run, occurs if not all peaks are resolved within specified intervals lying symmetrically around the exact frequencies. The defined intervals for the series of simulated signals used in this thesis are also listed in Appendix A. They are chosen based on the CR bounds at which the frequency intervals of CR bounds for the overlapping peaks touch one another. Take *Signal-1* for example. The half-width of the frequency intervals in which each peak should be resolved are respectively 0.0094 Hz and 0.0106 Hz. In fact, these numbers are the CR frequency bounds for each peak at the noise standard deviation $\sigma_\nu = 0.2$. As such, the sum of the contiguous half intervals 0.0094 Hz+0.0106 Hz= 0.02 Hz is exactly the same as the frequency difference between the two sinusoids. The failure rate is computed to indicate the percentage of failures or bad runs that an algorithm fails to resolve all peaks within these specified intervals.

RMSE, bias and standard deviation:

The values of (relative) RMSE, bias and standard deviation of the signal parameter estimates f_k, d_k, a_k and ϕ_k are computed using the *good* runs only (excluding failures). To be more specific, RMSE values are calculated for f_k and ϕ_k, and relative RMSE values with respect to exact parameter values are calculated for d_k and a_k. Although the exclusion of failures is not statistically sound, the same procedure was applied in [26][27] and is mainly justified from the NMR application as follows. In NMR spectroscopic data measurements, each peak of interest characterizes a certain metabolite and is expected to appear around an approximately known frequency. This information allows us to decide whether a run is successful or not. The corresponding parameter estimates only contain useful metabolite information when a peak estimate is found within the expected frequency interval. Therefore, failures should be excluded in our RMSE computation since it makes no sense to compare the parameters of a peak of interest with those of a noise peak that might appear far away from the expected frequency region. In addition, provided the failure rate of the methods under study are comparable, this procedure allows us to point out which method provides the best parameter estimates in case the peaks can be resolved.

Chapter 3

Prefiltering techniques

A typical SVD and subspace-based estimation method yields automatically all the $4K$ parameters when fitting a noisy data sets composed of K exponentially damped sinusoids modeled by (1.4). It can filter out part of the noise by truncating the SVD of the Hankel data matrix to rank K and discarding the non-significant singular values and vectors, as done by the HTLS Algorithm 1 introduced in the previous chapter. However, such SVD filtering is not always satisfactory especially at low SNR. Moreover, in some applications such as NMR spectroscopy [35][49], it frequently occurs that only some (fewer than K) of the spectral components that reside in a selected frequency region are of interest. To further suppress the noise and/or enhance sinusoids of interest, it is often desired to prefilter the input data. In this chapter, we propose prefiltering methods which can be applied for two purposes:

1. When all the sinusoids are of interest and the SNR's are low, a bandpass filter can be designed to encompass all the sinusoids and to reduce the noise outside the passband.

2. When only some of the sinusoids are of interest with known frequencies but unknown damping factors and amplitudes as typically occurs in NMR spectroscopy, a prefiltering technique can be designed to select only the wanted peaks and suppress the unwanted ones and the noise.

An LP-ZOOM technique has been proposed in [50] based on linear prediction. By selectively fitting the areas containing peaks of interest rather than fitting the whole spectrum, LP-ZOOM reduces the prediction filter length as well as the computation time, and produces more accurate parameter estimates

compared to the ordinary LP method. However, the improvement in accuracy is limited by the linear prediction principle.

Another prefiltering technique has been proposed in [51]. This method applies bandpass finite impulse response (FIR) prefiltering to the data prior to applying the MP estimation algorithm. The essence of this method is to multiply the original Hankel data matrix with a square, full-rank FIR filter matrix. However, the FIR filter matrix used in [51], referred to as $\mathbf{H}_{\mathrm{FIR}c}$ in this chapter, is circularly constructed from the filter impulse response without zero-padding. It implements discrete circular convolution without zero-padding, and therefore introduces an end-effect distortion [52]. Moreover, the restriction in $\mathbf{H}_{\mathrm{FIR}c}$ that the length of an FIR filter be equal to that of the input sometimes imposes a difficulty for the filter design.

In the following sections of this chapter, we present several new prefiltering techniques. First, we illustrate in Section 3.1 why our prefiltering using an FIR or IIR filter is performed via the multiplication of the Hankel data matrix with a filter matrix. Then, in Section 3.2, an improved filter matrix that implements the FIR prefiltering by means of discrete convolution with zero-padding, as well as other filter matrices for the backward and forward infinite impulse response (IIR) prefiltering are constructed. These matrices are applied prior to the HTLS method, referred to as prefiltered HTLS in Section 3.3. Subsequently, a theoretical analysis on a special case of two exponentially damped sinusoids is given in Section 3.4, revealing the relationship between the singular values/vectors of the prefiltered and original data matrices. Furthermore, the HTLS_F prefiltering technique is developed in Section 3.5, where prefiltering is accomplished based on a pre-estimation of the signal parameters using HTLS, instead of an FIR or IIR filter. Finally, comparison and combination of different prefiltering techniques are made and the benefits of these new techniques are confirmed through simulations in Section 3.6.

Note that the proposed prefiltering technique can be used for other subspace and SVD-based methods as well, such as the LP method, the MP method and Kung et al.'s method.

3.1 Hankel matrix and filtering

The SVD filtering, which is crucial to the subspace-oriented estimation algorithms, is based on the rank-K property of Hankel/Toeplitz data matrices in the absence of noise. However, a Hankel/Toeplitz matrix constructed from the filtered data, instead of the original noiseless data, does not possess the rank-K property anymore. Such changes in the rank of Hankel matrices, as is the case

for HTLS, are elucidated below from the signal poles' point of view.

In the absence of noise, the signal modeled by (1.4) as a sum of K exponentials is in fact an autoregressive and moving average (ARMA) signal with a rational z-transform:

$$\mathrm{X}(\mathrm{z}) = \sum_{k=1}^{K} \frac{c_k}{1 - z_k \mathrm{z}^{-1}} = \frac{\sum_{i=0}^{K-1} b_i \mathrm{z}^{-i}}{1 + \sum_{k=1}^{K} a_k \mathrm{z}^{-k}}$$

Consider a general ARMA model of

$$\breve{\mathrm{X}}(\mathrm{z}) = \frac{\sum_{i=0}^{r_b} b_i \mathrm{z}^{-i}}{\sum_{k=0}^{r_a} a_k \mathrm{z}^{-k}}, \quad a_0 \neq 0$$

In the time domain, its impulse response $\breve{x}_n$ can be calculated from

$$\breve{x}_n = \frac{1}{a_0}\Big(\sum_{i=0}^{r_b} b_i \delta_{n-i} - \sum_{k=1}^{r_a} a_k \breve{x}_{n-k}\Big)$$

using the initial conditions $\breve{x}_{-1} = \breve{x}_{-2} = \cdots = \breve{x}_{-r_a} = 0$. The forward recursion of this model can be written in matrix notation

$$\begin{bmatrix} & & & & \breve{x}_0 \\ & 0 & & \breve{x}_0 & \breve{x}_1 \\ & & \cdot & \breve{x}_1 & \breve{x}_2 \\ & \cdot & & & \vdots \\ \breve{x}_0 & \breve{x}_1 & \cdot & \cdot & \breve{x}_{r_a} \\ \breve{x}_1 & \breve{x}_2 & \cdot & \cdot & \breve{x}_{r_a+1} \\ \vdots & \vdots & & & \vdots \\ \vdots & \vdots & & & \vdots \end{bmatrix} \begin{bmatrix} a_{r_a} \\ a_{r_a-1} \\ \vdots \\ \vdots \\ a_0 \end{bmatrix} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{r_b} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Note that the matrix in the box is a Hankel matrix with $r_a + 1$ columns. As long as its number of rows is larger than r_a, the following statements are straight-forwardly proven from the rank deficiency of the Hankel matrix.

- If $r_a > r_b$, the Hankel matrix in the box, or starting from $\breve{x}_0$, has rank r_a, which is the number of poles.
- If $r_a \leq r_b$, the Hankel matrix starting from $\breve{x}_{r_b-r_a+1}$ has rank r_a.

Extending the Hankel matrix in the box with extra columns on the right hand does not increase its rank since these extra columns of the extended Hankel

matrix can always be found in a Hankel matrix in the box with more rows at the bottom. Hence the above statements also hold for a Hankel matrix with a bigger size.

For a signal modeled as a sum of K exponentially damped sinusoids as in the model (1.4) of x_n, we have $r_a = K, r_b = K-1$. Hence the rank-K property of the Hankel matrix $\mathbf{X}$ in (2.1) is proven again here from the above conclusion.

Once these data are fed to a filter with a z-transfer function $\mathsf{H}(\mathsf{z})$, the z-transform of the filtered data is $\tilde{\mathsf{X}}(\mathsf{z}) = \mathsf{X}(\mathsf{z})\mathsf{H}(\mathsf{z})$. A tilde, $\tilde{}$, is used above the variables to represent the filtered counterpart in this chapter. Denote the r_a, r_b parameters of $\tilde{\mathsf{X}}(\mathsf{z})$ by $r_{a(\tilde{x}_n)}, r_{b(\tilde{x}_n)}$ and we can see from the following that the new Hankel matrix constructed from the filtered data $\tilde{x}_n$ has not necessarily rank K.

Suppose an IIR filter with $r_a = r_{a(IIR)}$ and $r_b = r_{b(IIR)}$ is applied. Then $r_{a(\tilde{x}_n)} = K + r_{a(IIR)}$ and $r_{b(\tilde{x}_n)} = K - 1 + r_{b(IIR)}$ when no pole-zero cancellation occurs. Generally, $r_{a(IIR)} \geq r_{b(IIR)}$, thus $r_{a(\tilde{x}_n)} > r_{b(\tilde{x}_n)}$ and the new Hankel matrix starting from $\tilde{x}_0$ has rank $K + r_{a(IIR)}$ as a result of the first statement above. It is well known that the increase in the rank of the Hankel matrix and hence the number of unknowns results in larger variances of the estimated parameters. This is certainly to be avoided in accurate parameter estimation.

Suppose an FIR filter with $r_a = 0$ and $r_b = r_{b(FIR)}$ is applied. Then $r_{a(\tilde{x}_n)} = K$ and $r_{b(\tilde{x}_n)} = K - 1 + r_{b(FIR)}$ in general. According to the second statement above, the new Hankel matrix starting from $\tilde{x}_{r_{b(FIR)}}$ has rank K, while the Hankel matrix starting from $\tilde{x}_0$ has generally rank $K+1$. Therefore, the first $r_{b(FIR)}$ filtered data should be discarded to preserve the rank-K property. This unfortunately leads to a loss of accuracy of the estimated parameters. Intuitively this is to be expected because a lot of useful information is lost by deleting the initial part of the filtered signal which includes the portion of the decaying signal with the highest SNR.

To solve these problems of FIR and IIR prefiltering, we propose FIR and IIR filter matrices in the next section. The FIR or IIR prefiltering is then implemented via multiplication of the original Hankel data matrix $\mathbf{X}$ with the FIR or IIR filter matrix. The prefiltered matrix, or the product obtained thereof, keeps the rank-K property and can readily be processed by an SVD-based estimation algorithm.

3.2 FIR and IIR filter matrices

Most time-domain filtering processes are linear, and can thus be implemented by means of a matrix multiplication. Clearly stated, we should find a filter matrix $\mathbf{H} \in \mathbb{C}^{p\times p}$ such that $\tilde{\mathbf{x}} = \mathbf{H}\mathbf{x} \in \mathbb{C}^{p\times 1}$ corresponds to the filtered data vector, when $\mathbf{x} \in \mathbb{C}^{p\times 1}$ is the input data vector. If the input and output are row vectors instead of column vectors, the filter matrix is simply transposed and multiplied from the right as in $\tilde{\mathbf{x}}^T = \mathbf{x}^T\mathbf{H}^T$. The following two filter matrices, $\mathbf{H}_{\mathrm{FIR}t}$ and $\mathbf{H}_{\mathrm{IIR}}$, are constructed with $\mathbf{H}_{\mathrm{FIR}t}$ implementing the FIR filtering without end effect and $\mathbf{H}_{\mathrm{IIR}}$ implementing the IIR filtering. For comparison, the circular FIR filter matrix $\mathbf{H}_{\mathrm{FIR}c}$ used in [51] is reviewed as well.

3.2.1 FIR filter matrix: $\mathbf{H}_{\mathrm{FIR}}$

Suppose q is the length of a digital filter $\hat{h}_n$ and w_n is the window function, then we have the impulse response of the FIR filter $h_n = \hat{h}_n w_n$, $n = 1, 2, \cdots, q$. The output data from the FIR filter can be found using the discrete convolution or discrete circular convolution (denoted by $\otimes$ in this thesis) of input data $x_1, x_2, \ldots, x_p$ and the filter impulse response $h_1, h_2, \ldots, h_q$.

Because the discrete circular convolution assumes that both the sampled data $x_1, x_2, \ldots, x_p$ and $h_1, h_2, \ldots, h_q$ be one period of their respective infinite periodic functions, the imposed periodicity introduces what is known as an end effect [52]. This is a special phenomenon that arises when using a discrete circular convolution to approximate the continuous convolution. It can be avoided by appending sufficient zeros to both $x_1, x_2, \ldots, x_p$ and $h_1, h_2, \ldots, h_q$ to be convolved.

The following Toeplitz filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ below achieves such discrete circular convolution with zero padding.

$$\mathbf{H}_{\mathrm{FIR}t} \triangleq \begin{bmatrix} h_\tau & \cdots & h_1 & 0 & \cdots & \cdots & 0 \\ \vdots & & h_2 & \ddots & \ddots & & \vdots \\ h_q & & \vdots & & h_1 & \ddots & \vdots \\ 0 & \ddots & \vdots & & h_2 & \ddots & 0 \\ \vdots & \ddots & h_q & & \vdots & & h_1 \\ \vdots & & \ddots & \ddots & \vdots & & \vdots \\ 0 & \cdots & \cdots & 0 & h_q & \cdots & h_\tau \end{bmatrix} \in \mathbb{C}^{p\times p},$$

$$\text{where } \tau = \lfloor \frac{q+1}{2} \rfloor \tag{3.1}$$

The verification of FIR filtering using $\mathbf{H}_{\mathrm{FIR}t}$ can be done by writing out the result of discrete circular convolution with zero padding, that is,

$$
\begin{aligned}
&[\ \tilde{x}_1 \quad \tilde{x}_2 \ \cdots \ \tilde{x}_{p+q-1}]^T \\
&\quad = [x_1 \ x_2 \ \cdots \ x_p \ \underbrace{0 \ 0 \ \cdots \ 0}_{q-1}]^T \otimes [h_1 \ h_2 \ \cdots \ h_q \ \underbrace{0 \ 0 \ \cdots \ 0}_{p-1}]^T
\end{aligned}
$$

$$
= \begin{cases}
\begin{bmatrix}
h_1x_1 & & & & \\
h_1x_2+ & h_2x_1 & & & \\
\vdots & & & \ddots & \\
h_1x_q+ & h_2x_{q-1}+ & \cdots+\cdots & & h_qx_1 \\
h_1x_{q+1}+ & h_2x_q+ & \cdots+\cdots & & h_qx_2 \\
\vdots & & & & \vdots \\
h_1x_p+ & h_2x_{p-1}+ & \cdots+\cdots & & h_qx_{p+1-q} \\
 & h_2x_p+ & h_3x_{p-1}+ & \cdots+\cdots & h_qx_{p+2-q} \\
 & & \ddots & & \vdots \\
 & & & h_{q-1}x_p+ & h_qx_{p-1} \\
 & & & & h_qx_p
\end{bmatrix} & \text{if } q \le p \\
\begin{bmatrix}
h_1x_1 & & & & \\
h_1x_2+ & h_2x_1 & & & \\
\vdots & & & \ddots & \\
h_1x_p+ & h_2x_{p-1}+ & \cdots+ & & h_px_1 \\
h_2x_p+ & h_3x_{p-1}+ & \cdots+ & & h_{p+1}x_1 \\
\vdots & & & & \vdots \\
h_{q-p+1}x_p+ & h_{q-p}x_{p-1}+ & \cdots+ & & h_qx_1 \\
h_{q-p+2}x_p+ & h_{q-p+1}x_{p-1}+ & \cdots+ & h_qx_2 & \\
\vdots & & & & \\
h_{q-1}x_p+ & h_qx_{p-1} & & & \\
h_qx_p & & & &
\end{bmatrix} & \text{if } q > p
\end{cases}
$$

$$
= \begin{bmatrix}
h_1 & 0 & & \cdots & \cdots & \cdots & 0 \\
\vdots & \ddots & \ddots & & & & \vdots \\
h_\tau & \cdots & h_1 & 0 & & & 0 \\
\vdots & & & \ddots & \ddots & & \vdots \\
h_q & h_{q-1} & \cdots & & h_1 & 0 & 0 \\
0 & h_q & h_{q-1} & \cdots & & h_1 & \vdots \\
\vdots & \ddots & \ddots & \ddots & \cdots & & 0 \\
0 & & 0 & h_q & h_{q-1} & \cdots & h_1 \\
\vdots & & & \ddots & \ddots & & \vdots \\
0 & & & & 0 & h_q & h_\tau \\
\vdots & & & & & \ddots & \vdots \\
0 & & \cdots & \cdots & \cdots & 0 & h_q
\end{bmatrix}
\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}
$$

$$
= \mathbf{H}_{\mathrm{FIR0}}\mathbf{x} \qquad \text{(say)} \tag{3.2}
$$

Here, $\mathbf{H}_{\mathrm{FIR0}}$ is an FIR filter matrix without end effect. But it gives distortion at the tails of the filtered output due to the discontinuity caused by the imposed initial conditions. If we eliminate the filtered result in $\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{p+q-1}$ that correspond to zeros appended to x_n (or non-signal part) and adjust the time shift caused by $h_1, h_2, \cdots, h_q$ (the indices of $h_1, h_2, \cdots, h_q$ have a time delay of $\lfloor \frac{q-1}{2} \rfloor$, since the modulus of the filter impulse response is often symmetrical about time 0 for a filter with symmetric passband about its filter center), we end up with a truncated square FIR filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ of (3.1) in the outer box, which is obtained by deleting the $\tau - 1$ top rows and the $q - \tau$ bottom rows of the above filter matrix. A prefiltering using $\mathbf{H}_{\mathrm{FIR}t}$ removes most part of the distorted output and keeps those filtered data with highest SNR. Of course further truncation, as shown in the inner box, eliminates the distortion completely, but the number of output data available is very limited and the SNR is seriously decreased. Computer simulations indicate that the truncated square filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ gives best results among the several possible filter matrices discussed above. Therefore, we will only use $\mathbf{H}_{\mathrm{FIR}t}$ as FIR filter matrix in this chapter and compare $\mathbf{H}_{\mathrm{FIR}t}$ with $\mathbf{H}_{\mathrm{FIR}c}$ given below.

If we write out the discrete circular convolution result without zero padding, then for $p = q$, a circular filter matrix $\mathbf{H}_{\mathrm{FIR}c}$ used in [51] is obtained:

$$
\begin{aligned}
&[\tilde{x}_1 \ \tilde{x}_2 \quad \cdots \quad \tilde{x}_p]^T \\
&\qquad = [x_1 \ x_2 \ \cdots \ x_p]^T \otimes [h_1 \ h_2 \ \cdots \ h_p]^T
\end{aligned}
$$

$$= \begin{bmatrix} h_p & h_{p-1} & h_{p-2} & \cdots & h_1 \\ h_1 & h_p & h_{p-1} & \cdots & h_2 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ h_{p-2} & & \ddots & \ddots & h_{p-1} \\ h_{p-1} & h_{p-2} & \cdots & h_1 & h_p \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}$$

$$= \mathbf{H}_{\mathrm{FIR}c} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{bmatrix}$$

The advantages of $\mathbf{H}_{\mathrm{FIR}t}$ compared to the circular filter matrix $\mathbf{H}_{\mathrm{FIR}c}$ proposed in [51] lie in the zero-padding, which eliminates the end effect and allows an arbitrary choice of the filter length q as well. The end effect is caused by the summation of the discrete circular convolution result at both ends, if nonzero, of the cycle. In our case, the exponentially damped sinusoidal signal and the filter impulse response of a bandpass filter are both infinitely long in the time domain, that is, they never die out to exactly zero if a limited number of data samples is used. Therefore, zero-padding is always necessary for the discrete circular convolution here in order to avoid end effects. The benefits of $\mathbf{H}_{\mathrm{FIR}t}$ are more pronounced when the signal has small damping factors and/or the required FIR filter has a narrow passband in the frequency domain and thus a wide impulse response in the time domain.

Fig. 3.1 compares the results of filtering a single exponential using $\mathbf{H}_{\mathrm{FIR}c}$ and $\mathbf{H}_{\mathrm{FIR}t}$. It can be seen that there is an end-effect distortion at the tail when $\mathbf{H}_{\mathrm{FIR}c}$ is used, while the end effect is avoided when $\mathbf{H}_{\mathrm{FIR}t}$ is used.

3.2.2 IIR filter matrix: $\mathbf{H}_{\mathrm{IIR}}$

In applications in which we do not need to process the signals in real time, e.g. in NMR data quantification, we can consider both the backward and forward recursion.

Given the *backward recursion* of an rth-order IIR filter

$$\begin{aligned} \tilde{X}(z) &= \frac{b_0 z^r + b_1 z^{r-1} + \cdots + b_{r-1} z + b_r}{a_0 z^r + a_1 z^{r-1} + \cdots + a_{r-1} z + a_r} X(z) \\ \text{or} \quad a_r \tilde{x}_n &= b_r x_n + b_{r-1} x_{n+1} + \cdots + b_0 x_{n+r} \\ &\quad -a_{r-1} \tilde{x}_{n+1} - \cdots - a_1 \tilde{x}_{n+r-1} - a_0 \tilde{x}_{n+r}, \\ & \qquad\qquad\qquad\qquad a_0 \neq 0, a_r \neq 0 \end{aligned}$$

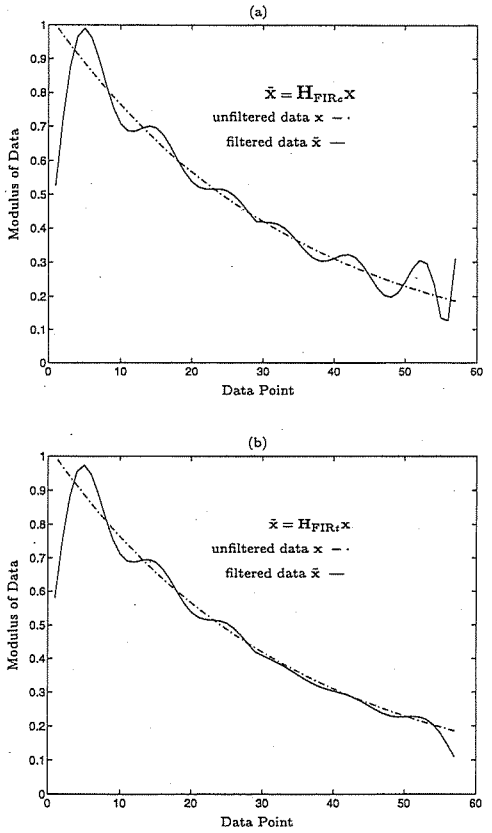


Figure 3.1: An example of FIR filtering using the filter matrix (a) $\mathbf{H}_{\mathrm{FIR}c}$ and (b) $\mathbf{H}_{\mathrm{FIR}t}$. The unfiltered data x is the first sinusoid of *Signal-4*, i.e., $e^{-0.02n\Delta t - j2\pi 0.2n\Delta t}$. The filter is the bandpass rectangular filter as used in Example 2 in the simulation section of this chapter.

with the initial conditions:

$$\begin{aligned} x_{p+1} &= x_{p+2} = \cdots = x_{p+r} = 0, \\ \tilde{x}_{p+1} &= \tilde{x}_{p+2} = \cdots = \tilde{x}_{p+r} = 0, \end{aligned}$$

we can set up a full-rank upper-triangular filter matrix for the backward recursion

$$\mathbf{H}_{\text{IIR}} = \begin{bmatrix} g_1 & g_2 & \cdots & g_p \\ 0 & g_1 & \cdots & g_{p-1} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_1 \end{bmatrix} \in \mathbb{C}^{p \times p} \tag{3.3}$$

with the elements

$$\begin{aligned} g_1 &= b_r/a_r, \\ g_i &= \begin{cases} (b_{r-i+1} - a_{r-1}g_{i-1} - a_{r-2}g_{i-2} - \cdots - a_{r-i+1}g_1)/a_r, \\ \qquad\qquad \text{for } 2 \le i \le r+1 \\ (-a_{r-1}g_{i-1} - a_{r-2}g_{i-2} - \cdots - a_0 g_{i-r})/a_r, \\ \qquad\qquad \text{for } r+2 \le i \le p \end{cases} \end{aligned}$$

This is verified by comparing the result of $\tilde{\mathbf{x}} = \mathbf{H}_{\text{IIR}}\mathbf{x}$ with that of the backward recursion.

The *forward recursion* of an IIR filter starts with the first samples of the signal instead of the last ones, as shown below.

Given the forward recursion of an rth-order IIR filter

$$\begin{aligned} \tilde{X}(z) &= \frac{b_0 + b_1 z^{-1} + \cdots + b_{r-1} z^{-(r-1)} + b_r z^{-r}}{a_0 + a_1 z^{-1} + \cdots + a_{r-1} z^{-(r-1)} + a_r z^{-r}} X(z) \\ \text{or } a_0 \tilde{x}_n &= b_0 x_n + b_1 x_{n-1} + \cdots + b_r x_{n-r} \\ &\quad -a_1 \tilde{x}_{n-1} - \cdots - a_{r-1}\tilde{x}_{n-(r-1)} - a_r \tilde{x}_{n-r}, \\ &\qquad\qquad a_0 \neq 0, a_r \neq 0 \end{aligned}$$

with the initial conditions:

$$\begin{aligned} x_{-r+1} &= x_{-r+2} = \cdots = x_{-1} = x_0 = 0, \\ \tilde{x}_{-r+1} &= \tilde{x}_{-r+2} = \cdots = \tilde{x}_{-1} = \tilde{x}_0 = 0, \end{aligned}$$

we can set up a full-rank lower-triangular filter matrix for the forward recursion

$$\mathbf{H}_{\text{IIR}} = \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ g_p & g_{p-1} & \cdots & g_1 \end{bmatrix} \in \mathbb{C}^{p \times p}$$

with the elements

$$\begin{aligned} g_1 &= b_0/a_0, \\ g_i &= \begin{cases} (b_{i-1} - a_1 g_{i-1} - a_2 g_{i-2} - \cdots - a_{i-1} g_1)/a_0, \\ \qquad\qquad \text{for } 2 \le i \le r+1 \\ (-a_1 g_{i-1} - a_2 g_{i-2} - \cdots - a_r g_{i-r})/a_0, \\ \qquad\qquad \text{for } r+2 \le i \le p \end{cases} \end{aligned}$$

Actually, the filter matrix for forward IIR can be obtained from that of backward IIR, that is, by first reversing the label sequence of a and b respectively, then calculating g_i in the same way, and finally forming a lower-triangular matrix or the transposition of an upper-triangular matrix.

For damped signals, a backward recursion, rather than a forward recursion, is recommended because the initial conditions can be easily fulfilled. That is, the signal should die out to zero after the pth data sample when using a backward recursion.

The phase distortion of an IIR or FIR (for non-linear phase) filter does not affect the parameter estimates. This is because the phases are calculated in a later step by fitting the model function using the estimated poles to the *original data*, not to the filtered data.

3.3 Prefiltered HTLS

By prefiltered HTLS we mean that a full-rank filter matrix $\mathbf{H}$ is multiplied with the Hankel data matrix $\mathbf{X}$ before applying the estimation algorithm HTLS. As formulated below, a right (resp., left) multiplication with a full-rank matrix $\mathbf{H}_r$ (resp., $\mathbf{H}_l$) of appropriate dimensions retains the rank-K property of the prefiltered data matrix and shift-invariant properties of $\tilde{\mathbf{U}}_K$ (resp., $\tilde{\mathbf{V}}_K^*$) in the absence of noise. These properties are required for a correct use of the HTLS method as shown below:

$$\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r = \tilde{\mathbf{U}}_K \tilde{\mathbf{\Sigma}}_K \tilde{\mathbf{V}}_K^H = \sum_{k=1}^{K} \tilde{\sigma}_k \tilde{\mathbf{u}}_k \tilde{\mathbf{v}}_k^H \;\Rightarrow\; \tilde{\mathbf{U}}_K^{\uparrow} = \tilde{\mathbf{U}}_{K\downarrow} \tilde{\mathbf{Z}}_K^{(U)}$$

$$\text{resp.,} \quad \tilde{\mathbf{X}} = \mathbf{H}_l\mathbf{X} = \tilde{\mathbf{U}}_K\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{V}}_K^H = \sum_{k=1}^{K}\tilde{\sigma}_k\tilde{\mathbf{u}}_k\tilde{\mathbf{v}}_k^H \;\Rightarrow\; \tilde{\mathbf{V}}_K^{*\uparrow} = \tilde{\mathbf{V}}_{K\downarrow}^{*}\tilde{\mathbf{Z}}_K^{(V)}$$

$\mathbf{H}_r$ and $\mathbf{H}_l$ are the filter matrices used in the case of right and left multiplication, respectively. Since a right (resp., left) multiplication with a full-rank square matrix does not change the column (resp., row) space of $\mathbf{X}$, there exists a unitary matrix $\mathbf{\Omega}$ such that $\tilde{\mathbf{U}} = \mathbf{U}\mathbf{\Omega}$ (resp., $\tilde{\mathbf{V}} = \mathbf{V}\mathbf{\Omega}$). Furthermore, it can be proven that $\mathbf{\Omega} = \begin{bmatrix} \mathbf{\Omega}_1 & 0 \\ 0 & \mathbf{\Omega}_2 \end{bmatrix}$ with $\mathbf{\Omega}_1 \in \mathbb{C}^{K\times K}$ (see Theorem 3-i at the end of the next section).

Algorithm 2

Outline of the prefiltered HTLS algorithm

Given. A model-order estimate K of all peaks, an FIR or IIR bandpass filter encompassing the frequency range of interest, and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel data matrix $\mathbf{X}$ as in (2.1).

Step 1. Construct an FIR filter matrix $\mathbf{H} = \mathbf{H}_{\mathrm{FIR}t}$ as in (3.1) or an IIR filter matrix $\mathbf{H} = \mathbf{H}_{\mathrm{IIR}}$ as in (3.3), where $\mathbf{H} \in \mathbb{C}^{p\times p}$. For the prefiltering via a right multiplication with the filter matrix, $p = M$ and $\mathbf{H}_r = \mathbf{H}^T$ are taken.

Step 2. Prefilter $\mathbf{X}$ by matrix multiplication:

$$\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$$

Step 3. Apply the pole-estimation part (Step 1-2) of the HTLS Algorithm 1 to $\tilde{\mathbf{X}}$ to find f_k and d_k, $k = 1, \cdots, K$.

Step 4. Apply the linear-parameter-estimation part (Step 3) of the HTLS Algorithm 1, to the original data x_n, $n = 0, 1, \cdots, N-1$, to find a_k and φ_k, $k = 1, \cdots, K$.

For prefiltering via a left multiplication of the filter matrix, simply replace $p = M$ and $\mathbf{H}_r = \mathbf{H}^T$ by $p = L$ and $\mathbf{H}_l = \mathbf{H}$ in step 1, and then perform $\tilde{\mathbf{X}} = \mathbf{H}_l\mathbf{X}$ instead of $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$ in step 2. In this case, $\mathbf{V}_K$, instead of $\mathbf{U}_K$, is used for computing f_k, d_k in the HTLS Algorithm 1.

The remainder of this chapter considers only the case of *right* multiplication, i.e., the prefiltering is performed as $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$.

Observe that $\tilde{\mathbf{X}}$ is no longer a Hankel matrix. Each row of $\tilde{\mathbf{X}}$ is the filtered output of the corresponding row of $\mathbf{X}$. Since $\mathbf{X}$ is a Hankel matrix composed of L rows of shifted data segments, such row-by-row filtering can be visualized as segment-by-segment filtering. The rectangular truncation of each segment introduces the Gibbs phenomenon, which can be reduced by weighting each segment with an appropriate *window*. This can be realized via matrix multiplication $\mathbf{X}\mathbf{W}$, where $\mathbf{W}$ is a full-rank diagonal weighting matrix. In this way, the prefiltering proceeds as $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{W}\mathbf{H}_r$. Since $\mathbf{H}_r$ and $\mathbf{W}$, if any, are square and of full rank, the size and the effective rank of $\tilde{\mathbf{X}}$ are the same as those of $\mathbf{X}$. Therefore the computational work involved in the SVD of $[\tilde{\mathbf{U}}_{K\downarrow} \quad \tilde{\mathbf{U}}_K^{\uparrow}]$ for computing the TLS solution $\tilde{\mathbf{Z}}_K^{(U)}$ in the succeeding HTLS estimation is basically unchanged.

3.4 SVD relationships between the prefiltered and original data matrices

In the absence of noise, i.e., $\mathbf{X} = \overset{\circ}{\mathbf{X}}, \mathbf{E} = \mathbf{0}$, the singular values and vectors of the Hankel data matrix $\mathbf{X}$ always depend on the model parameters. Such dependency has been revealed in [53] for a simple signal with only two sinusoids. After prefiltering, the SVD of the prefiltered matrix $\tilde{\mathbf{X}}$ depends not only on the model parameters, but also on the filter matrix. The relationships between the SVD of $\tilde{\mathbf{X}}$, the SVD of $\mathbf{X}$ and the model parameters are investigated in this section in order to justify the prefiltering method and give some insight in the effects of prefiltering.

3.4.1 Definitions and implications

First of all, we give some useful definitions and their implications.

Definition 1 *$\mathbf{s}_k$ and $\mathbf{t}_k$ are the L-tuple and M-tuple vectors of the kth exponentially damped sinusoid in model (1.4), i.e.,*

$$\mathbf{s}_k \triangleq [1, z_k, z_k^2, \cdots, z_k^{L-1}]^T, \quad \mathbf{t}_k \triangleq [1, z_k, z_k^2, \cdots, z_k^{M-1}]^T$$

Definition 2 *Some factors are defined as $\alpha_{ki} \triangleq \mathbf{s}_k^H\mathbf{s}_i$, $\beta_{ki} \triangleq \mathbf{t}_k^T\mathbf{t}_i^*$, $\tilde{\alpha}_{ki} \triangleq \mathbf{s}_k^H\mathbf{H}_l^H\mathbf{H}_l\mathbf{s}_i$, $\tilde{\beta}_{ki} \triangleq \mathbf{t}_k^T\mathbf{H}_r\mathbf{H}_r^H\mathbf{t}_i^*$, $\eta_{ki} \triangleq \mathbf{v}_k^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_i$.*

Obviously, the factors α_{ki} and β_{ki} are the correlation of the kth and ith exponentially damped sinusoids. Since they are very similar ($\alpha_{ki} = \beta_{ki}$ for $L = M$), we only comment on one of them, say, β_{ki}, as the prefiltering via a right multiplication of $\mathbf{H}_r$ is considered. In general, $|\beta_{ki}|/\sqrt{\beta_{kk}\beta_{ii}} \leq 1$ according to Cauchy's inequality. With the concept of a normalized correlation $|\beta_{ki}|/\sqrt{\beta_{kk}\beta_{ii}}$, two extreme cases of signals can be described using the time-domain data model (1.4).

- When the two peaks *coincide*, i.e., $z_k = z_i$, then $|\beta_{ki}|/\sqrt{\beta_{kk}\beta_{ii}}$ reaches its maximum, 1, since $\beta_{ki} = \beta_{kk} = \beta_{ii}$ in this case.
- When the two peaks are *ideally nonoverlapping*, i.e., $d_k = d_i = 0$ and $(f_i - f_k)/f_s = \pm 0.5$, then $|\beta_{ki}|/\sqrt{\beta_{kk}\beta_{ii}}$ reaches its minimum, 0 if M is even, or $1/M$ if M is odd. This is because $\beta_{kk} = \beta_{ii} = \|\mathbf{t}_k\|^2 = M$, and

$$\begin{aligned}\beta_{ki} &= \sum_{n=0}^{M-1}(z_k z_i^*)^n = \frac{1-(z_k z_i^*)^M}{1-(z_k z_i^*)} \\ &= \frac{1-\exp[-M(d_k+d_i)/f_s + j2M\pi(f_k-f_i)/f_s]}{1-\exp[-(d_k+d_i)/f_s + j2\pi(f_k-f_i)/f_s]} \\ &= \begin{cases} 0 & (M \text{ even}) \\ 1 & (M \text{ odd}) \end{cases}\end{aligned}$$

The degree of overlap can also be analyzed in terms of the discrete Fourier transform (DFT) according to the following Theorem. Note that the element index of a vector is denoted in parenthesis in this chapter, since the vector subscript has been reserved for distinguishing different sinusoids.

Theorem 1 *Let $\mathcal{T}_k(m), m = 1, 2, \ldots, M$ be the M-point DFT of $\mathbf{t}_k(n)$, $n = 1, 2, \ldots, M$:*

$$\mathcal{T}_k(m) = \sum_{n=0}^{M-1} \mathbf{t}_k(n) e^{-j2\pi mn/M}$$

Then,

$$\beta_{ki} = \sum_{n=1}^{M} \mathbf{t}_k(n)\mathbf{t}_i^*(n) = \frac{1}{M}\sum_{m=1}^{M} \mathcal{T}_k(m)\mathcal{T}_i^*(m)$$

which implies that β_{ki} can be expressed as the correlation of $\mathbf{t}_i(n)$ and $\mathbf{t}_k(n)$, or the correlation of $\mathcal{T}_i(m)$ and $\mathcal{T}_k(m)$ scaled by a factor $1/M$.

Proof. Let $\mathbf{y}(l)$ be the correlation sum of $\mathbf{t}_k$ and $\mathbf{t}_i$:

$$\mathbf{y}(l) = \sum_{n=1}^{M} \mathbf{t}_k^*(n)\mathbf{t}_i(l+n)$$

Then $\mathbf{y}(0) = \mathbf{t}_k^H \mathbf{t}_i$. According to correlation theorem [52], the DFT of $\mathbf{y}(l)$ is $\mathcal{Y}(m) = \mathcal{T}_k^*(m)\mathcal{T}_i(m)$. Using the DFT property of $\mathbf{y}(0) = \frac{1}{M}\sum_{m=1}^{M}\mathcal{Y}(m)$, we have a form of Parseval's theorem

$$\mathbf{t}_k^H \mathbf{t}_i = \frac{1}{M}\sum_{m=1}^{M} \mathcal{T}_k^*(m)\mathcal{T}_i(m)$$

Hence, the theorem is proven by conjugation. □

Considering that $\mathcal{T}_k(m)$ has the form $a_k/[d_k + j2\pi(m/M - f_k/f_s)f_s]$ for exponentials, the smaller d_k, the narrower the peak k is in the spectrum. When the two peaks, k and i, are *approximately nonoverlapping*, that is, they are sufficiently far apart and have sufficiently small damping factors, their correlation is small as can be deduced intuitively from the frequency domain.

$\tilde{\alpha}_{ki}$ and $\tilde{\beta}_{ki}$ are simply the filtered counterpart of α_{ki} and β_{ki}. Similarly, $|\tilde{\alpha}_{ki}|/\sqrt{\tilde{\alpha}_{kk}\tilde{\alpha}_{ii}} \leq 1, |\tilde{\beta}_{ki}|/\sqrt{\tilde{\beta}_{kk}\tilde{\beta}_{ii}} \leq 1$, and $\tilde{\alpha}_{ki} = \tilde{\beta}_{ki}$ for $L = M$. The relationship between $\tilde{\alpha}_{ki}$ and α_{ki}, $\tilde{\beta}_{ki}$ and β_{ki} depends mainly on the filter matrix. Here we are going to *assume an ideal rectangular bandpass filter with unit gain at passband*, which means that the frequency response of the bandpass filter is 1 inside the passband and 0 outside the passband and the lower and upper cut-off frequencies can be denoted by f_l and f_u. This is a useful simplification in that it delivers neat expressions which provide insights in the prefiltering method [1]. As indicated in the following theorem, the 2-norm of such a rectangular filter matrix is 1, i.e., the first singular value of $\mathbf{H}_r$ is 1. Hence, $\tilde{\beta}_{kk} = \mathbf{t}_k^T \mathbf{H}_r \mathbf{H}_r^H \mathbf{t}_k^* \leq \|\mathbf{H}_r\|_2 \mathbf{t}_k^T \mathbf{t}_k^* = \beta_{kk}$.

Theorem 2 *Suppose $\mathbf{H}$ is a $q \times q$ filter matrix of a rectangular digital bandpass filter, i.e., when $\mathbf{x} \in \mathbb{C}^{q\times 1}$ is the input data vector, $\mathbf{Hx} \in \mathbb{C}^{q\times 1}$ is the filtered output vector, and the r-point DFT of this filter can be written as*

$$|\mathcal{H}(m)| = \begin{cases} 1 & m = m_1, \cdots, m_2 \quad \textit{(inside passband)} \\ 0 & m = 1, \cdots, m_1 - 1, m_2 + 1, \cdots, r \quad \textit{(outside passband)} \end{cases}$$

[1] To realize a rectangular bandpass filter, a filter matrix of infinite dimension should be used. This is impossible in practice but is used here to simplify the theoretical analysis. That is why the filter is called ideal.

where m_1 and m_2 are lower and upper cut-off frequencies of the digital filter and $r \geq q$. Then $\|\mathbf{H}\|_2 = 1$, or, the first singular value of $\mathbf{H}$ is $\sigma_{H1} = 1$.

Proof. Let $\mathbf{x} \in \mathbb{C}^{q\times 1}$, $\tilde{\mathbf{x}} = \mathbf{H}\mathbf{x} \in \mathbb{C}^{q\times 1}$, $\|\mathbf{x}\|_2^2 = \frac{1}{r}\sum_{m=1}^{r} |\mathcal{X}(m)|^2 = 1$ (normalized). Then $\tilde{\mathcal{X}}(m) = \mathcal{X}(m)\mathcal{H}(m)$, where $\mathcal{X}, \tilde{\mathcal{X}}$ and $\mathcal{H}$ are the r-point DFT of $\mathbf{x}, \tilde{\mathbf{x}}$ and the filter impulse response, respectively. Using DFT properties, it follows that

$$\begin{aligned}
\sigma_{H1}^2 &= \|\mathbf{H}\|_2^2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{H}\mathbf{x}\|_2^2 = \max_{\|\mathbf{x}\|_2=1} \|\tilde{\mathbf{x}}\|_2^2 \\
&= \max_{\|\mathbf{x}\|_2=1} \frac{1}{r}\sum_{m=1}^{r} |\tilde{\mathcal{X}}(m)|^2 \quad \text{(Parseval's theorem)} \\
&= \max_{\|\mathbf{x}\|_2=1} \frac{1}{r}\sum_{m=1}^{r} |\mathcal{X}(m)\mathcal{H}(m)|^2 \quad \text{(convolution theorem)} \\
&= \max_{\|\mathbf{x}\|_2=1} \frac{1}{r}\sum_{m=m_1}^{m_2} |\mathcal{X}(m)|^2 \\
&\leq \frac{1}{r}\sum_{m=1}^{r} |\mathcal{X}(m)|^2 \\
&= \|\mathbf{x}\|_2^2 \quad \text{(Parseval's theorem)} \\
&= 1
\end{aligned}$$

The equality holds when the nonzero spectrum of signal $\mathbf{x}$ is completely inside the passband of the filter, i.e., $\mathcal{X}$ is zero outside the passband. On the other hand, $\tilde{\mathbf{x}} = \mathbf{H}\mathbf{x} = 0$ when $\mathcal{X}$ is zero inside the passband.

□

3.4.2 SVD expressions of X and $\tilde{\mathbf{X}}$ in terms of signal parameters

The above definitions are useful in that they simplify the expressions of $\mathbf{X}$, $\tilde{\mathbf{X}}$ and their singular values/vectors as functions of the signal parameters and the filter matrix. For example, the Vandermonde decomposition of $\mathbf{X}$ in (2.7) can be written in the form

$$\mathbf{X} = [\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_K]\ \mathrm{diag}(c_1, c_2, \cdots, c_K)\ [\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_K]^T = \sum_{k=1}^{K} c_k \mathbf{s}_k \mathbf{t}_k^T$$

To express the SVD in terms of the signal parameters, we invoke the following elementary properties relating eigenvalue decompositions and SVD's. For $k = 1, \cdots, K$, the first K nonzero eigenvalues, $\tilde{\lambda}_k$, of $\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H$ and $\tilde{\mathbf{X}}^H\tilde{\mathbf{X}}$ are $\tilde{\sigma}_k^2$. The corresponding eigenvectors of $\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H$ and $\tilde{\mathbf{X}}^H\tilde{\mathbf{X}}$ are $\tilde{\mathbf{u}}_k$ and $\tilde{\mathbf{v}}_k$, respectively. When $K = 2$, the prefiltered data matrix is simply

$$\tilde{\mathbf{X}} = (c_1\mathbf{s}_1\mathbf{t}_1^T + c_2\mathbf{s}_2\mathbf{t}_2^T)\mathbf{H}_r = (\sigma_1\mathbf{u}_1\mathbf{v}_1^H + \sigma_2\mathbf{u}_2\mathbf{v}_2^H)\mathbf{H}_r$$

Based on this description and our earlier definitions, the following relationships between the SVD and the signal parameters are proven in Theorem 3-ii at the end of this section. The left and right singular vectors $\tilde{\mathbf{u}}_k$ and $\tilde{\mathbf{v}}_k$ $(k = 1, 2)$ can be written as the linear combinations of $\mathbf{s}_1, \mathbf{s}_2$ and $\mathbf{t}_1^*, \mathbf{t}_2^*$:

$$\tilde{\mathbf{u}}_k = \tilde{b}_{uk1}\mathbf{s}_1 + \tilde{b}_{uk2}\mathbf{s}_2, \qquad \tilde{\mathbf{v}}_k = \mathbf{H}_r^H(\tilde{b}_{vk1}\mathbf{t}_1^* + \tilde{b}_{vk2}\mathbf{t}_2^*)$$

The eigenvalues $\tilde{\lambda}_k = \tilde{\sigma}_k^2$ $(k = 1, 2)$ are the roots of the quadratic equation:

$$\begin{aligned}
&\tilde{\lambda}^2 - [(\alpha_{11}\tilde{\beta}_{11}|c_1|^2 + \alpha_{22}\tilde{\beta}_{22}|c_2|^2 + 2\mathrm{Re}(\alpha_{12}^*\tilde{\beta}_{12}c_1c_2^*)]\tilde{\lambda} \\
&+|c_1|^2|c_2|^2(\alpha_{11}\alpha_{22} - \alpha_{12}\alpha_{21})(\tilde{\beta}_{11}\tilde{\beta}_{22} - \tilde{\beta}_{12}\tilde{\beta}_{21}) = 0
\end{aligned}$$

If no filter matrix is applied, or $\mathbf{H}_r = \mathbf{I}$, then $\tilde{\beta}_{ki} = \beta_{ki}$, the above quadratic equation reduces to that of λ_k, and $\mathbf{u}_k = b_{uk1}\mathbf{s}_1 + b_{uk2}\mathbf{s}_2$, $\mathbf{v}_k = b_{vk1}\mathbf{t}_1^* + b_{vk2}\mathbf{t}_2^*$, as given in [53].

Consider a special case of two ideally nonoverlapping sinusoids without applying prefiltering. Suppose L and M are even, it can be easily verified that $\alpha_{11} = \alpha_{22} = L, \beta_{11} = \beta_{22} = M, \alpha_{12} = \alpha_{21} = 0, \beta_{12} = \beta_{21} = 0$. This means that $\mathbf{s}_1$ and $\mathbf{s}_2$, $\mathbf{t}_1^*$ and $\mathbf{t}_2^*$ are already orthogonal and can be chosen as the singular vectors $\mathbf{u}_1$ and $\mathbf{u}_2$, $\mathbf{v}_1$ and $\mathbf{v}_2$ after appropriate scaling. The solution λ_k to the quadratic equation in this case is also quite simple. Without loss of generality, the peak with the largest power is numbered as the first peak. Then $\lambda_1 = \sigma_1^2 = \alpha_{11}\beta_{11}|c_1|^2 = LM|c_1|^2, \lambda_2 = \sigma_2^2 = \alpha_{22}\beta_{22}|c_2|^2 = LM|c_2|^2$.

When the two-peak signal is more general, and not ideally nonoverlapping, closed-form solutions of σ_k and $\tilde{\sigma}_k$ $(k = 1, 2)$ are always obtainable from the general root expressions of the quadratic equations.

3.4.3 SVD relationships between X and $\tilde{\mathbf{X}}$

In the previous subsection, we have derived the explicit expressions of $\tilde{\sigma}_k, \sigma_k$ etc. in function of the signal parameters. Here, we are going to compare directly the singular values/vectors of the filtered and unfiltered data matrix.

Recall that $\tilde{\mathbf{U}}_K = \mathbf{U}_K\mathbf{\Omega}_1$ (or see Theorem 3-i). For $K = 2$, we define the 2×2 matrix $\mathbf{\Omega}_1$ as $[\omega_{ki}]$, where ω_{ki} is the element on the kth row and

ith column of $\mathbf{\Omega}_1$. After some manipulations (see Theorem 3-**iii** at the end of this section), we get the general formulae below, which clearly reveal the relationships between the SVD of the filtered data matrix and the original one:

$$\begin{aligned}
\tilde{\sigma}_1^2 &= \frac{1}{2}\left[(\sigma_1^2\eta_{11}+\sigma_2^2\eta_{22}) + \sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2+(\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22})^2}\right] \\
\tilde{\sigma}_2^2 &= \frac{1}{2}\left[(\sigma_1^2\eta_{11}+\sigma_2^2\eta_{22}) - \sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2+(\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22})^2}\right] \\
|\omega_{11}|^2 &= |\omega_{22}|^2 = \frac{1}{2} + \frac{\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22}}{2\sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2+(\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22})^2}} \\
|\omega_{21}|^2 &= |\omega_{12}|^2 = \frac{1}{2} - \frac{\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22}}{2\sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2+(\sigma_1^2\eta_{11}-\sigma_2^2\eta_{22})^2}}
\end{aligned} \tag{3.4}$$

In general, the factor $\eta_{ki} = \mathbf{v}_k^H \mathbf{H}_r \mathbf{H}_r^H \mathbf{v}_i$ depends on both the signal parameters and the filter matrix. Expressing $\mathbf{v}_k$ as a linear combination of $\mathbf{t}_k^*$, we get:

$$\eta_{ki} = b_{vk1}^* b_{vi1} \tilde{\beta}_{11} + b_{vk2}^* b_{vi2} \tilde{\beta}_{22} + b_{vk1}^* b_{vi2} \tilde{\beta}_{12} + b_{vk2}^* b_{vi1} \tilde{\beta}_{21} \tag{3.5}$$

For simplification, we consider three ideal cases of filtering:

a. both sinusoids (peak 1 and 2) are successfully selected.

b. the first sinusoid (peak 1) is successfully selected and the second sinusoid (peak 2) is completely suppressed.

c. the second sinusoid (peak 2) is successfully selected and the first sinusoid (peak 1) is completely suppressed.

By *successful selection* and *complete suppression* we mean that the filtering retains the wanted peak without distortion. This can be realized under the assumption that an ideal rectangular filter is designed and the two peaks of the original signal have both zero damping factors such that infinitely long tails in the spectrum are avoided. The values of the factors $\tilde{\beta}_{ki}$ and thus η_{ki} for the three ideal cases of filtering are derived in Theorem 3-**iv** and listed in Table 3.1. Substituting the values of η_{ki} into (3.4) yields neat expressions of the SVD relationship between $\tilde{\mathbf{X}}$ and $\mathbf{X}$ as listed in the same table. These relationships clearly indicate that the wanted peaks are emphasized while the unwanted peaks are suppressed successfully after prefiltering via matrix multiplication.

Of course the results listed in Table 3.1 are derived in Theorem 3-**iii, iv** using the following assumption: (1) the filter is an ideal rectangular bandpass filter preserving the wanted peaks and ruling out the unwanted peaks in the

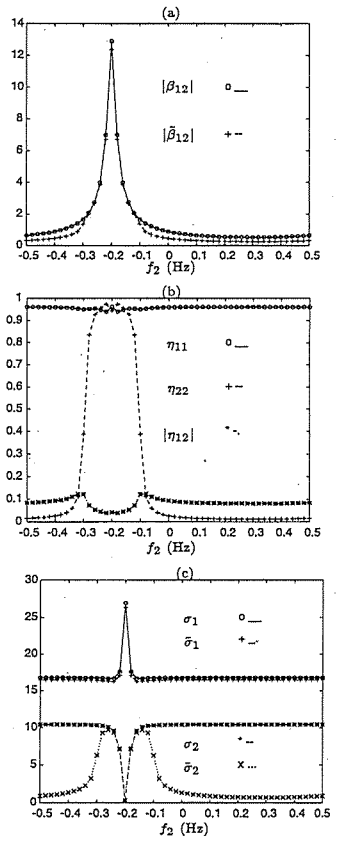


Figure 3.2: (a) $|\beta_{12}|$ and $|\tilde{\beta}_{12}|$ (b) $|\eta_{11}|$, $|\eta_{22}|$ and $|\eta_{12}|$ (c) σ_1, σ_2, $\tilde{\sigma}_1$ and $\tilde{\sigma}_2$ of a two-peak signal with $f_1 = -0.2$ Hz, f_2 variable, $d_1 = 0.03$ rad/s, $d_2 = 0.05$ rad/s, and $c_1 = c_2 = 1$. $f_l = f_1 - 0.1$ Hz, $f_u = f_1 + 0.1$ Hz, $q = 99$, $L = 62$, $M = 60$, $N = 121$.

filter characteristics	**a.** select both peaks	**b.** select the 1st peak	**c.** select the 2nd peak
$\tilde{\beta}_{11} =$	β_{11}	β_{11}	0
$\tilde{\beta}_{22} =$	β_{22}	0	β_{22}
$\tilde{\beta}_{12} =$	β_{12}	0	0
$\tilde{\beta}_{21} =$	β_{21}	0	0
$\eta_{11} =$	1	1	0
$\eta_{22} =$	1	0	1
$\eta_{12} =$	0	0	0
$\eta_{21} =$	0	0	0
$\tilde{\sigma}_1 =$	σ_1	σ_1	σ_2
$\tilde{\sigma}_2 =$	σ_2	0	0
$\boldsymbol{\Omega}_1 =$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

Table 3.1: Three ideal cases of filtering using a rectangular filter.

spectrum; (2) the damping factors of the sinusoids are zero (which enables a successful selection). In practice, the filter passband is not exactly rectangular and the signal to be processed is damped. But, again, a nice feature of the SVD is its robustness with respect to mild violations of these conditions. If both peaks are to be selected and the two peaks are sufficiently narrow and not far away in the spectrum, which means that d_1 and d_2 are close to 0, then the two peaks can be enhanced considerably by a proper design of the filter preserving the two peaks and suppressing the noise outside the passband. If only one of the two peaks is wanted, the selected peak can also be emphasized, provided that the wanted and unwanted peaks are sufficiently narrow and well separated, that is, d_1 and d_2 are close to 0 and $|f_2 - f_1|/f_s$ is close to 0.5. To give an intuitive idea of these results in general cases, we show in Fig. 3.2 the variations of $|\tilde{\beta}_{12}|, |\eta_{ki}|, \sigma_k$ and $\tilde{\sigma}_k$ $(k, i = 1, 2)$ when one of the two peaks varies along the frequency axis and peak 1 is selected. It can clearly be seen that $\tilde{\sigma}_1 \approx \sigma_1, \tilde{\sigma}_2 \approx \sigma_2$ when f_2 is around $f_1 = -0.2$ Hz or inside the filter passband, that is, when both peaks are selected; while $\tilde{\sigma}_1 \approx \sigma_1, \tilde{\sigma}_2 \approx 0$ when f_2 is around $f_1 + 0.5$ Hz $= 0.3$ Hz, which corresponds to the case that the two peaks are nonoverlapping and only the first peak is selected.

Theorem 3 *Let* $\mathbf{X}$ *be the rank-K Hankel data matrix of dimension* $L \times M$ *as defined in (2.1), and* $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$ *be the* $L \times M$ *filtered data matrix where* $\mathbf{H}_r$ *is full rank and square. Suppose the SVD of* $\mathbf{X}$ *and* $\tilde{\mathbf{X}}$ *are* $\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^H$ *and* $\tilde{\mathbf{X}} = \tilde{\mathbf{U}}\tilde{\boldsymbol{\Sigma}}\tilde{\mathbf{V}}^H$.

i. *There exists a unitary matrix* $\boldsymbol{\Omega} \in \mathbb{C}^{L\times L}$ *such that* $\tilde{\mathbf{U}} = \mathbf{U}\boldsymbol{\Omega}$, *where* $\boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{\Omega}_1 & 0 \\ 0 & \boldsymbol{\Omega}_2 \end{bmatrix}$ *and* $\boldsymbol{\Omega}_1 \in \mathbb{C}^{K\times K}$.

ii. *For a signal with* $K = 2$ *exponentials, the singular vectors* $\tilde{\mathbf{u}}_k$ *and* $\tilde{\mathbf{v}}_k$ $(k = 1, 2)$ *can be expressed as linear combinations of* $\mathbf{s}_1$, $\mathbf{s}_2$ *and* $\mathbf{t}_1^*$, $\mathbf{t}_2^*$ *defined in definition 1*

$$\tilde{\mathbf{u}}_k = \tilde{b}_{uk1}\mathbf{s}_1 + \tilde{b}_{uk2}\mathbf{s}_2, \qquad \tilde{\mathbf{v}}_k = \mathbf{H}_r^H(\tilde{b}_{vk1}\mathbf{t}_1^* + \tilde{b}_{vk2}\mathbf{t}_2^*)$$

the eigenvalues $\tilde{\lambda}_k$, $k = 1, 2$, *are the roots of the following quadratic equation using definition 2*

$$\tilde{\lambda}^2 - [\alpha_{11}\tilde{\beta}_{11}|c_1|^2 + \alpha_{22}\tilde{\beta}_{22}|c_2|^2 + 2Re(\alpha_{12}^*\tilde{\beta}_{12}c_1c_2^*)]\tilde{\lambda} + |c_1|^2|c_2|^2(\alpha_{11}\alpha_{22} - \alpha_{12}\alpha_{21})(\tilde{\beta}_{11}\tilde{\beta}_{22} - \tilde{\beta}_{12}\tilde{\beta}_{21}) = 0$$

and $\tilde{\nu}_{uk} = \tilde{b}_{uk1}/\tilde{b}_{uk2}$, $k = 1, 2$ *are the roots of the quadratic equation*

$$(\alpha_{21}\tilde{\beta}_{22}|c_2|^2 + \alpha_{11}\tilde{\beta}_{21}c_1^*c_2)\tilde{\nu}_u^2 + (\alpha_{22}\tilde{\beta}_{22}|c_2|^2 - \alpha_{11}\tilde{\beta}_{11}|c_1|^2 + \alpha_{12}\tilde{\beta}_{21}c_1^*c_2 - \alpha_{21}\tilde{\beta}_{12}c_1c_2^*)\tilde{\nu}_u - (\alpha_{12}\tilde{\beta}_{11}|c_1|^2 + \alpha_{22}\tilde{\beta}_{12}c_1c_2^*) = 0$$

iii. *For a signal with* $K = 2$ *exponentials, define* $\boldsymbol{\Omega} = \begin{bmatrix} \omega_{11} & \omega_{12} \\ \omega_{21} & \omega_{22} \end{bmatrix}$. *The relationship between the singular values* $\tilde{\sigma}_k$ *and* σ_k *as well as the relationship between the singular vectors* $\tilde{\mathbf{u}}_k$ *and* $\mathbf{u}_k$ *can be expressed as*

$$\begin{aligned}
\tilde{\sigma}_1^2 &= \frac{1}{2}\left[(\sigma_1^2\eta_{11} + \sigma_2^2\eta_{22}) + \sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2}\right] \\
\tilde{\sigma}_2^2 &= \frac{1}{2}\left[(\sigma_1^2\eta_{11} + \sigma_2^2\eta_{22}) - \sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2}\right] \\
|\omega_{11}|^2 &= |\omega_{22}|^2 = \frac{1}{2} + \frac{\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22}}{2\sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2}} \\
|\omega_{21}|^2 &= |\omega_{12}|^2 = \frac{1}{2} - \frac{\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22}}{2\sqrt{4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2}}
\end{aligned}$$

iv. *For a signal with ideally nonoverlapping exponentials, apply a rectangular filter. Then,*

1. we have

$\tilde{\beta}_{ki} = \beta_{ki}$ *when both peak k and i are totally encompassed inside the filter passband*

$\tilde{\beta}_{ki} = 0$ *when either peak k or i is totally ruled out of the filter passband*

2. using the classification in Subsection 3.4.3, we have

a. $\tilde{\beta}_{11} = \beta_{11}$, $\tilde{\beta}_{22} = \beta_{22}$, $\tilde{\beta}_{12} = \beta_{12}$, $\tilde{\beta}_{21} = \beta_{21}$;
$\eta_{11} = \eta_{22} = 1$, $\eta_{12} = \eta_{21} = 0$.

b. $\tilde{\beta}_{11} = \beta_{11}$, $\tilde{\beta}_{22} = \tilde{\beta}_{12} = \tilde{\beta}_{21} = 0$;
$\eta_{11} = 1$, $\eta_{22} = \eta_{12} = \eta_{21} = 0$.

c. $\tilde{\beta}_{22} = \beta_{22}$, $\tilde{\beta}_{11} = \tilde{\beta}_{12} = \tilde{\beta}_{21} = 0$;
$\eta_{22} = 1$, $\eta_{11} = \eta_{12} = \eta_{21} = 0$.

Proof.

i. Let

$$\mathbf{\Omega} = \mathbf{U}^{\mathbf{H}}\tilde{\mathbf{U}} = \begin{bmatrix} \mathbf{\Omega}_1 & \mathbf{\Omega}_4 \\ \mathbf{\Omega}_3 & \mathbf{\Omega}_2 \end{bmatrix}$$

Since $\mathbf{U}$ and $\tilde{\mathbf{U}}$ are both unitary matrices, $\mathbf{\Omega}$ is also unitary and

$$\tilde{\mathbf{U}} = \mathbf{U}\mathbf{\Omega} = [\mathbf{U}_K,\ \mathbf{U}_{M-K}] \begin{bmatrix} \mathbf{\Omega}_1 & \mathbf{\Omega}_4 \\ \mathbf{\Omega}_3 & \mathbf{\Omega}_2 \end{bmatrix}$$

From

$$\mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r = \tilde{\mathbf{U}}_K\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{V}}_K^H$$

$$\Rightarrow \quad \tilde{\mathbf{U}}_K = \mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\tilde{\mathbf{V}}_K\tilde{\mathbf{\Sigma}}_K^{-1}$$

$$\Rightarrow \quad \mathbf{\Omega}_1 = \mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\tilde{\mathbf{V}}_K\tilde{\mathbf{\Sigma}}_K^{-1}, \text{ and } \mathbf{\Omega}_3 = 0$$

Substituting $\mathbf{\Omega}_3 = 0$ into $\mathbf{\Omega}^H\mathbf{\Omega} = \mathbf{I}$ and $\mathbf{\Omega}\mathbf{\Omega}^H = \mathbf{I}$ leads to $\mathbf{\Omega}_4 = 0$. It is straight forward that $\mathbf{\Omega}_1$ and $\mathbf{\Omega}_2$ are unitary matrices.

ii. Using the Vandermonde decomposition of $\mathbf{X}$ and $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$, we have

$$\begin{aligned} \tilde{\mathbf{X}}\tilde{\mathbf{X}}^H &= (c_1\mathbf{s}_1\mathbf{t}_1^T + c_2\mathbf{s}_2\mathbf{t}_2^T)\mathbf{H}_r\mathbf{H}_r^H(c_1\mathbf{s}_1\mathbf{t}_1^T + c_2\mathbf{s}_2\mathbf{t}_2^T)^H \\ &= |c_1|^2\tilde{\beta}_{11}\mathbf{s}_1\mathbf{s}_1^H + c_1c_2^*\tilde{\beta}_{12}\mathbf{s}_1\mathbf{s}_2^H + c_1^*c_2\tilde{\beta}_{21}\mathbf{s}_2\mathbf{s}_1^H \\ &\quad +|c_2|^2\tilde{\beta}_{22}\mathbf{s}_2\mathbf{s}_2^H \end{aligned} \tag{3.6}$$

A direct observation is that the singular vectors $\tilde{\mathbf{u}}_k$ can be written as a linear combination of $\mathbf{s}_1, \mathbf{s}_2$

$$\tilde{\mathbf{u}}_k = \tilde{b}_{uk1}\mathbf{s}_1 + \tilde{b}_{uk2}\mathbf{s}_2$$

or $\quad \tilde{\mathbf{u}}_k = \tilde{b}_{uk2}(\tilde{\nu}_{uk}\mathbf{s}_1 + \mathbf{s}_2)$, when $\tilde{b}_{uk2} \neq 0$, $\tilde{\nu}_{uk} = \tilde{b}_{uk1}/\tilde{b}_{uk2}$

Hence,

$$\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H(\tilde{\nu}_{uk}\mathbf{s}_1 + \mathbf{s}_2) = \tilde{\lambda}_k(\tilde{\nu}_{uk}\mathbf{s}_1 + \mathbf{s}_2) \quad \text{for } k = 1, 2$$

Using (3.6), it can be expanded as

$$\begin{aligned} &\tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{11}|c_1|^2\mathbf{s}_1 + \alpha_{12}\tilde{\beta}_{11}|c_1|^2\mathbf{s}_1 + \tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{12}c_1c_2^*\mathbf{s}_1 \\ +\ &\alpha_{22}\tilde{\beta}_{12}c_1c_2^*\mathbf{s}_1 + \tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{21}c_1^*c_2\mathbf{s}_2 + \alpha_{12}\tilde{\beta}_{21}c_1^*c_2\mathbf{s}_2 \\ +\ &\tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{22}|c_2|^2\mathbf{s}_2 + \alpha_{22}\tilde{\beta}_{22}|c_2|^2\mathbf{s}_2 = \tilde{\lambda}_k\tilde{\nu}_{uk}\mathbf{s}_1 + \tilde{\lambda}_k\mathbf{s}_2 \end{aligned} \tag{3.7}$$

Generally, $\mathbf{s}_1, \mathbf{s}_2$ are linearly independent, that is, if $b_1\mathbf{s}_1 + b_2\mathbf{s}_2 = 0$, then $b_1 = b_2 = 0$. Applying this to (3.7) gives us

$$\begin{aligned} \tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{11}|c_1|^2 \ +\ &\alpha_{12}\tilde{\beta}_{11}|c_1|^2 + \tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{12}c_1c_2^* \\ &+\alpha_{22}\tilde{\beta}_{12}c_1c_2^* = \tilde{\lambda}_k\tilde{\nu}_{uk} \end{aligned} \tag{3.8}$$

$$\begin{aligned} \tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{21}c_1^*c_2 \ +\ &\alpha_{12}\tilde{\beta}_{21}c_1^*c_2 + \tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{22}|c_2|^2 \\ &+\alpha_{22}\tilde{\beta}_{22}|c_2|^2 = \tilde{\lambda}_k \end{aligned} \tag{3.9}$$

Substituting, respectively,

$$\tilde{\lambda}_k = \tilde{\nu}_{uk}\alpha_{11}\tilde{\beta}_{21}c_1^*c_2 + \alpha_{12}\tilde{\beta}_{21}c_1^*c_2 + \tilde{\nu}_{uk}\alpha_{21}\tilde{\beta}_{22}|c_2|^2 + \alpha_{22}\tilde{\beta}_{22}|c_2|^2$$

and

$$\tilde{\nu}_{uk} = (\tilde{\lambda}_k - \alpha_{12}\tilde{\beta}_{21}c_1^*c_2 - \alpha_{22}\tilde{\beta}_{22}|c_2|^2)/(\alpha_{11}\tilde{\beta}_{21}c_1^*c_2 + \alpha_{21}\tilde{\beta}_{22}|c_2|^2)$$

obtained from (3.9), into (3.8) leads us to the quadratic equations of $\tilde{\nu}_u$ and $\tilde{\lambda}$ in the theorem.

Similarly, starting from the expansion of $\tilde{\mathbf{X}}^H\tilde{\mathbf{X}}$ instead of $\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H$, we get

$$\tilde{\mathbf{v}}_k = \mathbf{H}_r^H(\tilde{b}_{vk1}\mathbf{t}_1^* + \tilde{b}_{vk2}\mathbf{t}_2^*)$$

iii. Equating

$$\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H = \tilde{\mathbf{U}}_K\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{U}}_K^H$$

and

$$\tilde{\mathbf{X}}\tilde{\mathbf{X}}^H = \mathbf{U}_K\mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{V}_K\mathbf{\Sigma}_K\mathbf{U}_K^H$$

and using $\tilde{\mathbf{U}}_K = \mathbf{U}_K\mathbf{\Omega}_1$, we get the relationship between $\tilde{\mathbf{\Sigma}}_K$ and $\mathbf{\Sigma}_K$

$$\mathbf{\Omega}_1\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{\Sigma}}_K\mathbf{\Omega}_1^H = \mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{V}_K\mathbf{\Sigma}_K \tag{3.10}$$

For $K = 2$, we have

$$\mathbf{\Omega}_1\tilde{\mathbf{\Sigma}}_K\tilde{\mathbf{\Sigma}}_K\mathbf{\Omega}_1^H = \begin{bmatrix} \tilde{\sigma}_1^2|\omega_{11}|^2 + \tilde{\sigma}_2^2|\omega_{12}|^2 & \tilde{\sigma}_1^2\omega_{11}\omega_{21}^* + \tilde{\sigma}_2^2\omega_{22}^*\omega_{12} \\ \tilde{\sigma}_1^2\omega_{11}^*\omega_{21} + \tilde{\sigma}_2^2\omega_{22}\omega_{12}^* & \tilde{\sigma}_1^2|\omega_{21}|^2 + \tilde{\sigma}_2^2|\omega_{22}|^2 \end{bmatrix}$$

$$\mathbf{\Sigma}_K\mathbf{V}_K^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{V}_K\mathbf{\Sigma}_K = \begin{bmatrix} \sigma_1^2\mathbf{v}_1^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_1 & \sigma_1\sigma_2\mathbf{v}_1^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_2 \\ \sigma_1\sigma_2\mathbf{v}_2^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_1 & \sigma_2^2\mathbf{v}_2^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_2 \end{bmatrix}$$
$$= \begin{bmatrix} \sigma_1^2\eta_{11} & \sigma_1\sigma_2\eta_{12} \\ \sigma_1\sigma_2\eta_{21} & \sigma_2^2\eta_{22} \end{bmatrix}$$

Using (3.10) and $\mathbf{\Omega}_1^H\mathbf{\Omega}_1 = \mathbf{\Omega}_1\mathbf{\Omega}_1^H = \mathbf{I}$, we have

$$\begin{aligned} \tilde{\sigma}_1^2|\omega_{11}|^2 + \tilde{\sigma}_2^2(1 - |\omega_{11}|^2) &= \sigma_1^2\eta_{11}, && (3.11) \\ \tilde{\sigma}_1^2(1 - |\omega_{11}|^2) + \tilde{\sigma}_2^2|\omega_{11}|^2 &= \sigma_2^2\eta_{22}, && (3.12) \\ \tilde{\sigma}_1^2\omega_{11}^*\omega_{21} + \tilde{\sigma}_2^2\omega_{22}\omega_{12}^* &= \sigma_1\sigma_2\eta_{21}, && (3.13) \\ |\omega_{11}| = |\omega_{22}|, \quad & |\omega_{21}| = |\omega_{12}|, && (3.14) \end{aligned}$$

First derive $\tilde{\sigma}_k$ from (3.11) and (3.12):

$$\begin{aligned} \tilde{\sigma}_1^2 &= \frac{\sigma_1^2\eta_{11}|\omega_{11}|^2 - \sigma_2^2\eta_{22}(1 - |\omega_{11}|^2)}{2|\omega_{11}|^2 - 1} \\ \tilde{\sigma}_2^2 &= \frac{-\sigma_1^2\eta_{11}(1 - |\omega_{11}|^2) + \sigma_2^2\eta_{22}|\omega_{11}|^2}{2|\omega_{11}|^2 - 1} \end{aligned} \tag{3.15}$$

then substitute these into (3.13). After some manipulations and using (3.14), we have

$$(|\omega_{11}|^4 - |\omega_{11}|^2)[4\sigma_1^2\sigma_2^2|\eta_{21}|^2 + (\sigma_1^2\eta_{11} - \sigma_2^2\eta_{22})^2] + \sigma_1^2\sigma_2^2|\eta_{21}|^2 = 0$$

Solve for $|\omega_{11}|^2$ from this equation and put it back into (3.15). Considering $\tilde{\sigma}_1 \geq \tilde{\sigma}_2$, the theorem is proven.

iv. 1. Let $\mathbf{x}_k = \mathbf{t}_k^*/\|\mathbf{t}_k\|_2, \tilde{\mathbf{x}}_k = \mathbf{H}_r^T\mathbf{x}_k$, then $\|\mathbf{x}\|_2^2 = \frac{1}{r}\sum_{m=1}^r |\mathcal{X}(m)|^2 = 1$ (normalized), and $\tilde{\mathcal{X}}_k(m) = \mathcal{X}_k(m)\mathcal{H}(m), m = 1, 2, \ldots, r$, where $\mathcal{X}_k, \tilde{\mathcal{X}}_k$ and $\mathcal{H}$ are the r-point DFT of $\mathbf{x}_k, \tilde{\mathbf{x}}_k$ and the filter impulse response, respectively. Using DFT properties, it follows that

$$\begin{aligned} \frac{\tilde{\beta}_{ki}}{\sqrt{\beta_{kk}\beta_{ii}}} &= \frac{\mathbf{t}_k^T\mathbf{H}_r\mathbf{H}_r^H\mathbf{t}_i^*}{\|\mathbf{t}_k\|_2\|\mathbf{t}_i\|_2} = \mathbf{x}_k^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{x}_i = \tilde{\mathbf{x}}_k^H\tilde{\mathbf{x}}_i \\ &= \frac{1}{r}\sum_{m=1}^r \tilde{\mathcal{X}}_k^*(m)\tilde{\mathcal{X}}_i(m) \quad \text{(see Theorem (1))} \\ &= \frac{1}{r}\sum_{m=1}^r \mathcal{X}_k^*(m)\mathcal{X}_i(m)|\mathcal{H}(m)|^2 \text{ (convolution theorem)} \\ &= \begin{cases} \frac{1}{r}\sum_{m=1}^r \mathcal{X}_k^*(m)\mathcal{X}_i(m) = \mathbf{x}_k^H\mathbf{x}_i = \frac{\beta_{ki}}{\sqrt{\beta_{kk}\beta_{ii}}} \\ \qquad \text{(when both peak } k \text{ and } i \text{ are totally} \\ \qquad \text{encompassed by the filter passband)} \\ 0 \qquad \text{(when either peak } k \text{ or } i \text{ is totally} \\ \qquad \text{ruled out of the filter passband)} \end{cases} \end{aligned}$$

2. Using the above results, the values of $\tilde{\beta}_{ki}$ under the assumptions **a**, **b**, **c** in Table 3.1 are easily derived. Since $\eta_{ki} = \mathbf{v}_k^H\mathbf{H}_r\mathbf{H}_r^H\mathbf{v}_i = b_{vk1}^*b_{vi1}\tilde{\beta}_{11} + b_{vk2}^*b_{vi2}\tilde{\beta}_{22} + b_{vk1}^*b_{vi2}\tilde{\beta}_{12} + b_{vk2}^*b_{vi1}\tilde{\beta}_{21}$ from (3.5), using the values of $\tilde{\beta}_{ki}$, we have

a. When both peak 1 and peak 2 are selected, $\eta_{ki} = b_{vk1}^*b_{vi1}\beta_{11} + b_{vk2}^*b_{vi2}\beta_{22} + b_{vk1}^*b_{vi2}\beta_{12} + b_{vk2}^*b_{vi1}\beta_{21} = \mathbf{v}_k^H\mathbf{v}_i$. Due to the orthogonality and normalization properties of singular vectors, $\mathbf{v}$, it follows that $\eta_{kk} = 1$ and $\eta_{ki} = 0, (k \neq i)$.

b. When only peak 1 is selected, and peak 1 and 2 are two ideally nonoverlapping peaks, $\mathbf{v}_k = \mathbf{t}_k^*/\|\mathbf{t}_k\|$ from **ii**, i.e., $|b_{vkk}| = 1/\|\mathbf{t}_k\| = 1/\sqrt{\beta_{kk}}, b_{vki} = 0, (k \neq i)$. From the results of $\tilde{\beta}_{ki}$, we get $\eta_{ki} = b_{vk1}^*b_{vi1}\beta_{11}$, that is, $\eta_{11} = 1$ and $\eta_{22} = \eta_{12} = \eta_{21} = 0$.

c. When only peak 2 is selected, the proof is similar to that of **b**.

□

The theorem for the case of left multiplication is similar and omitted.

3.5 SNR relationships between the prefiltered and original data matrices

The previous section is based on the assumption of *pure* sinusoids without noise. In this section, we examine the SNR relationship between the prefiltered and original data matrices *when there is noise present* with a noise covariance matrix of $\mathbf{E}^H\mathbf{E} = \mu^2\mathbf{I} = L\nu^2\mathbf{I}$, as assumed in (2.4) of Chapter 2, Section 2.2.1.

In Monte-Carlo studies, we have been using the *peak* SNR, which is defined in Chapter 2, Section 2.3. It is computed from the peak amplitude, a_k, and the variance of the complex-valued white noise, $2\sigma_\nu^2$, as shown below.

$$\text{SNR} = 10\log_{10}\left(\frac{a_k^2}{2\sigma_\nu^2}\right) \quad \text{for peak } k$$

After prefiltering, however, the noise is not white anymore.

In this section, we use another definition of SNR [29], which is matrix oriented and denoted by SNR_m. Thus, the SNR_m for the data matrix $\mathbf{X} = \overset{\circ}{\mathbf{X}} + \mathbf{E}$ is

$$\text{SNR}_m = 10\log_{10}\frac{\|\overset{\circ}{\mathbf{X}}\|_F^2}{\|\mathbf{E}\|_F^2}\ (\text{dB}) = 10\log_{10}\frac{\sum_{k=1}^K \sigma_k^2}{M\mu^2}\ (\text{dB})$$

After prefiltering, we have $\tilde{\mathbf{X}} = \overset{\circ}{\tilde{\mathbf{X}}} + \tilde{\mathbf{E}} = \overset{\circ}{\mathbf{X}}\mathbf{H}_r + \mathbf{E}\mathbf{H}_r$ and

$$\widetilde{\text{SNR}}_m = 10\log_{10}\frac{\|\overset{\circ}{\tilde{\mathbf{X}}}\|_F^2}{\|\tilde{\mathbf{E}}\|_F^2}\ (\text{dB}) = 10\log_{10}\frac{\sum_{k=1}^K \tilde{\sigma}_k^2}{\|\mathbf{H}_r\|\mu^2}\ (\text{dB})$$

To simplify our study, we continue to use the assumptions of *idealized* cases of prefiltering in the previous section.

Comparison of $\|\mathbf{E}\|_F^2$ and $\|\tilde{\mathbf{E}}\|_F^2$:

In Theorem 2, it is shown that $\|\mathbf{H}\|_2 = \sigma_{H1} = 1$ for an ideal rectangular bandpass filter with a unit gain at passband. It follows that

$$\|\mathbf{H}_r\|_F^2 = \|\mathbf{H}\|_F^2 = \sum_{i=1}^M \sigma_{Hi}^2 \le M\sigma_{H1}^2 = M$$

which means that

$$\|\tilde{\mathbf{E}}\|_F^2 = \text{trace}(\tilde{\mathbf{E}}^H\tilde{\mathbf{E}}) = \text{trace}(\mathbf{H}_r^H\mathbf{H}_r)\mu^2 \le \|\mathbf{E}\|_F^2 = \text{trace}(\mathbf{E}^H\mathbf{E}) = M\mu^2$$

Actually, for FIR filter matrices defined in Section 3.2, we can find that

$$\|\mathbf{H}_{\text{FIR}t}\|_F^2 < \|\mathbf{H}_{\text{FIR}0}\|_F^2 = M\sum_{i=1}^q |h_i|^2 = \frac{M}{q}\sum_{m=1}^q |\mathcal{H}(m)|^2 = M\frac{f_u - f_l}{f_s}$$

Since $(f_u - f_l)/f_s < 1$ is always smaller than 1, this is a tighter bound for $\|\mathbf{H}\|_F^2$.

Right now, we have not found such a tighter bound for the IIR filter matrix defined in Section 3.2.

Comparison of $\|\overset{\circ}{\mathbf{X}}\|_F^2$ and $\|\overset{\circ}{\tilde{\mathbf{X}}}\|_F^2$:

Here, we consider the ideal cases of prefiltering when the signal is composed of only one and two sinusoids.

1. When there is only one sinusoid in the noiseless signal, it can be proven that $\tilde{\sigma}_1 = \sigma_1$ if this sinusoid is successfully selected. The proof is similar to that of Theorem 3 and thus omitted.

2. When there are two sinusoids in the noiseless signal, it can be found from Table 3.1 that

   a. $\tilde{\sigma}_1 + \tilde{\sigma}_2 = \sigma_1 + \sigma_2$, if both sinusoids are successfully selected;

   b. $\tilde{\sigma}_1 + \tilde{\sigma}_2 = \sigma_1 < \sigma_1 + \sigma_2$, if the first sinusoid is successfully selected and the second sinusoid is completely suppressed;

   c. $\tilde{\sigma}_1 + \tilde{\sigma}_2 = \sigma_2 < \sigma_1 + \sigma_2$, if the second sinusoid is successfully selected and the first sinusoid is completely suppressed.

Comparison of SNR_m and $\widetilde{\text{SNR}}_m$:

1. When there is only one sinusoid in the noiseless signal and this sinusoid is successfully selected, then the SNR_m is improved due to prefiltering since the SNR_m increment is

   $$\widetilde{\text{SNR}}_m - \text{SNR}_m = 10\log_{10}\frac{\tilde{\sigma}_1^2}{\|\mathbf{H}_r\|_F^2\mu^2}\cdot\frac{M\mu^2}{\sigma_1^2} = 10\log_{10}\frac{M}{\|\mathbf{H}_r\|_F^2}\ (dB)$$

   which is positive.

2. When there are two sinusoids in the noiseless signal, then

a. The SNR_m is improved due to prefiltering if both sinusoids are successfully selected since the SNR_m increment is

$$\widetilde{\mathrm{SNR}}_m - \mathrm{SNR}_m = 10\log_{10}\frac{\tilde{\sigma}_1^2+\tilde{\sigma}_2^2}{\|\mathbf{H}_r\|_F^2\mu^2}\cdot\frac{M\mu^2}{\sigma_1^2+\sigma_2^2} = 10\log_{10}\frac{M}{\|\mathbf{H}_r\|_F^2} \quad (dB)$$

which is positive.

b. The SNR_m is not always improved due to prefiltering if the first sinusoid is successfully selected and the second sinusoid is completely suppressed since the SNR_m increment is

$$\widetilde{\mathrm{SNR}}_m - \mathrm{SNR}_m = 10\log_{10}\frac{\tilde{\sigma}_1^2+\tilde{\sigma}_2^2}{\|\mathbf{H}_r\|_F^2\mu^2}\cdot\frac{M\mu^2}{\sigma_1^2} \quad (dB)$$

which is not always positive.

c. The SNR_m is not always improved due to prefiltering if the second sinusoid is successfully selected and the first sinusoid is completely suppressed since the SNR_m increment is

$$\widetilde{\mathrm{SNR}}_m - \mathrm{SNR}_m = 10\log_{10}\frac{\tilde{\sigma}_1^2+\tilde{\sigma}_2^2}{\|\mathbf{H}_r\|_F^2\mu^2}\cdot\frac{M\mu^2}{\sigma_2^2} \quad (dB)$$

which is not always positive.

In summary, when all the sinusoids are successfully selected, the SNR_m increment is

$$\widetilde{\mathrm{SNR}}_m - \mathrm{SNR}_m = 10\log_{10}\frac{M}{\|\mathbf{H}_r\|_F^2}\begin{cases} > 10\log_{10}\frac{f_s}{f_u-f_l} & \text{if } \mathbf{H}_r = \mathbf{H}_{\mathrm{FIR}t} \\ = 10\log_{10}\frac{f_s}{f_u-f_l} & \text{if } \mathbf{H}_r = \mathbf{H}_{\mathrm{FIR}0} \\ > 0 \text{ (at least)} & \text{if } \mathbf{H}_r = \mathbf{H}_{\mathrm{IIR}} \end{cases}$$

However, when only part of the sinusoids are selected, the total signal power is reduced, although the power of the selected sinusoids remains the same. Therefore, the SNR_m is not always increased.

3.6 HTLS_F prefiltering

Another prefiltering technique, denoted by HTLS_F, is based on the repetitive use of HTLS and is therefore called HTLS_F prefiltering. It is motivated by the technique proposed by Pijnappel et al. [32] for Kung's method. The idea is to first invoke HTLS to get a rough estimation of all K sinusoids comprising

the original signal and then subtract the signal that fits the unwanted peaks from the original one. As a result, the residual signal essentially consists of the wanted peaks and white noise only, which is actually a prefiltered signal. Finally, the wanted peaks in the selected region can be estimated by the HTLS method again. This frequency-selective technique is called HTLS_F-HTLS in this thesis and outlined below.

Algorithm 3

Outline of the HTLS_F-HTLS algorithm

Given. The model order estimates K and K_s of the original and residual signal respectively, and N uniformly sampled original data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel data matrix $\mathbf{X}$ as in (2.1).

Step 1. Apply the HTLS Algorithm 1 to $\mathbf{X}$ using model order K.

Step 2. Reconstruct an intermediate signal $\tilde{x}_n$ from the estimates of unwanted peaks.

Step 3. Arrange the prefiltered or residual data $\tilde{x}_n = x_n - \tilde{x}_n$ into an $L \times M$ Hankel matrix $\tilde{\mathbf{X}}$.

Step 4. Apply the HTLS Algorithm 1 to $\tilde{\mathbf{X}}$ using model order K_s.

The filtering effect of HTLS_F is different from that of a bandpass FIR or IIR filter due to the infinitely long tails of exponentially damped sinusoids in the spectrum. HTLS_F, in ideal cases, completely eliminates the unwanted peaks in the whole spectrum, without suppressing the tails of the wanted peaks. But a bandpass FIR or IIR filter not only keeps the tails of the unwanted peaks inside the passband but also reduces the tails of the wanted peaks outside the passband, especially if the wanted and unwanted peaks are strongly overlapping. However, when the unwanted peaks are also difficult to resolve so that the estimation of $\tilde{x}_n$ in Steps 1-2 of Algorithm 3 is not at all accurate, the final parameter estimation in Step 4 of Algorithm 3 based on the residual signal cannot be accurate either. In this case, a combination of the two prefiltering schemes called HTLS_F-$\mathbf{H}_{\mathrm{FIR}t}$-HTLS and HTLS_F-$\mathbf{H}_{\mathrm{IIR}}$-HTLS may help, where $\mathbf{H}_{\mathrm{FIR}t}$-HTLS and $\mathbf{H}_{\mathrm{IIR}}$-HTLS instead of HTLS is invoked in Step 4 of the HTLS_F-HTLS Algorithm 3 to further emphasize the wanted region and suppress the unwanted one.

Note that HTLS is not the only choice for prefiltering in Step 1. Besides HTLS_F prefiltering, we can use HSVD_F prefiltering, LP_F prefiltering, MP_F prefiltering, etc., prior to the final HTLS estimation. However, we only use HTLS_F prefiltering in this chapter.

3.7 Simulations

In this section, results of computer simulations are presented to confirm the theoretical predictions of earlier sections. Three different simulation examples are studied in various scenarios.

The algorithms considered here are the HTLS Algorithm 1 and FIR prefiltering, IIR prefiltering, HTLS_F prefiltering, followed by the HTLS Algorithm 1, outlined respectively in Sections 2.2.2, 3.3 and 3.6. Two types of FIR filters are tried out for different simulated signals: (1) a rectangular bandpass filter; (2) an exponential filter with frequency response $1/[1+4\pi^2(f-f_0)^2/\gamma^2]$, where f_0 is the center of the passband and γ is a constant controlling the bandwidth. Both filters use the Kaiser window with β parameter of 5.658 as the window function to truncate the impulse response [54]. These two types of FIR filter matrices are referred to by the subscript 1 and 2, respectively. For example, $\mathbf{H}_{\text{FIR}c1}$ and $\mathbf{H}_{\text{FIR}t1}$ are the circular and the FIR filter matrices generated from the first type of FIR filter, while $\mathbf{H}_{\text{FIR}c2}$ and $\mathbf{H}_{\text{FIR}t2}$ correspond to the second type. Similarly, $\mathbf{H}_{\text{IIR}}$ denotes the IIR filter matrix. In Example 1, the signal is real-valued and a 2nd-order IIR filter is used which is symmetric in positive and negative frequencies. Its z-transfer function is $\mathsf{H}(\mathsf{z}) = \xi(\mathsf{z}-1)(\mathsf{z}+1)/[(\mathsf{z}-\rho e^{j2\pi f_0})(\mathsf{z}-\rho e^{-j2\pi f_0})]$, where the poles are located at equal distance from the two sinusoids, $f_0 = (f_1+f_2)/2$. In Example 2, the signals are complex and we use the complex non-symmetric 1st-order IIR filters with transfer function $\mathsf{H}(\mathsf{z}) = \xi[\mathsf{z} - e^{j2\pi(f_0+f_s/2)}]/(\mathsf{z}-\rho e^{j2\pi f_0})$. ρ is a factor, fixed at 1.25 in our examples, controlling the bandwidth of the filter. ξ is a factor chosen such that the filter has a unit gain at the poles.

Example 1: *Signal-3*

In [51], a real-valued signal, called *Signal-3* in this thesis and described in Fig. A.3 and Table A.5, A.6 in Appendix A, has been simulated to study the effects of FIR prefiltering using a circular filter matrix $\mathbf{H}_{\text{FIR}c}$. Here, this signal is studied again using the prefiltered HTLS method and different filter matrices. Only the first type of FIR filters with a rectangular passband is considered here. Suppose both sinusoids are of interest and the frequency region is known. We choose the parameters for the HTLS and prefiltered HTLS algorithms as follows. $N = 30, M = q = 15, f_l = 0.0796$ Hz, $f_u = 0.191$ Hz. Since this signal is far from dying out at the 30th data point, a diagonal full-rank weighting matrix $\mathbf{W}$ is constructed from the Kaiser window with $\beta = 5.658$ to compensate the Gibbs phenomenon induced by the truncation of input data as discussed in section 3.3. When a filter matrix, say $\mathbf{H}$, is applied with or without weighting matrix $\mathbf{W}$, the method is called **H**-**W**-HTLS or **H**-HTLS, respectively.

Different prefiltering schemes using FIR filters and the HTLS method are compared as shown in Fig. 3.3. Although $\mathbf{H}_{\text{FIR}c1}$-HTLS lowers the failure rate of HTLS considerably, $\mathbf{H}_{\text{FIR}t1}$-HTLS performs better by eliminating the end effect. The reason is that the damping factors of the current signal are small and the 30-point signal has not died out sufficiently. This causes serious end effects affecting the accuracy of the $\mathbf{H}_{\text{FIR}c1}$-HTLS estimates, as mentioned in Section 3.2. The use of a weighting matrix $\mathbf{W}$ improves the resolution of the prefiltering method, $\mathbf{H}_{\text{FIR}c1}$-HTLS, because the signal dies out more quickly after weighting, and as a result, the end effect is reduced. The accuracy of all these algorithms is not much different, although the one using $\mathbf{H}_{\text{FIR}t1}$ seems a bit better. In conclusion, the application of FIR prefiltering using the filter matrix $\mathbf{H}_{\text{FIR}t}$ suppresses the noise outside the passband most effectively and enhances the resolution of all the peaks at comparable accuracy.

Observe that in Fig. 3.3(c), some accuracy curves at low SNR are closer to the CR bound curve than at high SNR. The reason is that the curves here are obtained by averaging over *good* runs, instead of all the 200 runs. Those bad runs or failures, which are eliminated, are typically the results of difficult signals with large noise. The exclusion of failures amounts to increasing the SNR of our simulated noisy signals, while CR bounds are calculated at the assumed SNR. This is where the inconsistency comes from. Actually, as we will see later on, this inconsistency becomes more serious when the failure rate gets higher, and the corresponding RMSE values may be even lower than the CR lower bounds at low SNR.

Example 2: *Signal-4*

This is a complex signal with three exponentially damped sinusoids as shown in Fig. A.4 and described in Tables A.7, A.8 in Appendix A. Suppose only the first peak is of interest as shown in Fig. 3.4, and the frequency, $f_1 = -0.2$ Hz, of this peak is known. When estimating other parameters of this peak, we can of course use the HTLS method to get parameters of all the peaks and pick out the interesting one. But this is a small peak severely corrupted

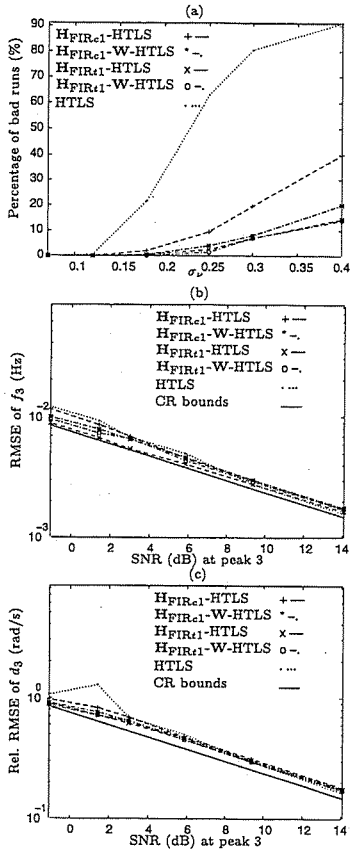


Figure 3.3: Example 1: (a) Failure rate versus σ_ν. (b) RMSE of $f_3 = 0.1$ Hz and (c) Relative RMSE of $d_3 = 0.02\pi$ rad/s versus SNR at peak 3.

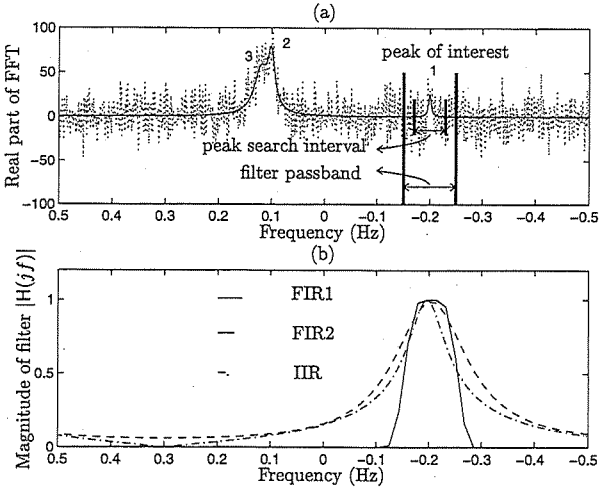


Figure 3.4: Example 2: (a) Noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.5$) spectra of *Signal-4*. (b) The frequency response of the filters used in the prefiltered HTLS method.

by noise and difficult to resolve by HTLS. Note also that this peak has a small damping factor and lies far away from the other two peaks, that is, it is approximately nonoverlapping with the other peaks. Hence, it is worthwhile to try our prefiltering technique in order to extract the interesting peak and enhance its resolution. Different prefiltering schemes are tried out using the parameters $N = 95, M = q = 57, f_l = -0.3$ Hz, $f_u = -0.1$ Hz, $\gamma = 0.5 f_s = 0.5$ Hz. Around the M=57th data point, the signal is dying out. $L = 39 < M$ is taken so that the total number of data points N, and hence the computational work, needed in the algorithm is small. The frequency interval in which the first peak should be resolved is $[f_1 - 0.03$ Hz, $f_1 + 0.03$ Hz $]$. If more than one estimated peak fall into this interval, the one with a frequency closest to f_1 is taken.

Fig. 3.5 compares the results of different methods. It is clear that the reso-

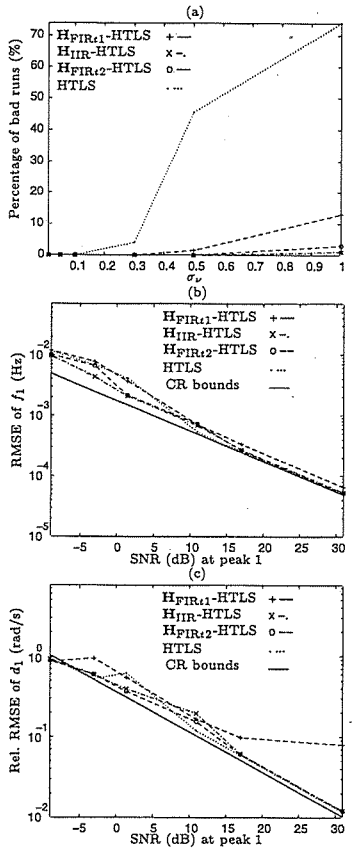


Figure 3.5: Example 2: (a) Failure rate versus σ_ν. (b) RMSE of f_1 and (c) Relative RMSE of d_1 versus SNR at peak 1.

lution of the selected peak 1 is greatly enhanced by all the prefiltering schemes used, especially at low SNR level. A more careful examination shows that $\mathbf{H}_{\mathrm{FIR}t1}$-HTLS is not as good as $\mathbf{H}_{\mathrm{FIR}t2}$-HTLS and $\mathbf{H}_{\mathrm{IIR}}$-HTLS in both resolution and parameter accuracy. In particular, the RMSE value of d_1 obtained by $\mathbf{H}_{\mathrm{FIR}t1}$-HTLS deviates from the CR bound at very high SNR. One explanation is that the FIR filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ introduces distortion due to the discontinuity at the initial conditions. Another possible explanation is that $\mathbf{H}_{\mathrm{FIR}t1}$ is constructed for a rectangular filter with abrupt changes around cut-off frequencies, while the spectrum of one exponentially damped sinusoid spreads smoothly along the frequency axis. The filtered sinusoid does not bear the original shape in the spectrum anymore and the signal characteristic is changed completely. The filter matrix $\mathbf{H}_{\mathrm{FIR}t2}$ corresponds to a bandpass filter with smooth spectrum centered at the peak of interest and hence preserves the signal spectrum better than $\mathbf{H}_{\mathrm{FIR}t1}$. Similarly, a high-order IIR filter with steep roll-off at cut-off frequencies performs poorly compared to a lower-order IIR filter (figures of these experiments are omitted). This artifact is more pronounced at high SNR. Therefore, when selecting only one sinusoid it is suggested to choose a bandpass filter that preserves the original signal characteristics as much as possible. Note that the phase characteristic of the filter is not important since the phase of the sinusoid is estimated from the original data as done in Section 2.2.2, instead of the filtered data.

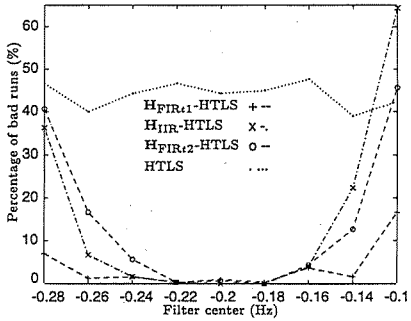


Figure 3.6: Example 2: Failure rate versus varying filter center. $\sigma_\nu = 0.5$.

In NMR spectroscopy, the frequency of a peak corresponds to a specific

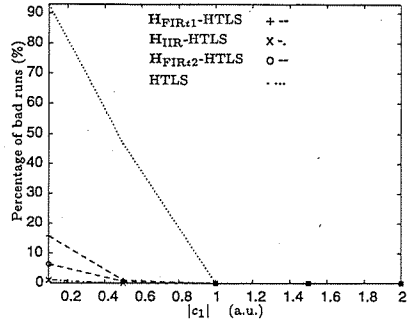


Figure 3.7: Example 2: Failure rate versus varying amplitude $|c_1|$ when other signal parameters are fixed. $\sigma_\nu = 0.5$.

metabolite and is therefore often known *a priori* from NMR tables and the estimation of damping factors and amplitudes is the main concern. In this case, the center frequency of the filter to be designed can be determined in advance. Alternatively the center frequency can also be estimated roughly from the FFT spectrum when the peak SNR is sufficiently high. Fig. 3.6 shows the sensitivity with respect to the chosen filter center. As expected, the prefiltered method based on a filter center slightly deviating from the exact center frequency of interest still performs better than the unfiltered method. The deviation from the exact filter center may be larger for the $H_{FIR\ell 1}$-HTLS method, which uses a filter with flat passband.

If we vary the absolute value of the complex amplitude, $|c_1|$, of the first sinusoid and keep the remaining parameters of this example unchanged, we can see that the smaller the $|c_1|$, the more we benefit from prefiltering, as shown in Fig. 3.7. In other words, prefiltering is more important when the peak of interest has a comparatively low SNR and thus needs to be emphasized.

Fig. 3.8 includes the results of HTLS$_F$ prefiltering also when the damping factor, d_1, of the first sinusoid is varied and the other parameters is fixed. The HTLS$_F$ prefiltering itself does not lower the failure rate of HTLS very much. It gives rather high failure rate at large d_1. This is because the HTLS$_F$ prefiltering is only effective when the unwanted peaks are easily resolved so that a sufficiently accurate first-step estimation can be obtained. When d_1 gets larger,

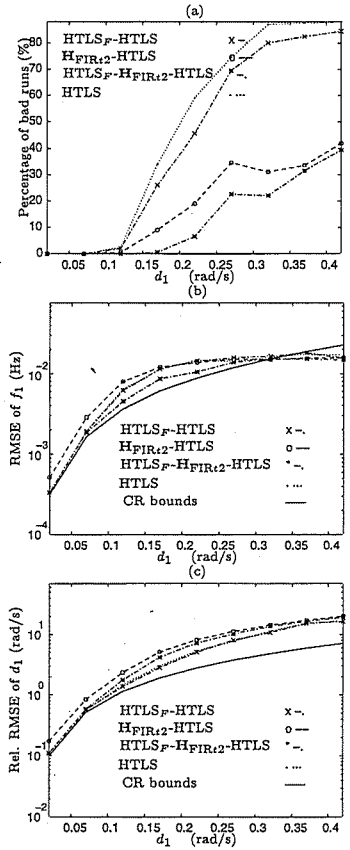


Figure 3.8: Example 2: (a) Failure rate, (b) RMSE of f_1 and (c) Relative RMSE of d_1 versus varying damping factor d_1 when other signal parameters are fixed. $\sigma_\nu = 0.5$.

the spectrum of the first peak becomes wider and wider, and hence more and more overlapping with the second and third peaks. In this case, results of a pre-estimation deteriorates. However, a further emphasis of the wanted peak can be made through an FIR or IIR bandpass filtering once the unwanted tail is removed by the HTLS$_F$ prefiltering. That is why the HTLS$_F$-$\mathbf{H}_{\text{FIR}t2}$-HTLS method gives the best resolution here when the unwanted peaks are not difficult to resolve.

Example 3: *Signal-5*

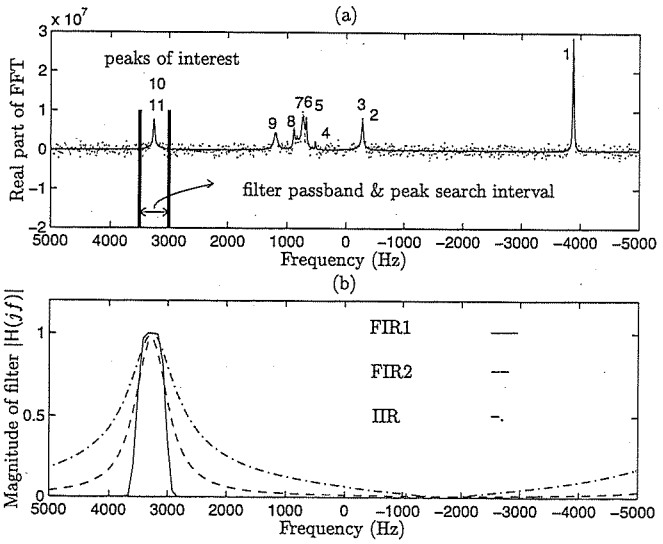


Figure 3.9: Example 3: (a) Noise-free (full line) and noisy (dotted line, $\sigma_\nu = 3 \times 10^4$) spectra of *Signal-5*. (b) The frequency response of the filters used in the prefiltered HTLS method.

Fig. 3.9 shows a complex simulated signal, called *Signal-5* and described in

Appendix A. It represents a typical signal encountered in NMR spectroscopy in which the glycogen peaks, 10 and 11 in this example, at the same known frequency are to be quantified [55]. The important but unknown parameters are the damping factor, which is the inverse of the relaxation time, and the amplitude representing the concentration of the metabolite, as explained in Chapter 1. Since only the glycogen peaks are of interest here and these peaks are not much overlapping with the remaining peaks, we apply several bandpass filters centered at the exact frequency of the glycogen peaks. The parameters for the prefiltered and unfiltered HTLS are: $N = 200, M = q = 121, f_l = 3000$ Hz, $f_u = 3500$ Hz, $\gamma = 0.2 f_s = 2000$ Hz. The region of wanted peaks, as well as the frequency interval in which the two glycogen peaks should be resolved, is defined as $[f_l, f_u]$. The estimated peak with a damping factor closest to d_{10} or d_{11} is classified as the estimate of peak 10 or 11.

An immediate observation of Fig. 3.10 is that the resolution of the prefiltered HTLS methods, $\mathbf{H}_{\text{FIR}t2}$-HTLS and HTLS$_F$-$\mathbf{H}_{\text{FIR}t2}$-HTLS, are much better than the HTLS and HTLS$_F$-HTLS methods. HTLS and HTLS$_F$-HTLS performs well only at very high SNR, e.g. at $\sigma_\nu = 5 \times 10^3$. They performs poorly at low SNR, e.g. at $\sigma_\nu = 3 \times 10^4$ and 5×10^4, since they can not resolve the two glycogen peaks in most of the noise trials while the other prefiltered HTLS methods are able to resolve these peaks in almost all trials. Although the HTLS$_F$ prefiltering does not lowers the failure rate of HTLS very much, it removes the unwanted peaks and their tails at the wanted region such that the HTLS$_F$ prefiltering followed by a bandpass prefiltering, like the HTLS$_F$-$\mathbf{H}_{\text{FIR}t2}$-HTLS in this figure, offers significant improvement. In parameter accuracy, $\mathbf{H}_{\text{FIR}t2}$-HTLS gives large RMSE values because the tails of the wanted peaks are suppressed by the stopband while the tails of the unwanted peaks are enlarged by the passband of the FIR filter. HTLS$_F$-$\mathbf{H}_{\text{FIR}t2}$-HTLS performs better due to the removal of unwanted peaks by HTLS$_F$ prefiltering. Note that the RMSE values of a_{10} and a_{11} at low SNR are lower than the CR bounds. This is due to the fact that prior knowledge of $f_{10} = f_{11} = 3250$ Hz has been used as the center frequency of the filter passband. However, the RMSE values of HTLS at low SNR are not reliable since they are averaged over the *good* runs only, excluding failures, and HTLS has very small numbers of good runs at low SNR.

Fig. 3.11 gives the performance of prefiltered HTLS methods versus the choice of M/N. The range of the optimal values of M/N is at least as large as that of the unfiltered HTLS, which is between 1/3 and 2/3 typically.

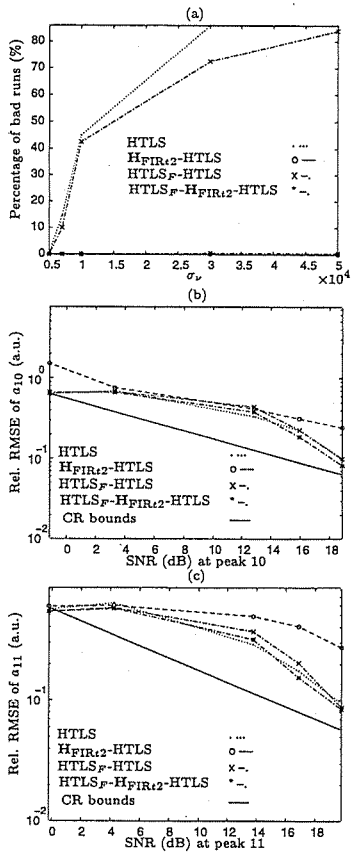


Figure 3.10: Example 3: (a) Failure rate versus σ_ν. Relative RMSE of (b) a_{10} and (c) a_{11} versus SNR at peak 10 and 11, respectively.

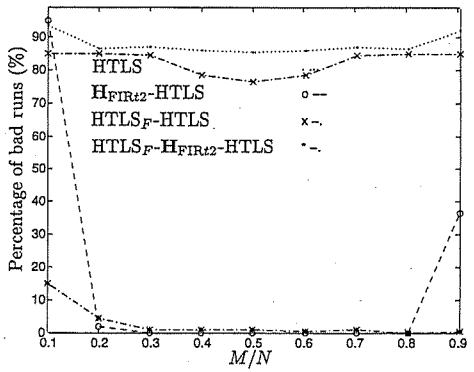


Figure 3.11: Example 3: Failure rate versus M/N. $\sigma_\nu = 3 \times 10^4$.

3.8 Conclusions

In estimating parameters of a sum of K exponentially damped sinusoids, two types of prefiltering techniques have been presented for the HTLS method. These techniques can be applied in two ways. First, when all the K sinusoids are of interest and the SNR's are low, a filter can be designed to encompass all the sinusoids and reduce the noise outside the passband. Secondly, when only some of the sinusoids (fewer than K) are of interest, a filter can be designed to encompass only the wanted peaks and suppress the unwanted ones and the noise.

The first type of prefiltering is realized by constructing a full-rank square filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ or $\mathbf{H}_{\mathrm{IIR}}$ from an FIR or IIR filter, and then multiplying it with the original Hankel data matrix before applying the HTLS estimation method. One of the characteristics of this prefiltering technique is that it does not increase the rank of the Hankel data matrix in the absence of noise. As a result, the computational work in the following HTLS estimation, as well as the variance of estimates, are not increased either. Appropriate application of the prefiltering technique not only reduces the noise but also suppresses the unwanted peaks outside the filter passband as demonstrated by means of a theoretical analysis.

The second type of prefiltering is the HTLS$_F$ prefiltering. It first invokes HTLS to get a pre-estimation of all K sinusoids comprising the original signal, then subtracts the signal that fits the unwanted peaks from the original one, and finally estimates the parameters of wanted peaks from the residual signal. It is usually applied when fewer than K sinusoids are of interest and the unwanted sinusoids are easily resolvable.

The benefits of the proposed filter matrices, $\mathbf{H}_{\text{FIR}t}$ and $\mathbf{H}_{\text{IIR}}$, are confirmed through simulations. When an FIR or IIR filter is appropriately designed, the resulting resolution is enhanced at comparable parameter accuracy, especially for low SNR. The parameter accuracy can be improved when the FIR or IIR filtering is preceded by the HTLS$_F$ prefiltering, that is, through the combination of the two types of prefiltering techniques.

It should be noted that the FIR and IIR prefiltering technique can be used prior to other SVD and subspace-based methods as well, such as the LP method, the MP method, Kung et al.'s method, etc.

Nevertheless, there still remain open problems with the proposed FIR and IIR prefiltering techniques. Recall that in Section 2.2 of Chapter 2, there is a requirement for the HTLS estimator that the noise covariance matrix be a scalar multiple of an identity matrix, or the noise be asymptotically white. However, when the noisy data matrix $\mathbf{X}$ is prefiltered, say, via $\bar{\mathbf{X}} = \mathbf{X}\mathbf{H}_r$, the filtered noise covariance matrix is generally not a scalar multiple of identity matrix anymore. This can be seen from

$$\bar{\mathbf{X}}^H\bar{\mathbf{X}} = \mathbf{H}_r^H \overset{\circ}{\mathbf{X}}^H \overset{\circ}{\mathbf{X}} \mathbf{H}_r + \mathbf{H}_r^H\mathbf{E}^H\mathbf{E}\mathbf{H}_r = \mathbf{H}_r^H \overset{\circ}{\mathbf{X}}^H \overset{\circ}{\mathbf{X}} \mathbf{H}_r + \mu^2\mathbf{H}_r^H\mathbf{H}_r$$

This problem cannot be solved by a whitening operation when the filter matrix $\mathbf{H}_r$ is *square*, which is the choice in this chapter. For example, the following whitening operation after a prefiltering operation using a square filter matrix $\mathbf{H}_r$ returns back to the HTLS algorithm, since

$$\mathbf{H}_r = \mathbf{Q}\mathbf{R}, \quad \bar{\mathbf{X}} = \mathbf{X}\mathbf{H}_r\mathbf{R}^{-1} = \mathbf{X}\mathbf{Q} = \mathbf{U}\boldsymbol{\Sigma}\bar{\mathbf{V}}$$

where $\mathbf{Q}$ is unitary and the left singular vectors of $\bar{\mathbf{X}}$ are exactly the same as those of $\mathbf{X}$.

Once the above problem is alleviated, e.g., as is the case of the HTLS(PK) method that incorporates prior knowledge of known signal poles (see Chapter 7, p. 145), there will be further improvement in resolution and parameter accuracy.

Chapter 4

Minimum variance signal enhancement

In practical signal processing, there are inevitably imperfections in the measurement data. However, parameter estimates of high resolution and accuracy are in demand. Therefore, it is often desired to apply a "cleaning" process, or what we call signal enhancement, to remove noise, measurement distortion, or even theoretical mismatch present in the given data set, before extracting parameters. A variety of signal enhancement algorithms have been developed for such purposes. These enhancement algorithms can be classified with respect to our objectives in mind:

- The enhanced signal itself must be "cleaned up" and smoothed.
- The estimated parameters of the enhanced signal must be closer to the true parameters of noise-free signal.

Though these two aspects are closely related, the latter is the emphasis of this study. In the current and succeeding chapters, we investigate, respectively, suboptimal and optimal signal enhancement techniques with a view to improving resolution and parameter accuracy, which is the main concern throughout this thesis.

This chapter is devoted to subspace-based signal enhancement methods in which the SVD plays a key role. These enhancement algorithms, though suboptimal, deliver good approximation of the original data matrix and good parameter estimates via the use of SVD. The success of the SVD in subspace methods can be justified from its special characteristics, such as the following three frequently cited features:

1. The information on the underlying data model is completely contained in certain subspaces of the signal space, which can be determined from the SVD.
2. The complexity of the model is given by the (approximate) rank of the data matrix, which can be estimated from the singular values.
3. In applications where the data are corrupted by additive noise, it is expected that the SVD has a certain noise filtering effect (also taking into account the relative insensitivity of the SVD with respect to perturbations).

Of course, the success of the SVD also depends on how it is applied in the algorithms. It is the main purpose of this chapter to clarify when and which SVD-based enhancement algorithm is most successful.

4.1 Introduction

Conceptually, a signal enhancement is accomplished through the process of slightly modifying the measured signal so that the modified signal possesses certain attributes or properties. The modification of the given data set serves as a preprocessing step of "cleaning up" the data before estimating the values of the signal parameters. Intuitively, this enhanced signal would provide a better representation of the underlying signal than the original measured signal.

In [56][57][58], Cadzow proposed an enhancement algorithm for the measured data represented in the form of a data matrix, such as the Hankel matrix $\mathbf{X}$ in (2.1). Frequently, the underlying data matrix is known to possess certain theoretical properties such as having a Hankel structure, a fixed rank, or a prescribed singular value characterization. The measurement process, however, usually results in a matrix which fails to possess all (or any) of these properties. Cadzow's method achieves the matrix property matching of Hankel structure and fixed rank via the use of the SVD. The application of Cadzow's enhancement procedure up to convergence followed by an SVD-based linear prediction algorithm has been performed on NMR spectroscopy data in [59][60]. This method works well for estimating nonoverlapping peaks. However, its high computational cost brings into question its practical use in NMR data analysis, and its resolution performance needs further improvement as well.

In the rest of this chapter, an improved signal-enhancement algorithm for exponential data quantification, presented in [61] and applied in [62] is considered and compared with Cadzow's algorithm. As outlined in Section 4.2, this approach is based on the minimum variance (MV) estimation method [29][61],

which first arranges the data in a very rectangular (instead of a square) Hankel structured matrix in order to make the corresponding signal-only data matrix orthogonal to the noise, then computes an estimate of the signal-only data matrix, and finally restores the Hankel structure of the computed estimate. Similar to Cadzow's method, little prior information (except model order) is required and the proposed method can be applied as a preprocessing procedure to a large class of subspace-based signal-estimation methods. Section 4.3 discusses the computational efficiency, choice of matrix sizes, etc., of the MV and Cadzow's enhancement. Simulation results are given in Section 4.4 to demonstrate a significant improvement in resolution performance at a comparable parameter accuracy and a low computational cost. Furthermore, the convergence of the MV and Cadzow's enhancement procedures is found to be unnecessary when followed by an SVD-based estimator such as HTLS, since the accuracy of the estimated parameters deteriorates despite a smoother spectrum, i.e., a "cleaner" spectrum does not guarantee better parameter accuracy. Finally, the proposed signal-enhancement algorithms are applied to the quantitative time-domain analysis of NMR measurement data in Section 3.5.

4.2 Enhancement algorithms

Although the enhancement algorithms discussed in this chapter also start from a Hankel matrix comprising the same data set as in subspace-based parameter estimation algorithms like HTLS, the size of the Hankel data matrix may differ. To distinguish the variables of preprocessing algorithms from that of parameter estimation algorithms, we use a caret, $\hat{}$, above the variables in the preprocessing algorithms. That is, the Hankel data matrix $\hat{\mathbf{X}}$ of dimension $\hat{L} \times \hat{M}$ can be generated by two unknown matrices: an exact data matrix $\hat{\hat{\mathbf{X}}}$ and an additive perturbation matrix $\hat{\mathbf{E}}$, so that

$$\hat{\mathbf{X}} = \hat{\hat{\mathbf{X}}} + \hat{\mathbf{E}}$$

Algorithm 4

Outline of the SVD-based data enhancement procedure

Given. *A model-order estimate K, N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, and tolerance ϵ.*

Step 1: Initialization. $\hat{\mathbf{x}} = [x_0, x_1, \ldots, x_{N-1}]$.

Step 2: Iteration. Repeat

(a) *Form an $\hat{L} \times \hat{M}$ Hankel matrix $\hat{\mathbf{X}}$ from the data vector $\hat{\mathbf{x}}$ with $\hat{L} \geq \hat{M} > K$ and $N = \hat{L} + \hat{M} - 1$.*

(b) *Compute the SVD of $\hat{\mathbf{X}} = \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^H$, where $\hat{\mathbf{\Sigma}} \in \mathbb{C}^{\hat{L}\times\hat{M}}$ contains the singular values $\hat{\sigma}_1, \ldots, \hat{\sigma}_{\hat{M}}$ with $\hat{\sigma}_1 \geq \ldots \geq \hat{\sigma}_{\hat{M}}$.*

(c) *Correct the singular values by applying a correction function, or rank-shaping function, f_{corr} that truncates to rank K:*

$$\hat{\mathbf{X}}_K = \hat{\mathbf{U}}_K f_{corr}(\hat{\mathbf{\Sigma}}_K)\hat{\mathbf{V}}_K^H$$

where $\hat{\mathbf{U}}_K \in \mathbb{C}^{\hat{L}\times K}, \hat{\mathbf{V}}_K \in \mathbb{C}^{\hat{M}\times K}$ are respectively the first K columns of $\hat{\mathbf{U}} \in \mathbb{C}^{\hat{L}\times\hat{L}}, \hat{\mathbf{V}} \in \mathbb{C}^{\hat{M}\times\hat{M}}$ and $\hat{\mathbf{\Sigma}}_K = diag(\hat{\sigma}_1, \cdots, \hat{\sigma}_K)$.

(d) *Set $\hat{\mathbf{x}}_{\text{old}} = \hat{\mathbf{x}}$. Compute the "cleaned-up" data vector $\hat{\mathbf{x}}$ from the rank-reduced data matrix $\hat{\mathbf{X}}_K$ by arithmetic averaging along its anti-diagonals.*

Until

$$\|\hat{\sigma}_{K+1}\| \leq \epsilon \tag{4.1}$$

$$or \quad \|\hat{\mathbf{x}}_{\text{old}} - \hat{\mathbf{x}}\| \leq \epsilon \tag{4.2}$$

After data preprocessing or signal enhancement outlined above, the signal parameters are extracted from the final "cleaned-up" data by means of an appropriate subspace-based signal-estimation algorithm. For example, the $4K$ signal parameters f_k, d_k, a_k and $\varphi_k, k = 1, 2, \ldots, K$ in the exponential data model (1.4) can be estimated using the LP-based method, the MP method, Kung et. al's method, or preferably, the HTLS method, Algorithm 1, outlined in Section 2.2.2 of Chapter 2. The procedure is then simply based on replacing the original noisy data $x_0, x_1, \ldots, x_{N-1}$ used in the $L \times M$ Hankel data matrix $\mathbf{X}$ by the "cleaned-up" data $\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{N-1}$.

In the following subsections, we describe two special forms of data preprocessing, the Cadzow and MV algorithms. The two algorithms differ only in the correction function f_{corr} in Step 2(c).

4.2.1 Cadzow's algorithm

Basically, Cadzow's algorithm modifies a given data set by exploiting two important properties of the data matrix $\hat{\mathbf{X}}$:

1. property of rank K
2. property of Hankel structure

The above preprocessing procedure reduces to *Cadzow's estimation method* for the following choice of f_{corr}

$$\boxed{f_{\text{corr}}(\hat{\mathbf{\Sigma}}_K) = \hat{\mathbf{\Sigma}}_K \quad \text{and} \quad \hat{L} = L,\ \hat{M} = M}$$

i.e. the singular values are not corrected and the Hankel matrix used in the preprocessing procedure has the *same* dimensions as the Hankel matrix used in the chosen signal-parameter-estimation method. For example, if HTLS is used, a square Hankel matrix $\hat{\mathbf{X}}_{\hat{L}\times\hat{M}}$ with $\hat{L} = \hat{M}(+1)$ is considered in the preprocessing procedure. Actually, the first two steps of a Cadzow's iteration are simply the LS estimation procedure given in the subsection 2.2.1.

4.2.2 Minimum variance algorithm

The MV estimation algorithm differs from the Cadzow algorithm in basically two aspects

- The MV algorithm starts from a very rectangular matrix, of which the number of rows is much larger than the number of columns, instead of a square matrix.
- The MV algorithm makes further use of the singular value property of a data matrix in addition to the property of rank K and the Hankel structure.

In terms of the above data preprocessing algorithm, the *MV estimation method* is based on the following choice of f_{corr} [29][61]

$$\boxed{\hat{L} >> \hat{M} > K \quad \text{and} \quad f_{\text{corr}}(\hat{\mathbf{\Sigma}}_K) = (\hat{\mathbf{\Sigma}}_K^2 - \hat{L}\nu_{\text{est}}^2\mathbf{I})\hat{\mathbf{\Sigma}}_K^{-1} \ \text{ with } \ \hat{\sigma}_K^2 > \hat{L}\nu_{\text{est}}^2} \tag{4.3}$$

where ν_{est}^2 is an estimate of the element-wise variance ν^2, which is equal to σ_ν^2 for real noise, or $2\sigma_\nu^2$ for complex noise (recall that in Section 2.3 of Chapter 2, σ_ν is defined as the noise standard deviation of real and complex part, respectively).

An MV iteration is based on the minimum variance estimation of $\hat{\overset{\circ}{\mathbf{X}}}$, or the orthogonal projection of $\hat{\overset{\circ}{\mathbf{X}}}$ onto the column space of $\hat{\mathbf{X}}$, represented as follows:

$$\min_{\hat{\mathbf{Y}}\in\mathbb{C}^{M\times M}} \|\hat{\mathbf{X}}\hat{\mathbf{Y}} - \hat{\overset{\circ}{\mathbf{X}}}\|_F^2$$

Setting to zero the derivatives of the object function

$$\|\hat{\mathbf{X}}\hat{\mathbf{Y}} - \hat{\overset{\circ}{\mathbf{X}}}\|_2^2 = \text{trace}(\hat{\mathbf{Y}}^H\hat{\mathbf{X}}^H\hat{\mathbf{X}}\hat{\mathbf{Y}} + \hat{\overset{\circ}{\mathbf{X}}}^H\hat{\overset{\circ}{\mathbf{X}}} - 2\hat{\mathbf{Y}}^H\hat{\mathbf{X}}^H\hat{\overset{\circ}{\mathbf{X}}})$$

with respect to the elements of $\hat{\mathbf{Y}}$ results in

$$\hat{\mathbf{Y}} = (\hat{\mathbf{X}}^H\hat{\mathbf{X}})^{-1}\hat{\mathbf{X}}^H\hat{\overset{\circ}{\mathbf{X}}}$$

so that the MV estimate of $\hat{\overset{\circ}{\mathbf{X}}}$ is given by

$$\begin{aligned}\hat{\mathbf{X}}_{\text{MV}} &= \hat{\mathbf{X}}\hat{\mathbf{Y}} \\ &= \hat{\mathbf{X}}(\hat{\mathbf{X}}^H\hat{\mathbf{X}})^{-1}\hat{\mathbf{X}}^H\hat{\overset{\circ}{\mathbf{X}}} \\ &= \hat{\mathbf{U}}_K(\hat{\mathbf{\Sigma}}_K^2 - \mu^2\mathbf{I}_K)\hat{\mathbf{\Sigma}}_K^{-1}\hat{\mathbf{V}}_K^H\end{aligned}$$

The last equality uses the SVD expression of $\hat{\overset{\circ}{\mathbf{X}}}$ and $\mathbf{X}$ in (2.2) and (2.5), as well as the canonical angles expression in (2.6), which are derived in Section 2.2.1 under three assumptions C1, C2, C3 and $\hat{\mathbf{E}}^H\hat{\mathbf{E}} = \mu^2\mathbf{I} = \hat{L}\nu^2\mathbf{I}$. In practice, these algebraic and geometrical assumptions may not be satisfied. But as the number of rows $\hat{L} \to \infty$, *asymptotically*, these conditions are satisfied [29] under statistical assumptions which are commonly observed in practice: the noise elements of $\hat{\mathbf{E}}$ have zero mean, bounded fourth order moments and are independently and identically distributed with possibly unknown variance ν^2. This is why $\hat{L} >> \hat{M}$ and white noise (such that $\hat{\mathbf{E}}^H\hat{\mathbf{E}} = \hat{L}\nu^2\mathbf{I}$) is required in the MV algorithm.

It should be noted that ν_{est}^2 is the estimated noise variance and can be known or computed either from the last data points provided they are pure noise (tail of the signal) or from the noise singular values, i.e. $\nu_{\text{est}}^2 = \frac{1}{(\hat{M}-K)\hat{L}}\sum_{i=K+1}^{\hat{M}}\hat{\sigma}_i^2$. Note that ν_{est} changes when we iterate the method. Hence, in the higher iterations ν_{est} is no longer known but can only be computed as explained above.

4.2.3 Composite property mapping

A detailed proof of the convergence of Cadzow's and MV enhancement algorithms is given in [56] and [61] in terms of composite property mapping. In essence, both algorithms make use of signal properties that identify the underlying useful information in the data set. The original problem is decomposed into several simpler subproblems related to each individual signal property, such as: $\hat{\mathbf{X}}$ has Hankel structure, $\hat{\mathbf{X}}$ has rank K, etc.. The basic requirement of this

simplified approach entails the ability to generate each property solution set for an arbitrary signal matrix, or each mapping operation that maps the signal matrix into the set of optimum signal approximations. Since the signal mappings are each closed in Cadzow's and MV algorithms, the convergence of such composite property mapping, implemented in the form of iterative processing, is assured.

4.3 Computational complexity

Based on the number of operations involved (see e.g., ref. [6] p. 239), it is easy to see that the computationally most time-consuming part of the algorithm, namely the SVD and the computation of the truncated matrix $\hat{\mathbf{X}}_K$, requires significantly less work whenever $\hat{L} >> \hat{M}$ [61]. Define by one flop the amount of work associated with one floating point operation (one addition, multiplication, division,...). If only third order terms are considered the number of flops involved in preprocessing a real $\hat{L} \times \hat{M}$ rank K matrix $\hat{\mathbf{X}}$ of the form (2.1) is given by:

$$\begin{aligned} & 6\hat{L}\hat{M}^2 + 20\hat{M}^3 + 2\hat{M}K\hat{L} \quad \text{if} \quad \hat{L} \geq 5\hat{M}/3 \\ & 14\hat{L}\hat{M}^2 + 8\hat{M}^3 + 2\hat{M}K\hat{L} \quad \text{if} \quad \hat{L} < 5\hat{M}/3 \\ \Rightarrow \quad & 22\hat{M}^3 + 2\hat{M}^2K \quad \text{if} \quad \hat{L} = \hat{M}\end{aligned}$$

Denote by $\hat{L}_{\text{CA}}$ ($\hat{M}_{\text{CA}}$) the number of rows (columns) as used in Cadzow's method, and by $\hat{L}_{\text{MV}}$ ($\hat{M}_{\text{MV}}$) the number of rows (columns) as used in the MV estimation method. We must have: $N = \hat{M}_{\text{CA}} + \hat{L}_{\text{CA}} - 1 = \hat{M}_{\text{MV}} + \hat{L}_{\text{MV}} - 1$, i.e. the number of data points is the same. If $\mathbf{X}$ in the HTLS estimation algorithm is approximately square ($L \approx M$) then also $\hat{L}_{\text{CA}} \approx \hat{M}_{\text{CA}}$. This means that an MV iteration requires roughly only

$$\frac{3\hat{L}_{\text{MV}}\hat{M}_{\text{MV}}^2 + 10\hat{M}_{\text{MV}}^3 + \hat{M}_{\text{MV}}K\hat{L}_{\text{MV}}}{7\hat{L}_{\text{CA}}\hat{M}_{\text{CA}}^2 + 4\hat{M}_{\text{CA}}^3 + \hat{M}_{\text{CA}}K\hat{L}_{\text{CA}}}100\% \tag{4.4}$$

of the number of flops performed by one Cadzow iteration in the preprocessing procedure. Since only the highest order terms are considered the expressions above only hold for large $\hat{L}$, $\hat{M}$. For complex matrices the same factor can be used (depending on the algorithm used for the complex SVD this ratio can alter a little bit). Although flop counting is a crude approach to measuring an algorithm's efficiency [6] it gives us nevertheless a good idea of the possible gain in computational efficiency.

For example, assume that $N = 25$ and $K = 2$ (as in [63, Example 2] or the simulated signal called *Signal-1* in Appendix A in this thesis). Choose

$\hat{M}_{\text{CA}} = \hat{L}_{\text{CA}} = 13$, $\hat{L}_{\text{MV}} = 22$ and $\hat{M}_{\text{MV}} = 4$. Then (4.4) implies that an MV iteration requires 8% of the computation time needed for a Cadzow iteration. In practical computations using Matlab, we obtained a ratio of 12.8% for real matrices and a ratio of 14.9% for complex matrices. Of course, since $N, \hat{M}, \hat{L}$ are small, (4.4) is too crude. If $N = 128$ and $K = 5$, as is the case of the simulated signal, called *Signal-2*, which will be studied extensively further on, then the work involved in preprocessing a 119×10 data matrix $\hat{\mathbf{X}}$ by each MV iteration should, according to (4.4), only be 2% of the work involved in processing a 65×64 matrix as done in a Cadzow's iteration. With Matlab, ratios of 3.6% and 4.4% were respectively obtained for real and complex matrices.

A similar saving can be obtained with the subsequent HTLS algorithm, provided that the preprocessing procedure has been applied up to convergence. This is due to the fact that upon convergence, the preprocessing procedure described above has reduced a full-rank Hankel matrix to one of exactly rank K. In this situation, we can simplify the subsequent parameter estimation as shown below.

- If the number of samples, N, is excessively large, (i.e. $N \gg 2K$), the number of data samples of the enhanced signal used in the pole-estimation algorithm can be reduced from N to as small as $2K-1$ without affecting the resulting estimates. Hence the Hankel data matrix can be reduced in size as much as possible to increase efficiency.

- Any SVD-based parameter estimation method using any matrix size $L \times M$ $(L, M \geq K)$ produces the same estimates after the enhancement. In fact, no SVD is required and the signal poles are simply the eigenvalues of the solution $\tilde{\mathbf{Z}}$ in $\mathbf{X}^{\uparrow} = \mathbf{X}_{\downarrow}\tilde{\mathbf{Z}}$. However, if the enhancement is not fully convergent, the most efficient and accurate estimator, HTLS, can be taken.

Of course, this saving is only possible when the preceding enhancement procedure was performed up to convergence. Such preprocessing may take hundreds of iterations depending on the convergence criteria used, which is defined in (4.1) and (4.2).

Another remark is that $\hat{M}$ should not be too small in the MV algorithm in order to guarantee a sufficiently large gap between the smallest signal singular value $\hat{\sigma}_K$ and the largest noise singular value $\hat{\sigma}_{K+1}$ of the $\hat{L} \times \hat{M}$ Hankel matrix $\hat{\mathbf{X}}$. As $\hat{M}$ decreases (i.e. $\hat{L}$ increases since $N = \hat{L} + \hat{M} - 1$ is constant and limited), the singular values of $\hat{\mathbf{X}}$ also decrease. The smallest signal singular value $\hat{\sigma}_K$ decreases quite strongly if $\hat{M} \approx K$, since $\hat{\mathbf{X}}$ contains fewer elements. This implies that the ratio $\hat{\sigma}_K/\hat{\sigma}_{K+1}$ decreases because the gap between the

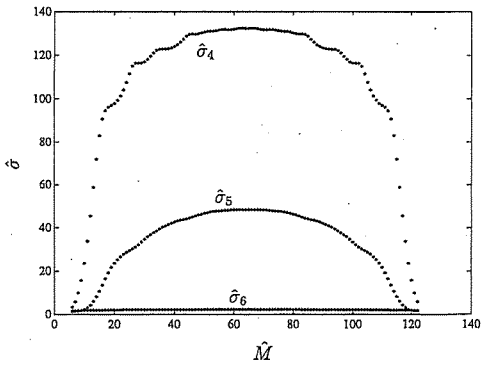


Figure 4.1: The singular values $\hat{\sigma}_{K-1}, \hat{\sigma}_K$ and $\hat{\sigma}_{K+1}$ (i.e., $\hat{\sigma}_4, \hat{\sigma}_5$ and $\hat{\sigma}_6$) of the $(129 - \hat{M}) \times \hat{M}$ Hankel matrix built up with 128 data points of a noisy $(\sigma_\nu = 0.1)$ simulated ^{31}P NMR signal, *Signal-2*, with model order $K = 5$.

smallest signal singular value $\hat{\sigma}_K$ and the largest noise singular value $\hat{\sigma}_{K+1}$ narrows. This can be observed in Fig. 4.1. The (fast) decrease of the smallest signal singular value of $\hat{\mathbf{X}}$ may have a deteriorating effect on the accuracy of the estimated signal subspace and subsequently affect the accuracy of the signal-parameter estimates. Referring to (2.6) and [29][61], a smaller gap means that the canonical angles between the exact signal subspace and the computed one enlarge, thereby increasing the bias of the estimates, especially those of the most difficult peak (smallest in amplitude, overlapping, ...), as can be seen in the simulation study given further on. On the other hand, $\hat{M}$ should not be large since then the signal-only data cannot be orthogonal to the noise, implying that the MV estimation properties no longer hold. For that reason, we choose $\hat{M}$ between $K + 5$ and $2K$.

Finally, the condition that the noise should be white is not always a restriction. If the noise is not white then $\hat{\mathbf{E}}^H\hat{\mathbf{E}} = \mu^2\mathbf{I}$ is no longer valid so that the singular value correction function is no longer correct. However if $\hat{\mathbf{E}}^H\hat{\mathbf{E}}$ is known or can be estimated up to an unknown scalar, then we can factor $\hat{\mathbf{E}}^H\hat{\mathbf{E}} = \hat{\mathbf{R}}^H\hat{\mathbf{R}}$ (or $\hat{\mathbf{E}} = \hat{\mathbf{Q}}\hat{\mathbf{R}}$, $\hat{\mathbf{Q}}^H\hat{\mathbf{Q}} = \mathbf{I}$ if $\hat{\mathbf{E}}$ is directly available) where

$\hat{\mathbf{R}} \in \mathbb{C}^{\tilde{M}\times\tilde{M}}$ is a nonsingular square root. In this case, we consider the matrix

$$\tilde{\mathbf{X}} = \hat{\mathbf{X}}\hat{\mathbf{R}}^{-1} = \overset{\circ}{\mathbf{X}}\hat{\mathbf{R}}^{-1} + \hat{\mathbf{E}}\hat{\mathbf{R}}^{-1}$$

Observe that $(\hat{\mathbf{E}}\hat{\mathbf{R}}^{-1})^H(\hat{\mathbf{E}}\hat{\mathbf{R}}^{-1})$ is again a multiple of the identity matrix so that we can again apply the same procedure, i.e. compute the SVD of $\tilde{\mathbf{X}}$, truncate to rank K and correct the singular values (according to (4.3) in order to obtain $\tilde{\mathbf{X}}_K$. The only modification is that $\tilde{\mathbf{X}}_K$ should be post-multiplied with $\hat{\mathbf{R}}$, i.e. $\hat{\mathbf{X}}_K = \tilde{\mathbf{X}}_K\hat{\mathbf{R}}$, before restoring the Hankel structure.

The fact that $\hat{\mathbf{E}}^H\hat{\mathbf{E}}$ or an estimate should be known up to a scalar is not always a trivial requirement. In some cases, this matrix can be estimated during the absence of the signal. For instance, when modeling damped exponential signals as is the case in NMR spectroscopy, $\hat{\mathbf{E}}$ can be computed from the tail of the signal where the exact signal has died out.

4.4 Simulation results

In this section, we illustrate, via a computer simulation study, the differences in resolution enhancement that may occur from the use of the different signal-enhancement procedures when the complex data are fitted to *Signal-2*, described in Fig. A.2 and Tables A.3, A.4 in Appendix A. This signal, modeled by a fifth order exponential function defined by Eq. (1.4), represents a ^{31}P NMR signal from a perfused rat liver.

The signal-parameter-estimation algorithm considered here is HTLS as outlined in the previous section, where the Hankel matrix is as square as possible, i.e. $L \times M = 64 \times 65$. The number of data points used in each fit is $N = 128$ (corresponding to the signal disappearing in the noise in the time domain), the matrix sizes used in the MV method and Cadzow's method are, respectively, 119×10 and 64×65, and the estimated model order is $K = 5$. If no preprocessing is performed, the method is called HTLS. If the MV method, Cadzow's method, or MV followed by Cadzow's method is used as signal-enhancement algorithm prior to HTLS, the method is abbreviated in this Chapter, respectively, as nMV-HTLS, nCA-HTLS or MV-nCA-HTLS, where n indicates the number of iterations performed; n is deleted if only one iteration is performed and n is replaced by c if the method has been used up to convergence. The convergence criterion used is (4.1) where the tolerance is set to $\epsilon = 10^{-6}$. The maximum number of iterations is set to 100.

Various scenarios have been simulated in our Monte Carlo study to compare the different signal-enhancement algorithms. First, the performance of these

peak k	method	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k(°)
1	HTLS	-1374	249	6.9	10
	CA-HTLS	-1371	262	7.0	8
	MV-HTLS	-1374	242	6.1	10
	CR bound	5	30	0.6	6
2	HTLS	-682	247	10.6	10
	CA-HTLS	-681	246	10.6	11
	MV-HTLS	-683	236	9.7	12
	CR bound	4	26	0.7	4
3	HTLS	-263	205	6.2	10
	CA-HTLS	-266	194	6.0	10
	MV-HTLS	-265	185	5.2	11
	CR bound	4	31	0.7	6
4	HTLS	391	710	19.0	28
	CA-HTLS	380	722	18.7	31
	MV-HTLS	357	157	3.2	-1
	CR bound	10	61	1.1	22
5	HTLS	4337	21	0.7	-52
	CA-HTLS	4342	-1	0.8	-59
	MV-HTLS	463	843	16.7	25
	CR bound	19	117	2.0	7

Table 4.1: Parameter estimates and CR lower bound of the noisy ($\sigma_\nu = 2$, see Fig. 4.2) simulated ^{31}P NMR signal, *Signal-2*, using the HTLS, CA-HTLS and MV-HTLS algorithms.

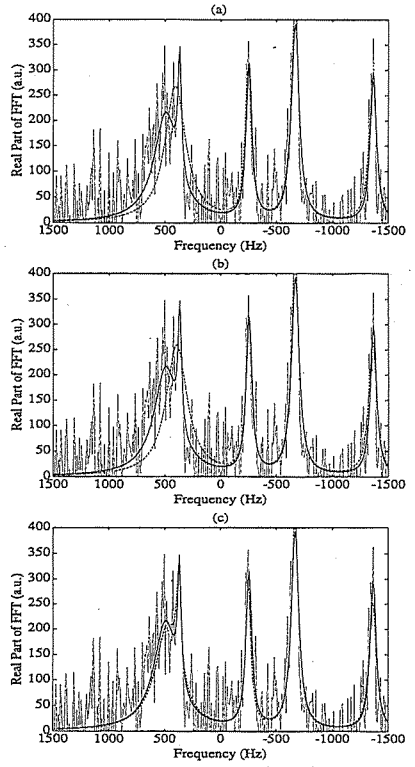


Figure 4.2: Real part of the FFT of the reconstructed simulated ^{31}P NMR signal, *Signal-2*, (dashed line) as estimated by (a) HTLS, (b) CA-HTLS, and (c) MV-HTLS from the same set of noisy data ($\sigma_\nu = 2$, dotted line). The full line represents the exact signal.

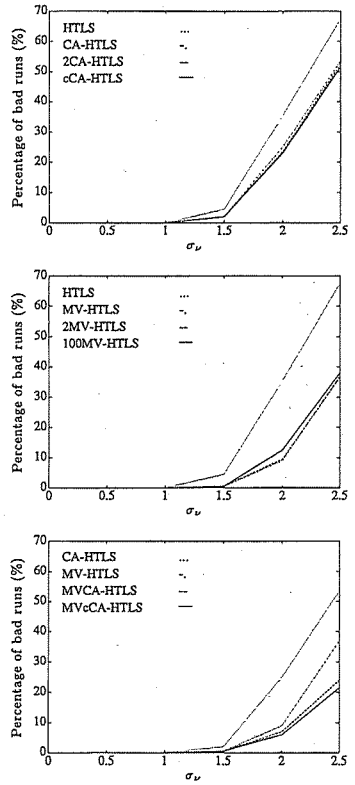


Figure 4.3: Failure rate of HTLS, nCA-HTLS, nMV-HTLS, and MV-nCA-HTLS versus noise standard deviation. Here the MV method does not converge within 100 iterations.

algorithms after one iteration is compared and next the performance upon convergence is considered.

Fig. 4.2 and Table A.3 indicate that peaks 4 and 5 of the signal are two difficult peaks with closely spaced frequencies. Peak 4 lies on the flank of peak 5 in the noise-free case. However, in the presence of white noise, say $\sigma_\nu = 2$, these two peaks are often indistinguishable (dotted line in Fig. 4.2). The detection of these peaks in the presence of noise has been studied using three different algorithms HTLS, CA-HTLS and MV-HTLS, as shown in Fig. 4.2 and Table 4.1. Apparently, only MV-HTLS is able to resolve the closely spaced peaks 4 and 5 in this case. This is also clear from the failure rate plots in Fig. 4.3. These plots indicate that both the MV and Cadzow's enhancement procedures using one iteration improve the resolution significantly. However, the enhanced frequency resolution is better for the MV estimation method than for Cadzow's method. All detected failures in resolution are due to the fact that one of the two overlapping peaks, especially peak 4, cannot be resolved. Furthermore, the occurrence of a failure of MV-HTLS is mostly due to the fact that the estimated peak 4 (or 5) falls outside although close to the specified frequency interval. This is in contrast to Cadzow's method, CA-HTLS, which has more failures. If a failure occurs, peaks 4 and 5 are estimated by only one peak, as illustrated in Fig. 4.2 and Table 4.1, and an additional noise peak appears far away from the exact frequency, often around $\pm$4000Hz. Note also that applying an MV iteration followed by a Cadzow iteration prior to HTLS further reduces the failure rate.

The differences in parameter accuracy between the MV-HTLS and the CA-HTLS methods are small, especially for the estimates of well-resolved peaks 1, 2 and 3. For overlapping peaks 4 and 5, the accuracy of the damping factors and amplitudes of the two methods are comparable as shown in Fig. 4.4 but the frequencies and phases (not shown) are estimated somewhat more accurately by CA-HTLS (maybe due to a smaller bias). Note also that these results are based on average values over all *good* runs. This number differs from one method to the other. It certainly reflects the fact that MV-HTLS is still able to compute accurate estimates of all parameters during the runs in which CA-HTLS fails to resolve peak 4 or 5. The slight loss in accuracy, as discussed earlier in this section, is mainly due to the occurrence of a smaller singular value gap $\hat{\sigma}_K/\hat{\sigma}_{K+1}$ in the $\hat{L} \times \hat{M}$ data matrix $\hat{\mathbf{X}}$ used by MV-HTLS compared to the one used by CA-HTLS as shown in Fig. 4.1 and Table 4.2. The increase in the bias of all parameter estimates due to the smaller gap can be observed from Table 4.3. It is to be noted that applying MV-CA-HTLS further improves the accuracy of the estimates wherever CA-HTLS performs better than MV-HTLS.

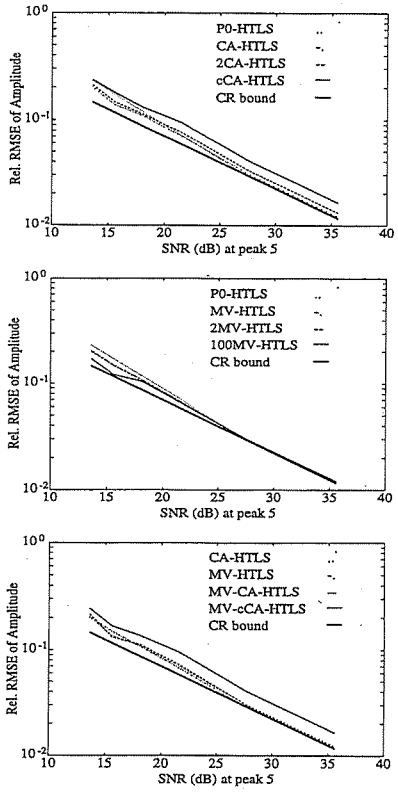


Figure 4.4: Relative RMSE values of a_5 (averaged over good runs) as estimated by HTLS, nCA-HTLS, nMV-HTLS, and MV-nCA-HTLS. The thick full line is the CR bound. $\sigma_\nu = 2$ corresponds to SNR=15.58 dB at peak 5.

method	singular values					
	1st	2nd	3rd	4th	5th	6th
0MV	144.02	122.94	90.75	31.63	24.44	23.99
1MV	139.54	118.97	86.04	24.60	5.75	4.32
2MV	139.39	118.84	85.87	24.30	4.54	1.96
3MV	139.38	118.83	85.85	24.24	4.29	1.63
20MV	139.30	118.77	85.67	23.91	3.42	8.96e-01
500MV	138.99	118.52	84.99	23.26	2.40	2.69e-01
0CA	192.99	152.97	139.79	134.78	49.28	30.31
1CA	192.67	152.43	139.91	134.74	49.05	4.57
2CA	192.58	152.27	139.98	134.82	49.22	2.67
3CA	192.52	152.18	140.02	134.90	49.29	1.75
20CA	192.36	152.03	140.10	135.14	49.39	6.46e-03
51CA	192.36	152.02	140.10	135.14	49.39	3.31e-07
1MV-1CA	187.32	146.52	133.83	130.19	47.25	4.24
1MV-2CA	187.24	146.36	133.90	130.27	47.40	2.55
1MV-20CA	187.05	146.11	134.04	130.55	47.58	8.63e-03
1MV-52CA	187.05	146.11	134.04	130.55	47.58	3.18e-07

Table 4.2: The first $K+1(=6)$ singular values of the noisy ($\sigma_\nu = 2$, see Fig. 4.2) simulated ^{31}P NMR signal, *Signal-2*, processed by different enhancement procedures.

MV-HTLS requires the estimation of the noise variance ν^2_{est} in order to correct the singular values of the Hankel data matrix. For one MV iteration, no differences in accuracy or resolution have been observed between the use of the exact noise variance and the use of one of the estimates, either based on the tail of the signal or based on the average of the squared noise singular values.

When the signal is *very noisy*, it often happens that noise peaks are identified instead of the relevant peaks we are interested in. Overestimating the model order (i.e., taking the order ≥ 5 in our example) at low SNR further reduces the failure rate of both MV-HTLS and CA-HTLS methods, as illustrated in Fig. 4.5, while the differences in failure rate remain relatively the same. However, if K is chosen too large, the accuracy of the computed parameters does not improve and is mostly slightly worse probably due to a smaller accuracy of the estimates of the difficult runs (in which the non-overestimated method fails).

If we increase the number of iterations performed during preprocessing,

peak k/Method	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k (°)
peak 1				
HTLS	-0.28±6.35	-2.98±36.6	0.03±0.72	0.25±7.61
CA-HTLS	-0.24±7.35	5.16±52.0	0.19±0.92	0.27±8.69
MV-HTLS	2.58±7.10	-3.02±46.5	-0.83±0.78	-2.60±7.42
MV-CA-HTLS	0.69±7.44	2.11±49.0	-0.76±0.83	-0.60±8.31
peak 2				
HTLS	0.13±5.91	5.47±35.8	0.14±0.92	-0.04±5.89
CA-HTLS	0.44±6.02	1.34±34.6	0.06±0.92	-0.03±6.11
MV-HTLS	0.74±5.95	4.37±33.6	-0.57±0.85	-0.69±5.65
MV-CA-HTLS	0.17±5.81	0.73±34.4	-0.66±0.89	0.09±5.70
peak 3				
HTLS	-0.26±7.56	1.36±41.5	0.16±0.84	-0.42±9.56
CA-HTLS	0.13±6.98	3.18±42.1	0.23±0.86	-1.12±9.56
MV-HTLS	-1.34±6.92	3.45±38.2	-0.37±0.76	0.43±9.09
MV-CA-HTLS	-0.75±6.81	3.75±40.3	-0.44±0.77	0.16±9.04
peak 4				
HTLS	2.61±17.1	20.1±108.	1.28±2.48	-3.69±35.4
CA-HTLS	0.87±15.1	-9.46±90.9	0.55±1.84	-1.85±36.2
MV-HTLS	-2.05±15.7	-20.6±102.	0.04±2.29	0.75±37.9
MV-CA-HTLS	0.26±14.8	-1.84±107.	0.34±2.01	-1.88±34.9
peak 5				
HTLS	10.7±28.4	-88.3±232.	-1.01±3.81	0.71±9.55
CA-HTLS	0.82±26.9	-57.4±179.	-0.25±3.41	1.02±8.73
MV-HTLS	-15.8±25.7	-69.8±161.	-1.18±3.28	4.83±9.57
MV-CA-HTLS	-6.20±28.3	-43.3±177.	-1.46±3.38	3.02±8.83

Table 4.3: Bias±standard deviation values of all parameter estimates of the noisy ($\sigma_\nu = 2.5$) simulated ^{31}P NMR signal, *Signal-2* (see Fig. 4.3 through Fig. 4.4).

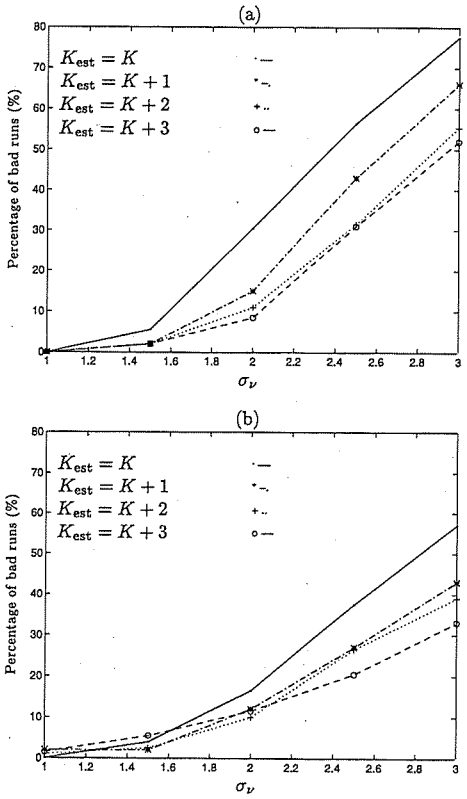


Figure 4.5: Failure rate of (a) CA-HTLS and (b) MV-HTLS versus σ_ν for different estimated model orders K_{est}.

the following conclusions can be drawn. As shown in Fig. 4.3, the failure rate decreases in the first few iterations, after which it hardly changes. To be more specific, the first iteration is the most decisive one while the procedure applied up to convergence hardly improves the resolution. Inspection of the spectra plotted in Fig. 4.6 reveals that the "cleaning" effect of the enhancement procedures becomes smaller as the number of iterations increases. The typical number of iterations required by the different methods is listed in Table 4.4. Although the MV method needs only 4.4% of the computation time of Cadzow's method for each iteration, it needs a much larger number of iterations to achieve the same degree of smoothness in the spectrum. A similar increase in the number of required iterations is needed when Cadzow's method is applied to a rectangular (instead of a square) data matrix. Although the computation time for one iteration reduces significantly, the number of iterations needed for convergence increases at such a rate that the overall computation time is even larger than the time needed to apply Cadzow's method to a square data matrix up to convergence (in contrast to the assertion in [60]). Tests have also been performed with the MV method using square (instead of long) Hankel matrices, and the required number of iterations is comparable to that required by Cadzow's method. *Hence, it is the size of the Hankel matrix used and not the applied singular-value correction function which is crucial for a quick convergence.* This is due to the fact that the square matrix has a larger gap between the Kth and $(K+1)$th singular values, as shown in Table 4.2, and averages over more elements when restoring the Hankel structure.

Furthermore, when the number of iterations in the enhancement procedure increases, the accuracy of the estimated parameters as illustrated in Fig. 4.4 remains the same or deteriorates. One reason might be that *the MV and Cadzow's enhancement procedures converge to a suboptimal instead of the optimal solution as pointed out in [29][64], i.e., these procedures converge to a "clean" rank-K Hankel matrix but not to the one closest to the original noisy data matrix.* See Chapter 5 for a comparison of these suboptimal enhancement methods with optimal ones. Another more important reason is that the MV and Cadzow's iterations introduce colored noise into the preprocessed data. As pointed out in [65], the SVD truncation and element averaging along the cross diagonals in each iteration is equivalent to subtraction of a zero-phase filtered version of the original signal from itself. Such a filtering operation masks the noise at the K signal peaks, and thus turns white noise into colored one. The colored noise causes the deviation of the convergent "clean" signal peaks from the exact ones (bias). Therefore, a "cleaner" spectrum does not guarantee better accuracy. *The enormous computational cost and the unsatisfactory results lead to the conclusion that it is not necessary to apply the MV and Cadzow's*

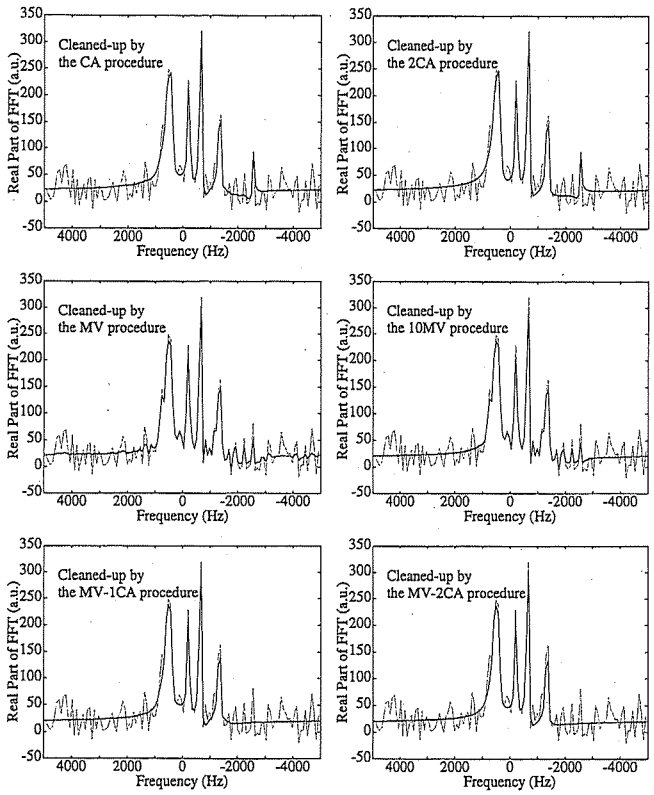


Figure 4.6: "Cleaned-up" spectra (full line) of the same set of noisy data ($\sigma_\nu = 2$, dotted line) using the nMV, nCA and MV-nCA enhancement procedures.

enhancement procedures up to convergence.

enhancement	for visually "clean" spectrum	for convergence
nMV	$n \approx 10$	$n \geq 1000$
nCA	$n \approx 1$ or 2	$n \approx 47$
MV-nCA	$1+n$ and $n \approx$(1 or 2)	$1+n$ and $n \approx 48$

Table 4.4: Typical numbers of iterations required by different enhancement procedures for the simulated noisy ($\sigma_\nu = 2$) ^{31}P NMR signal, *Signal-2*, with model order $K = 5$.

In summary, the MV-HTLS method is the best choice for estimating parameters of signals with overlapping exponentially damped peaks at low computational cost, when the SNR is low and no prior knowledge is available. The model order can be chosen equal to or somewhat larger than the exact one and a matrix size of $K + 5 \leq \hat{M} \leq 2K$ is suggested for the MV enhancement procedure.

4.5 Application to real-world NMR signals

The application of the enhancement algorithms discussed above to real-world NMR signals is described in this section. The measurement data are assumed to be modeled by the function defined by Eq. (1.4) and perturbed by additive white noise. When $t_0 \neq 0$ and $\phi^{(0)} \neq 0$, the signal parameters estimated by HTLS must be extrapolated to the time origin and the phases must be corrected.

Fig. 4.7 shows the signal to be processed, which is a ^{31}P NMR FID signal from a perfused rat liver. It was obtained in a 4.7 Tesla, 30 cm wide-bore magnet, equipped with a Biospec spectrometer, and acquired at 81.1 MHz. Two hundred fifty-six scans were accumulated using a 60 μs (72°) pulse and a 0.69 s repetition time. The peaks of interest are PDE (phospho-diesters), P_i (inorganic phosphate), and PME (phospho-monoesters) which together form a cluster in the spectrum. The peak at the highest frequency is an external standard with known concentration, which allows absolute concentration estimation from the computed amplitudes (extrapolated to the time origin) of the peaks. The zero-order phase $\phi^{(0)} = \varphi^{(0)} * 180/\pi$ is 23°, determined from the spectrometer phase settings, and the dead time is 0 s. In our fit, 150 data points are taken, starting from the sixth point (x_6) on (the first 5 are deleted in

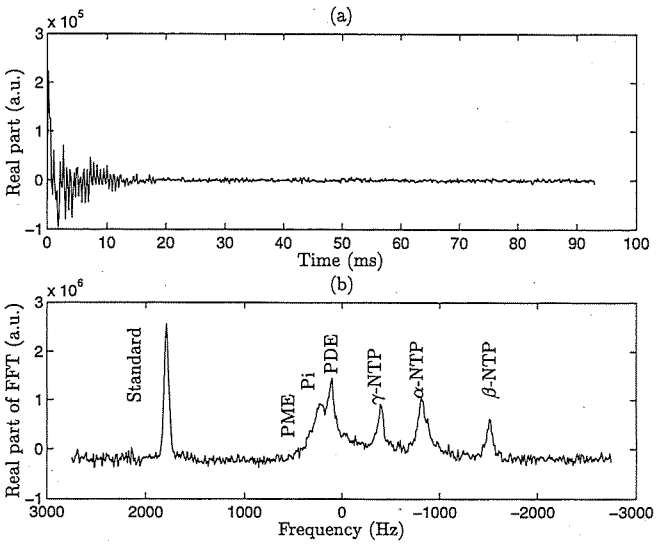


Figure 4.7: (a) ^{31}P-NMR FID signal of perfused rat liver. (b) The corresponding NMR spectrum (after omission of the first five data points).

order to eliminate the broad hump that appears in the original spectrum). The chosen model order, $K = 15$, is slightly overestimated in order to obtain estimates of all three PDE-P_i-PME peaks, among which the smallest PME peak is rather difficult to resolve. The size of the Hankel data matrix is 129×22 in the MV procedure and 76×75 in Cadzow's procedure. The ν_{est} used in the MV method is estimated from the noise singular values.

Table 4.5 presents the extrapolated and phase-corrected parameter estimates of the three overlapping peaks PDE-P_i-PME using the methods HTLS, CA-HTLS and MV-HTLS respectively. CR bounds are provided to indicate the accuracy of our estimates. Based on the physical reality that phases should be close to zero after extrapolation and phase correction, the MV-HTLS method produces the best fit of the three individual peaks. In addition, it provides the

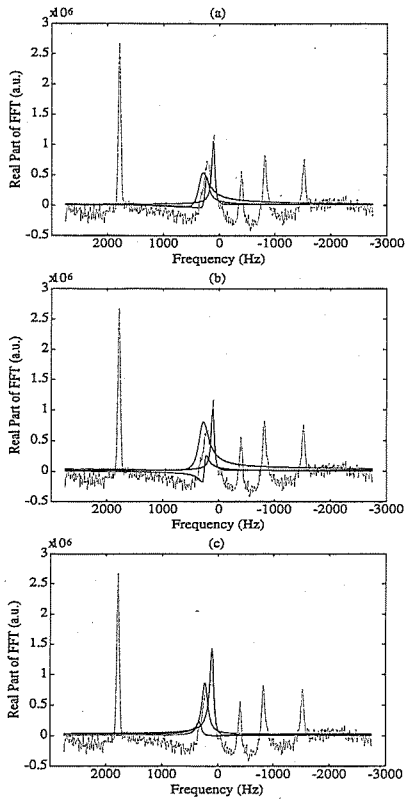


Figure 4.8: Spectra of the exponentially damped sinusoidal components that reconstruct the three overlapping peaks PDE-P_i-PME of the real-world NMR signal, as computed by (a) HTLS (b) CA-HTLS and (c) MV-HTLS.

physically most relevant damping factors. The other two methods, HTLS and CA-HTLS, underestimate the damping factor of the P_i peak and overestimate the damping factor of the PME peak. These can be visualized by drawing the Fourier transform of the exponentially damped sinusoidal components that fit each peak separately as shown in Fig. 4.8. It can be clearly seen that the "P_i" and "PME" peaks obtained by HTLS and CA-HTLS do not bear physical meaning anymore although the overall fits (i.e., the combination of all peaks) by all three methods mentioned above have a similar, small residue. The parameter estimates obtained using cCA-HTLS, cMV-HTLS, MV-CA-HTLS and MV-cCA-HTLS are even worse and are not included here. We therefore conclude that MV-HTLS is the best method for this case.

peak k	method	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k (°)
PDE	HTLS	103± 2	197± 12	37716± 2475	3± 4
	CA-HTLS	103± 2	191± 12	35276± 2382	2± 4
	MV-HTLS	95±11	266± 67	69178±29303	10±24
P_i	HTLS	236± 9	257± 54	24324± 8668	-44±20
	CA-HTLS	249 ±12	232± 75	18487±11958	-85±37
	MV-HTLS	225 ±12	397± 77	60455±20266	-4±19
PME	HTLS	322±13	673± 79	69440±11021	-44± 9
	CA-HTLS	292±13	681± 80	97950±14810	-29± 9
	MV-HTLS	318±23	325±147	13522±10757	13±46

[a] computed from the parameter estimates.

Table 4.5: Real-world NMR signal-parameter estimates and their CR bounds[a] corresponding to the three exponentially damped sinusoidal components which fit the peaks PDE-P_i-PME.

4.6 Conclusions

A new enhancement algorithm based on minimum variance estimation for fitting a sum of damped exponential signals embedded in white noise has been described. Comparison of the MV method and Cadzow's method is made, based on extensive simulations and real-world NMR signals. When nonoverlapping peaks are considered and one iteration of the enhancement procedure is used, both methods are comparable in resolution and parameter accuracy but the MV method is much more efficient. However, when overlapping peaks are

to be quantified, then an MV iteration doubles the resolution independently of the noise variance at comparable parameter accuracy, while the computation time is less than 10% of the time needed to execute Cadzow's iteration, depending on the number of data points and the model order. Considering both resolution and accuracy, one or two iterations of the enhancement procedure are sufficient while the convergence of the procedure is not necessary, offering drastic reduction in the computational cost, which is valuable in practical NMR signal processing. Furthermore, overestimating the model order is helpful in quantifying some difficult peaks.

Similar to the problem of prefiltering techniques in the previous chapter, however, the MV and Cadzow enhancement procedures introduce colored noise into the preprocessed data. This is because truncating the SVD and averaging the elements along the cross diagonals in each iteration is equivalent to subtracting a zero-phase filtered version of the original signal from itself as proven in [65]. Such a filtering operation masks the noise at the K signal peaks, and thus turns white noise into colored noise. New procedures that minimize the effects of the colored noise are being developed (see [66]).

Chapter 5

Structured TLN and structured TLS in signal enhancement

In SVD and subspace-based parameter estimation algorithms, the rank-K matrix, computed from the "signal" subspace of the original Hankel data matrix, does not preserve the Hankel structure anymore. This problem has been handled in the previous chapter where signal-enhancement algorithms at convergence recover both the Hankel structure and the rank-K property of data matrices through composite property mapping. Such preprocessing procedures result in enhanced data matrices of which the elements better represent the underlying signal components than did the original data. With the development of numerical techniques, new algorithms for the problem of approximating structured rank-deficient matrices have emerged. As extensions of the TLS technique, the constrained TLS (CTLS) [67][43], structured TLS (STLS) [68][64] and structured total least norm (STLN) techniques [69][70][71] have been proposed one after another. Recently, CTLS and STLS have been proven to be equivalent [72]. Hence, we will focus only on the application of STLN and STLS to signal enhancement of our parameter estimation problem in this chapter.

The subsequent sections are organized as follows. Section 5.1 formulates the STLN and STLS problem. Section 5.2 gives solutions and outlines of the STLN and STLS algorithms for complex Hankel/Toeplitz structured data matrices. Section 5.3 presents a more accurate parameter estimator, HTLN, for

our problem of exponential data fitting, which starts from the same Hankel data matrix as in HTLS but uses the STLN technique prior to the HTLS parameter estimation method. In Section 5.4, comparisons of the STLN (HTLN), STLS algorithms with the SVD-based suboptimal procedures are made using complex exponential data in terms of computational efficiency and accuracy.

5.1 Formulation of STLN and STLS

Let

$$[\mathbf{A}_{L\times K} \;\; \mathbf{B}_{L\times(M-K)}] = \mathbf{X}, \qquad L \geq K, M \geq K+1, N = L+M-1$$

where $\mathbf{A}$ consists of the first K columns and $\mathbf{B}$ consists of the remaining $M-K$ columns of $\mathbf{X}$, which is the Hankel data matrix defined in (2.1). Since the rank of the matrix $\mathbf{X} = [\mathbf{A} \;\; \mathbf{B}]$ is K when no noise is added, there exists a $K \times (M-K)$ matrix $\mathbf{Y}$ such that $\mathbf{AY} = \mathbf{B}$. If noise is added, the corresponding set of linear equations $\mathbf{AY} \approx \mathbf{B}$ is no longer compatible, since $[\mathbf{A} \;\; \mathbf{B}]$ is typically of full rank. The rank-K property of $[\mathbf{A} \;\; \mathbf{B}]$ can be restored by computing appropriate corrections $[\Delta\mathbf{A} \;\; \Delta\mathbf{B}]$ such that $[\mathbf{A}+\Delta\mathbf{A} \;\; \mathbf{B}+\Delta\mathbf{B}]$ has rank K, i.e., $(\mathbf{A}+\Delta\mathbf{A})\mathbf{Y} = \mathbf{B}+\Delta\mathbf{B}$ is exactly solvable.

TLS problem

The well-known TLS technique for solving $\mathbf{AY} \approx \mathbf{B}$ is formulated as

$$\min_{\Delta\mathbf{A},\Delta\mathbf{B},\mathbf{Y}} \|[\Delta\mathbf{A} \;\; \Delta\mathbf{B}]\|_2^2 \quad \text{subject to } (\mathbf{A}+\Delta\mathbf{A})\mathbf{Y} = \mathbf{B}+\Delta\mathbf{B}$$

This method allows to modify both matrices $\mathbf{A}$ and $\mathbf{B}$ by adding correction matrices $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$, respectively, and is justified when $\mathbf{A}$, as well as $\mathbf{B}$, are affected by noise. The computational method generally used for solving TLS is based on the SVD of $[\mathbf{A} \;\; \mathbf{B}]$ [6]. But in applications where the matrix $\mathbf{A}$ or $[\mathbf{A} \;\; \mathbf{B}]$ has a special structure like Hankel, Toeplitz, etc., the SVD-based methods may not always be appropriate since they do not preserve the special structure. In fact, the matrix $[\Delta\mathbf{A} \;\; \Delta\mathbf{B}]$ obtained from the SVD will typically be dense, with no structure, even when $\mathbf{A}$ or $[\mathbf{A} \;\; \mathbf{B}]$ is structured.

STLN problem

The above problem can be alleviated by the newly developed technique, STLN, proposed by Rosen, Park and Glick [69] and extended to multiple right-hand sides by Van Huffel, Park and Rosen [71]. The STLN technique preserves

any given affine structure in $\Delta\mathbf{A}$ or $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$, such as Hankel, Toeplitz, or sparse, and can minimize these corrections in the discrete L_p norm, where $p = 1, 2$ or ∞. It can be formulated as

$$\min_{\Delta\mathbf{A},\Delta\mathbf{B},\mathbf{Y}} \quad \|[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]\|_p^2$$
$$\text{subject to} \quad \begin{cases} (\mathbf{A}+\Delta\mathbf{A})\mathbf{Y} = (\mathbf{B}+\Delta\mathbf{B}) \\ [\Delta\mathbf{A}\ \ \Delta\mathbf{B}] \text{ preserves the given structure} \end{cases}$$
$$p = 1, 2, \infty$$

where $\Delta\mathbf{A}$ and $\Delta\mathbf{B}$ are correction matrices with the given special structure.

This formulation implies that STLN for $p = 2$ reduces to TLS if all elements of $[\mathbf{A}\ \mathbf{B}]$ are subject to error and the structure is not taken into account.

In our application, *only the STLN algorithm for L_2 norm is considered, that is, $p = 2$.* In particular, when $\mathbf{B}$ is only a column vector $\mathbf{B} = \mathbf{b}$ and $\mathbf{X} = [\mathbf{A}\ \mathbf{b}]$ is Hankel/Toeplitz structured, the STLN problem which approximates $\mathbf{A}\mathbf{y} \approx \mathbf{b}$ by the closest compatible set $(\mathbf{A}+\Delta\mathbf{A})\mathbf{y} = (\mathbf{b}+\Delta\mathbf{b})$ of the same structure can be formulated as

$$\min_{\alpha,\beta,\mathbf{y}} \left\| \mathbf{D} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \right\|_2^2 \quad \text{subject to} \quad \mathbf{r}(\alpha,\beta,\mathbf{y}) = \mathbf{0} \tag{5.1}$$

where $\alpha \in \mathbb{C}^{q_\alpha \times 1}$, $\beta \in \mathbb{C}^{q_\beta \times 1}$ are the correction vectors representing, respectively, the q_α, q_β *different* elements of the correction matrix $\Delta\mathbf{A}$, $\Delta\mathbf{b}$ subject to error; $\mathbf{D} \in \mathbb{R}^{(q_\alpha+q_\beta)\times(q_\alpha+q_\beta)}$ is a diagonal matrix with positive weights that takes into account the multiplicity of the elements of α, β in $[\Delta\mathbf{A}\ \ \Delta\mathbf{b}]$; and $\mathbf{r}(\alpha,\beta,\mathbf{y}) = (\mathbf{b}+\Delta\mathbf{b}) - (\mathbf{A}+\Delta\mathbf{A})\mathbf{y}$ is the modified residual vector of the set of linear equations. This formulation of the STLN problem for $p = 2$ is derived in the next section. Actually, it is an optimization problem with quadratic objective function and nonlinear constraints.

STLS problem

When $\mathbf{B}$ has only one column, $\mathbf{B} = \mathbf{b}$, and $\mathbf{X} = [\mathbf{A}\ \mathbf{b}]$ is affinely structured, the STLS technique proposed by De Moor [68][64] can also be used after being extended from real data to complex data. STLS attempts to approximate affinely structured matrices by similarly structured rank-deficient ones, while minimizing an L_2-error criterion, which approximates a given data sequence $\mathbf{x} = [x_0, x_1, \ldots, x_{N-1}]^T$ by $\hat{\mathbf{x}} = [\hat{x}_0, \hat{x}_1, \ldots, \hat{x}_{N-1}]^T$ such that the (weighted) sum of squares of residuals is minimized.

Let $\mathbf{W} = \text{diag}(w_0, w_1, \ldots, w_{N-1}) \in \mathbb{R}^{N\times N}$ be a diagonal weighting matrix. Choose N fixed matrices $\mathsf{X}_n \in \mathbb{R}^{L\times M}, n = 0, 1, \ldots, N-1$, representing the structure of $[\mathbf{A}\ \mathbf{b}]$ so that an affinely structured matrix can be written as an affine matrix function of its components: $\mathbf{X} = \Sigma_{n=0}^{N-1} x_n \mathsf{X}_n$ and $\hat{\mathbf{X}} = \Sigma_{n=0}^{N-1} \hat{x}_n \mathsf{X}_n$. Then STLS is formulated as

$$\min_{\hat{\mathbf{x}}\in\mathbb{C}^{N\times 1}, \mathbf{y}\in\mathbb{C}^{M\times 1}} \Sigma_{n=0}^{N-1} |x_n - \hat{x}_n|^2 w_n \quad \text{subject to} \quad \begin{cases} \hat{\mathbf{X}}\mathbf{y} = \mathbf{0} \\ \mathbf{y}^H\mathbf{y} = 1 \end{cases} \tag{5.2}$$

Remark that the structure of $\hat{\mathbf{X}}$ is forced to be the same as the structure of $\mathbf{X}$ since $\hat{\mathbf{X}}$ is written as an affine combination of the same X_n as $\mathbf{X}$. This formulation implies that STLS reduces to TLS if no structure of the matrix $\mathbf{X} = [\mathbf{A}\ \mathbf{b}]$ or weighting of its elements is taken into account.

5.2 STLN and STLS algorithms applied to complex Hankel/Toeplitz data matrices

In what follows, we solve the STLN and STLS problem and outline the STLN, using the 2-norm ($p = 2$), and STLS algorithms for data matrices having complex elements and a Hankel structure. The results also hold for Toeplitz matrices, since Toeplitz matrices simply transform to Hankel matrices by permutations. Algorithms for other structures and norms are not treated here and can be found in [69][68].

5.2.1 The STLN algorithms applied to complex Hankel data matrices

We consider two STLN algorithms according to the dimension of the complex Hankel data matrix:

1. $[\mathbf{A}_{L\times K}\ \mathbf{b}_{L\times 1}]$ has $M = K+1$ columns, which corresponds to the single-right-hand side problem $\mathbf{A}\mathbf{y} \approx \mathbf{b}$.
2. $[\mathbf{A}_{L\times K}\ \mathbf{B}_{L\times(M-K)}]$ has $M > K+1$ columns, which corresponds to the multiple-right-hand side problem $\mathbf{A}\mathbf{Y} \approx \mathbf{B}$.

STLN for Hankel structured $[\mathbf{A}_{L\times K}\ \mathbf{b}_{L\times 1}]$

The main trick of STLN in approximating $\mathbf{A}\mathbf{y} \approx \mathbf{b}$ by $(\mathbf{A}+\Delta\mathbf{A})\mathbf{y} = (\mathbf{b}+\Delta\mathbf{b})$ is to use the correction vectors α and β which represent *different* non-zero elements of the correction matrices $\Delta\mathbf{A}$ and $\Delta\mathbf{b}$. Here, the vector α, β and the matrix $\Delta\mathbf{A}, \Delta\mathbf{b}$ are equivalent in the sense that α, β is known when

$\Delta\mathbf{A}, \Delta\mathbf{b}$ is given and vice versa. In this way, the STLN problem turns into an optimization problem (5.1) over the vectors $\alpha, \beta, \mathbf{y}$, based on the following relationship between $\Delta\mathbf{A}$ and α, $\Delta\mathbf{b}$ and β, as shown below.

- To represent the vector $\Delta\mathbf{A}\mathbf{y}$ in terms of α, we define a matrix $\check{\mathbf{Y}} \in \mathbb{C}^{L \times q_\alpha}$ such that $\Delta\mathbf{A}\mathbf{y} = \check{\mathbf{Y}}\alpha$. The construction of $\check{\mathbf{Y}}$ (starting with a zero matrix) is carried out according to the following rule:
  If α_k is the $(i,j)^{th}$ element of $\Delta\mathbf{A}$, then y_j is the $(i,k)^{th}$ element of $\check{\mathbf{Y}}$, where $i = 1, \ldots, L$, $j = 1, \ldots, K$, and $k = 1, \ldots, q_\alpha$.
  The matrix $\check{\mathbf{Y}}$ consists of the elements of $\mathbf{y}$, with suitable repetition, giving $\check{\mathbf{Y}}$ a special structure. In fact, for Hankel structured $\Delta\mathbf{A}$, we have

$$\begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_K \\ \alpha_2 & \alpha_3 & \cdots & \alpha_{K+1} \\ \vdots & \vdots & & \vdots \\ \alpha_L & \alpha_{L+1} & \cdots & \alpha_{q_\alpha} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_K \end{bmatrix}$$

$$= \begin{bmatrix} y_1 & y_2 & \cdots & y_K & 0 & \cdots & 0 \\ 0 & y_1 & y_2 & \cdots & y_K & \ddots & \vdots \\ \vdots & \ddots & y_1 & y_2 & \cdots & y_K & 0 \\ 0 & \cdots & 0 & y_1 & y_2 & \cdots & y_K \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{q_\alpha} \end{bmatrix} \tag{5.3}$$

- Since all the elements of $\mathbf{b}$ are subject to error in our case, we have $\beta = \Delta\mathbf{b}$, with $q_\beta = L$.
- In Hankel structured $[\mathbf{A}\ \mathbf{b}]$, the elements of $\mathbf{b}$ are linearly related with the elements of $\mathbf{A}$. It follows that β is also linearly related with α such that

$$\beta = \mathbf{P}_\beta \beta_0 + \mathbf{P}_\alpha \alpha,$$

  where $\mathbf{P}_\beta = [0, \cdots, 0, 1]^T$, , $\mathbf{P}_\alpha = \begin{bmatrix} \mathbf{0}_{(L-1)\times K} & \mathbf{I}_{(L-1)\times(L-1)} \\ \mathbf{0}_{1\times K} & 0_{1\times(L-1)} \end{bmatrix}$, and $\beta_0 \in \mathbb{C}^{1\times 1}$.
- In error minimization, α and β_0 should be weighted by diagonal weighting matrices $\mathbf{D}_\alpha$ and $\mathbf{D}_\beta$, respectively, that account for the repetition of the elements of α and β_0 in $[\Delta\mathbf{A}\ \ \Delta\mathbf{b}]$. For our Hankel structured matrix $[\Delta\mathbf{A}\ \ \Delta\mathbf{b}]$, $\begin{bmatrix} \mathbf{D}_\alpha^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta^2 \end{bmatrix} = \mathrm{diag}(1, 2, \cdots, q, \cdots, q, \cdots, 2, 1)$, $q = \min(L, K+1)$, and the object function in (5.1) can be written as

$$f(\alpha, \beta_0) = \| \begin{bmatrix} \mathbf{D}_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta \end{bmatrix} \begin{bmatrix} \alpha \\ \beta_0 \end{bmatrix} \|_2^2 = \alpha^H \mathbf{D}_\alpha^2 \alpha + \beta_0^H \mathbf{D}_\beta^2 \beta_0. \tag{5.4}$$

- The modified residual vector in the nonlinear constraint of the optimization problem in (5.1) is a function of $\alpha, \beta, \mathbf{y}$ and can be written as

$$\begin{aligned} \mathbf{r}(\alpha, \beta, \mathbf{y}) &= (\mathbf{b} + \Delta\mathbf{b}) - (\mathbf{A} + \Delta\mathbf{A})\mathbf{y} \\ &= \mathbf{b} + \beta - \mathbf{A}\mathbf{y} - \check{\mathbf{Y}}\alpha \end{aligned}$$

  which is nonlinear in the parameters $\alpha, \mathbf{y}$ due to the last term on the right-hand side. The nonlinearity can also be seen from

$$\begin{aligned} \mathbf{r} & (\alpha + \Delta\alpha, \beta + \Delta\beta, \mathbf{y} + \Delta\mathbf{y}) \\ &= \mathbf{b} + (\beta + \Delta\beta) - \mathbf{A}(\mathbf{y} + \Delta\mathbf{y}) - (\check{\mathbf{Y}} + \Delta\check{\mathbf{Y}})(\alpha + \Delta\alpha) \\ &= (\mathbf{b} + \beta - \mathbf{A}\mathbf{y} - \check{\mathbf{Y}}\alpha) + \Delta\beta - \mathbf{A}\Delta\mathbf{y} - \check{\mathbf{Y}}\Delta\alpha - \Delta\check{\mathbf{Y}}\alpha \\ &\quad -\Delta\check{\mathbf{Y}}\Delta\alpha \\ &= \mathbf{r}(\alpha, \beta, \mathbf{y}) - \check{\mathbf{Y}}\Delta\alpha + \Delta\beta - (\mathbf{A} + \Delta\mathbf{A})\Delta\mathbf{y} - \Delta\check{\mathbf{Y}}\Delta\alpha \\ &= \mathbf{r}(\alpha, \beta_0, \mathbf{y}) - (\check{\mathbf{Y}} - \mathbf{P}_\alpha)\Delta\alpha + \mathbf{P}_\beta\Delta\beta_0 - (\mathbf{A} + \Delta\mathbf{A})\Delta\mathbf{y} \\ &\quad -\Delta\check{\mathbf{Y}}\Delta\alpha \end{aligned} \tag{5.5}$$

Several solution methods to solve our problem (5.1) with the constraints given in (5.5) have been presented up till now. The accuracy and convergence rate of the STLN algorithm depends on which method is used to solve this optimization problem. In the sequel, two methods labeled as STLN Algorithm-1 and STLN Algorithm-2, respectively, are introduced and outlined.

STLN Algorithm-1

STLN Algorithm-1 is the weighting method described in [69][70][71]. It corresponds to the penalty function method in optimization theory [73][74]. Let $\mathbf{z} = \begin{bmatrix} \alpha \\ \beta_0 \\ \mathbf{y} \end{bmatrix}$ and define a penalty function by weighting the constraint violations $\mathbf{r}(\mathbf{z})^H\mathbf{r}(\mathbf{z})$, and adding it to the objective function in (5.4).

$$\begin{aligned} P(\mathbf{z}) &= f(\mathbf{z}) + w^2\mathbf{r}(\mathbf{z})^H\mathbf{r}(\mathbf{z}) \\ &= \alpha^H\mathbf{D}_\alpha^2\alpha + \beta_0^H\mathbf{D}_\beta^2\beta_0 + w^2\mathbf{r}(\alpha, \beta_0, \mathbf{y})^H\mathbf{r}(\alpha, \beta_0, \mathbf{y}) \end{aligned} \tag{5.6}$$

The constrained optimization problem is then reduced to an unconstrained optimization problem:

$$\min_{\mathbf{z}} P(\mathbf{z}) \quad \text{or} \quad \min_{\alpha,\beta_0,\mathbf{y}} \left\| \begin{bmatrix} w\mathbf{r}(\alpha,\beta_0,\mathbf{y}) \\ \mathbf{D}_\alpha\alpha \\ \mathbf{D}_\beta\beta_0 \end{bmatrix} \right\|_2^2 \tag{5.7}$$

w is a non-negative weight, which has to be taken large for the unconstrained and the constrained problem to be equivalent. It can be shown that, under mild conditions, the solution of (5.7) converges to the solution of (5.1) when w approaches infinity [74]. However, w cannot be taken too large due to the numerical problems, since the condition number of the Hessian matrix of the penalty function $P(\mathbf{z})$ increases as w becomes larger, with singularity occurring in the limit. The problems in choosing a suitable weight w makes automatic implementation of this algorithm difficult.

In the following outline of STLN Algorithm-1, $\mathbf{r}(\alpha,\beta_0,\mathbf{y})$ is linearized by omitting the second order terms $-\Delta\check{\mathbf{Y}}\Delta\alpha$ in (5.5):

$$\begin{aligned} \mathbf{r} \quad & (\mathbf{z}+\Delta\mathbf{z}) \\ = \quad & \mathbf{r}(\alpha+\Delta\alpha,\beta_0+\Delta\beta_0,\mathbf{y}+\Delta\mathbf{y}) \\ \approx \quad & \mathbf{r}(\alpha,\beta_0,\mathbf{y}) - (\check{\mathbf{Y}}-\mathbf{P}_\alpha)\Delta\alpha + \mathbf{P}_\beta\Delta\beta_0 - (\mathbf{A}+\Delta\mathbf{A})\Delta\mathbf{y} \\ = \quad & \mathbf{r}(\mathbf{z}) + [\mathbf{P}_\alpha - \check{\mathbf{Y}} \quad \mathbf{P}_\beta \quad -(\mathbf{A}+\Delta\mathbf{A})]\,\Delta\mathbf{z} \end{aligned} \tag{5.8}$$

so that our problem transforms to a sequence of LS minimization subproblems.

Algorithm 5

Outline of the STLN Algorithm-1 for Hankel structured $[\mathbf{A}_{L\times K} \ \mathbf{b}_{L\times 1}]$

Given. $\mathbf{A}, \mathbf{b}, \mathbf{D}_\alpha, \mathbf{D}_\beta$, *Hankel structure of* $[\mathbf{A}\ \mathbf{b}]$, *tolerance* ϵ.

Step 1: Initialization. *Choose a large number* w. *Set* $\alpha = \mathbf{0}$, $\beta_0 = \mathbf{0}$, $\Delta\mathbf{A} = \mathbf{0}$, *compute the LS solution* $\mathbf{y}$ *from* $\min_{\mathbf{y}} \|\mathbf{A}\mathbf{y} - \mathbf{b}\|_2$, $\check{\mathbf{Y}}$ *from* $\mathbf{y}$ *according to (5.3), and* $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{y}$.

Step 2: Iteration. *Repeat*

(a) Solve the following LS problem

$$\min_{\Delta\alpha,\Delta\beta_0,\Delta\mathbf{y}} \left\| \begin{bmatrix} w(\mathbf{P}_\alpha - \check{\mathbf{Y}}) & w\mathbf{P}_\beta & -w(\mathbf{A}+\Delta\mathbf{A}) \\ \mathbf{D}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0} \end{bmatrix} \right.$$

$$\left. \begin{bmatrix} \Delta\alpha \\ \Delta\beta_0 \\ \Delta\mathbf{y} \end{bmatrix} + \begin{bmatrix} w\mathbf{r} \\ \mathbf{D}_\alpha\alpha \\ \mathbf{D}_\beta\beta_0 \end{bmatrix} \right\|_2^2$$

(b) $\alpha := \alpha + \Delta\alpha, \beta_0 := \beta_0 + \Delta\beta_0, \mathbf{y} := \mathbf{y} + \Delta\mathbf{y}$

(c) Construct $\Delta\mathbf{A}$ *from* α, $\Delta\mathbf{b}$ *from* α *and* β_0, $\check{\mathbf{Y}}$ *from* $\mathbf{y}$. *Compute* $\mathbf{r} = (\mathbf{b}+\Delta\mathbf{b}) - (\mathbf{A}+\Delta\mathbf{A})\mathbf{y}$.

Until $(\|\Delta\alpha\|, \|\Delta\beta_0\|, \|\Delta\mathbf{y}\|, \|\mathbf{r}\| \le \epsilon)$ (5.9)

STLN Algorithm-2

The STLN Algorithm-2 used in [45] is different from the STLN Algorithm-1 only in Step 2. It is also based on linearizing the nonlinear constraints of our optimization problem (5.1) as done in (5.8). Unlike the weighting method in STLN Algorithm-1, however, which reduces the equality constrained problem to an unconstrained one, STLN Algorithm-2 solves the resulting sequence of LS subproblems with linear equality constraints (LSE) of the following form using the generalized SVD (GSVD) [75] :

$$\min_{\Delta\mathbf{z}} \left\|\hat{\mathbf{A}}\Delta\mathbf{z} - \hat{\mathbf{b}}\right\|_2^2 \quad \text{subject to} \quad \hat{\mathbf{B}}\Delta\mathbf{z} - \hat{\mathbf{d}} = \mathbf{0} \tag{5.10}$$

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{D}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0} \end{bmatrix} \in \mathbb{C}^{\hat{m}\times\hat{n}}, \ \Delta\mathbf{z} = \begin{bmatrix} \Delta\alpha \\ \Delta\beta_0 \\ \Delta\mathbf{y} \end{bmatrix} \in \mathbb{C}^{\hat{n}\times 1},$$

$$\hat{\mathbf{B}} = \begin{bmatrix} \mathbf{P}_\alpha - \check{\mathbf{Y}} & \mathbf{P}_\beta & -(\mathbf{A}+\Delta\mathbf{A}) \end{bmatrix} \in \mathbb{C}^{\hat{p}\times\hat{n}},$$

$$\hat{\mathbf{b}} = -\begin{bmatrix} \mathbf{D}_\alpha\alpha \\ \mathbf{D}_\beta\beta_0 \end{bmatrix} \in \mathbb{C}^{\hat{m}\times 1}, \ \hat{\mathbf{d}} = -\mathbf{r} \in \mathbb{C}^{\hat{p}\times 1},$$

$$\hat{m} = N = L+K, \ \hat{n} = L+2K, \ \hat{p} = L, \ \text{and} \ \hat{p} < \hat{m} < \hat{n}.$$

The basic idea is to perform a GSVD of $\hat{\mathbf{A}}$ and $\hat{\mathbf{B}}$, such that

$$\hat{\mathbf{A}} = \mathbf{U}_A\mathbf{D}_A\mathbf{Z}_{AB}, \quad \hat{\mathbf{B}} = \mathbf{U}_B\mathbf{D}_B\mathbf{Z}_{AB} \tag{5.11}$$

where $\mathbf{U}_A \in \mathbb{C}^{\hat{m}\times\hat{m}}$ and $\mathbf{U}_B \in \mathbb{C}^{\hat{p}\times\hat{p}}$ are unitary, $\mathbf{Z}_{AB} \in \mathbb{C}^{\hat{n}\times\hat{n}}$ is a full-rank matrix with the same singular values as those of $\begin{bmatrix} \hat{\mathbf{A}} \\ \hat{\mathbf{B}} \end{bmatrix}$. $\mathbf{D}_A = [\mathbf{P}_A, \ \mathbf{0}] \in \mathbb{R}^{\hat{m}\times\hat{n}}$, $\mathbf{P}_A = \text{diag}(\rho_{A1}, \rho_{A2}, \ldots, \rho_{A\hat{m}})$, $1 \geq \rho_{A1} \geq \rho_{A2} \geq \ldots \geq \rho_{A\hat{m}} > 0$, and $\mathbf{D}_B = [\mathbf{0}, \ \mathbf{P}_B] \in \mathbb{R}^{\hat{p}\times\hat{n}}$, $\mathbf{P}_B = \text{diag}(\rho_{B1}, \rho_{B2}, \ldots, \rho_{B\hat{p}})$. $0 < \rho_{B1} \leq \rho_{B2} \leq \ldots \leq \rho_{B\hat{p}} < 1$. In this way, the LSE problem of (5.10) is simplified to:

$$\min_{\Delta\check{\mathbf{z}}} \ \|\mathbf{D}_A\Delta\check{\mathbf{z}} - \check{\mathbf{b}}\|_2^2 \quad \text{subject to} \quad \mathbf{D}_B\Delta\check{\mathbf{z}} = \check{\mathbf{d}} \tag{5.12}$$

where $\Delta\check{\mathbf{z}} = \mathbf{Z}_{AB}\Delta\mathbf{z}$, $\check{\mathbf{b}} = \mathbf{U}_A^H\hat{\mathbf{b}}$, and $\check{\mathbf{d}} = \mathbf{U}_B^H\hat{\mathbf{d}}$.

The special diagonal structure of $\mathbf{D}_A$ and $\mathbf{D}_B$ enables an easy computation of the elements $\Delta\check{z}_i$, $i = 1, 2, \ldots, \hat{n}$ of $\Delta\check{\mathbf{z}}$:

$$\Delta\check{z}_i = \begin{cases} \check{b}_i/\rho_{Ai}, & i = 1, 2, \ldots, (\hat{n} - \hat{p}) \\ \check{d}_{(i+\hat{p}-\hat{n})}/\rho_{B(i+\hat{p}-\hat{n})}, & i = \hat{n} - \hat{p} + 1, \hat{n} - \hat{p} + 2, \ldots, \hat{n} \end{cases} \tag{5.13}$$

Once $\Delta\check{\mathbf{z}}$ is obtained, $\Delta\mathbf{z}$ can be found from $\Delta\mathbf{z} = \mathbf{Z}_{AB}^{-1}\Delta\check{\mathbf{z}}$.

Algorithm 6

Outline of the STLN Algorithm-2 for Hankel structured $[\mathbf{A}_{L\times K} \ \ \mathbf{b}_{L\times 1}]$

Given. $\mathbf{A}, \mathbf{b}, \mathbf{D}_\alpha, \mathbf{D}_\beta$, *Hankel structure of* $[\mathbf{A} \ \mathbf{b}]$, *tolerance* ϵ.

Step 1: Initialization. *Set* $\alpha = \mathbf{0}$, $\beta_0 = \mathbf{0}$, $\Delta\mathbf{A} = \mathbf{0}$, *compute the LS solution* $\mathbf{y}$ *from* $\min_{\mathbf{y}}\|\mathbf{A}\mathbf{y} - \mathbf{b}\|_2$, $\check{\mathbf{Y}}$ *from* $\mathbf{y}$ *according to (5.3), and* $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{y}$.

Step 2: Iteration. *Repeat*

(a) $\hat{\mathbf{A}} := \begin{bmatrix} \mathbf{D}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0} \end{bmatrix}$, $\Delta\mathbf{z} := \begin{bmatrix} \Delta\alpha \\ \Delta\beta_0 \\ \Delta\mathbf{y} \end{bmatrix}$, $\hat{\mathbf{b}} := -\begin{bmatrix} \mathbf{D}_\alpha\alpha \\ \mathbf{D}_\beta\beta_0 \end{bmatrix}$,

$\hat{\mathbf{B}} := \begin{bmatrix} (\mathbf{P}_\alpha - \check{\mathbf{Y}}) & \mathbf{P}_\beta & -(\mathbf{A} + \Delta\mathbf{A}) \end{bmatrix}$, $\hat{\mathbf{d}} = -\mathbf{r}$.

(b) Use the GSVD method to solve the following LSE problem

$$\min_{\Delta\mathbf{z}} \ \left\|\hat{\mathbf{A}}\Delta\mathbf{z} - \hat{\mathbf{b}}\right\|_2^2 \quad \textit{subject to} \quad \hat{\mathbf{B}}\Delta\mathbf{z} - \hat{\mathbf{d}} = \mathbf{0}$$

(c) $\alpha := \alpha + \Delta\alpha, \beta_0 := \beta_0 + \Delta\beta_0, \mathbf{y} := \mathbf{y} + \Delta\mathbf{y}$

(d) Construct $\Delta\mathbf{A}$ *from* α, $\Delta\mathbf{b}$ *from* α *and* β_0, $\check{\mathbf{Y}}$ *from* $\mathbf{y}$. *Compute* $\mathbf{r} = (\mathbf{b} + \Delta\mathbf{b}) - (\mathbf{A} + \Delta\mathbf{A})\mathbf{y}$.

Until $(\|\Delta\alpha\|, \|\Delta\beta_0\|, \|\Delta\mathbf{y}\|, \|\mathbf{r}\| \leq \epsilon)$ (5.14)

STLN for structured $[\mathbf{A}_{L\times K} \ \mathbf{B}_{L\times(M-K)}]$

The key idea of generalizing the one right-hand-side STLN problem to a multiple-right-hand-side one is to vectorize $\mathbf{A}_{L\times K}\mathbf{Y}_{K\times(M-K)} \approx \mathbf{B}_{L\times(M-K)}$ with $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_{(M-K)}], \mathbf{y}_i \in \mathbb{C}^{K\times 1}$ and $\mathbf{B} = [\mathbf{b}_1, \cdots, \mathbf{b}_{(M-K)}], \mathbf{b}_i \in \mathbb{C}^{L\times 1}$:

$$\begin{bmatrix} \mathbf{A} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{A} & \cdots & \mathbf{0} \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{A} \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_{(M-K)} \end{bmatrix} \approx \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_{(M-K)} \end{bmatrix}$$

$$\text{or} \quad \vec{\mathbf{A}}\vec{\mathbf{y}} \approx \vec{\mathbf{b}} \tag{5.15}$$

As detailed in [71], the multiple-right-hand-side STLN problem is then reduced to a one-right-hand-side STLN problem formulated below

$$\min_{\alpha,\beta,\vec{\mathbf{y}}} \ \left\| \begin{bmatrix} \mathbf{D}_\alpha\alpha \\ \mathbf{D}_\beta\beta \end{bmatrix} \right\|_2^2, \quad \text{subject to} \quad \vec{\mathbf{r}}(\alpha, \beta, \vec{\mathbf{y}}) = \mathbf{0}$$

where $\vec{\mathbf{r}}(\alpha, \beta, \vec{\mathbf{y}}) = (\vec{\mathbf{b}} + \Delta\vec{\mathbf{b}}) - (\vec{\mathbf{A}} + \Delta\vec{\mathbf{A}})\vec{\mathbf{y}}$ is the modified residual vector of the set of vectorized linear equations. Hence, the same algorithms described above can be applied.

5.2.2 The STLS algorithm applied to complex Hankel data matrices

The problem of approximating a matrix $\mathbf{X}$, consisting of data x_n, by a reduced-rank matrix $\hat{\mathbf{X}}$, consisting of data $\hat{x}_n$ and with the same structure as $\mathbf{X}$, can also be solved using the following theorem.

Theorem 4 *For a complex data matrix* $\mathbf{X}$, *the solution of the STLS optimization problem (5.2) can be obtained from the triplet* $(\mathbf{u}, \tau, \mathbf{v})$ *of the Riemannian SVD defined below:*

Find the triplet $(\mathbf{u}, \tau, \mathbf{v})$ *corresponding to the minimal* τ *that satisfies*

$$\begin{cases} \mathbf{X}\mathbf{v} = \mathbf{D}_v\mathbf{u}\tau & \mathbf{u}^H\mathbf{D}_v\mathbf{u} = 1 \\ \mathbf{X}^H\mathbf{u} = \mathbf{D}_u\mathbf{v}\tau & \mathbf{v}^H\mathbf{D}_u\mathbf{v} = 1 \end{cases} \tag{5.16}$$

where

- *The elements of the desired rank-reduced matrix* $\hat{\mathbf{X}}$ *are obtained from* $\hat{x}_n = x_n - (\mathbf{u}^H\mathbf{X}_n\mathbf{v})^*\tau, n = 0, 1, \ldots, N-1$, *if* N *fixed matrices* $\mathbf{X}_n \in \mathbb{R}^{L\times M}$ *are chosen to represent the structure of the data matrix such that* $\mathbf{X} = \Sigma_{n=0}^{N-1} x_n\mathbf{X}_n$ *and* $\hat{\mathbf{X}} = \Sigma_{n=0}^{N-1}\hat{x}_n\mathbf{X}_n$.
- *The vector* $\mathbf{y}$ *is given as* $\mathbf{y} = \mathbf{v}/\|\mathbf{v}\|$.
- $\mathbf{D}_u$ *and* $\mathbf{D}_v$ *are positive or non-negative definite matrices defined via*

$$\sum_{n=0}^{N-1}\frac{1}{w_n}\mathbf{X}_n^H(\mathbf{u}^H\mathbf{X}_n\mathbf{v})\mathbf{u} = \mathbf{D}_u\mathbf{v} \text{ and } \sum_{n=0}^{N-1}\frac{1}{w_n}\mathbf{X}_n(\mathbf{u}^H\mathbf{X}_n\mathbf{v})^*\mathbf{v} = \mathbf{D}_v\mathbf{u}$$

  respectively.

- *The modified data vector is orthogonal to the correction vector in the sense that* $\hat{\mathbf{x}}^H\mathbf{W}(\mathbf{x} - \hat{\mathbf{x}})$.

This theorem gives a solution for any affinely structured complex data matrix. When the structure is Hankel, say, $\mathbf{X}$ is defined as (2.1), the STLS solution can simply be obtained from

$$\hat{\mathbf{x}} = \mathbf{x} - (\mathbf{u}\otimes\mathbf{v}^*)\ \tau$$
$$\mathbf{D}_u = \check{\mathbf{T}}_u\mathbf{W}^{-1}\check{\mathbf{T}}_u^H \quad \text{and} \quad \mathbf{D}_v = \check{\mathbf{T}}_v\mathbf{W}^{-1}\check{\mathbf{T}}_v^H \tag{5.17}$$

where $\otimes$ denotes convolution, $\check{\mathbf{T}}_u$ and $\check{\mathbf{T}}_v$ are Toeplitz matrices consisting of the elements of $\mathbf{u}$ and $\mathbf{v}$, respectively:

$$\check{\mathbf{T}}_u = \begin{bmatrix} u_1 & u_2 & \cdots & u_L & & 0 \\ & \ddots & & & \ddots & \\ 0 & & u_1 & u_2 & \cdots & u_L \end{bmatrix}$$

The above solutions for a complex data matrix with an affine structure, as well as a special case of Hankel structure, are actually direct extensions of the solutions for real data matrix given in [64].

The STLS algorithm, presented in [64], is based on inverse iteration. For given $\mathbf{D}_u$ and $\mathbf{D}_v$, we perform one step of inverse iteration, using the QR decomposition of $\mathbf{X}$, in order to obtain new estimates of $\mathbf{u}$ and $\mathbf{v}$. Then we update $\mathbf{D}_u$ and $\mathbf{D}_v$ for a new iteration.

Let $\mathbf{X} = [\mathbf{Q}_1 \quad \mathbf{Q}_2]\begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix} = \mathbf{Q}_1\mathbf{R}$, $\mathbf{Q}_1 \in \mathbb{C}^{L\times M}$, $\mathbf{Q}_2 \in \mathbb{C}^{L\times(L-M)}$, $\mathbf{R} \in \mathbb{C}^{M\times M}$. Then we can decompose $\mathbf{u}$ as

$$\mathbf{u} = [\mathbf{Q}_1 \quad \mathbf{Q}_2]\begin{bmatrix}\check{\mathbf{u}}_{Q1}\\ \check{\mathbf{u}}_{Q2}\end{bmatrix} = \mathbf{Q}_1\check{\mathbf{u}}_{Q1} + \mathbf{Q}_2\check{\mathbf{u}}_{Q2}$$

for certain $\check{\mathbf{u}}_{Q1} \in \mathbb{C}^{M\times 1}$, $\check{\mathbf{u}}_{Q2} \in \mathbb{C}^{(L-M)\times 1}$. From $\mathbf{X}^H\mathbf{u} = \mathbf{D}_u\mathbf{v}\tau$, $\mathbf{Q}_2^H(\mathbf{D}_u\mathbf{u}\tau) = \mathbf{Q}_2^H(\mathbf{X}\mathbf{v}) = \mathbf{0}$, $\mathbf{Q}_1^H(\mathbf{D}_v\mathbf{u}\tau - \mathbf{X}\mathbf{v}) = \mathbf{0}$, we find

$$\begin{bmatrix} \mathbf{R}^H & \mathbf{0} & \mathbf{0} \\ \mathbf{Q}_2^H\mathbf{D}_v\mathbf{Q}_1 & \mathbf{Q}_2^H\mathbf{D}_v\mathbf{Q}_2 & \mathbf{0} \\ \mathbf{Q}_1^H\mathbf{D}_v\mathbf{Q}_1 & \mathbf{Q}_1^H\mathbf{D}_v\mathbf{Q}_2 & -\mathbf{R} \end{bmatrix}\begin{bmatrix}\check{\mathbf{u}}_{Q1}\\ \check{\mathbf{u}}_{Q2}\\ \mathbf{v}/\tau\end{bmatrix} = \begin{bmatrix}\mathbf{D}_u\mathbf{v}\tau\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}$$

Solving the above system of linear equations with a block triangular system matrix iteratively results in the following STLS algorithm. Only Hankel structured data matrices are treated here, where $\mathbf{D}_u$ and $\mathbf{D}_v$ are explicitly defined by (5.17). For these structures, the STLS algorithm reduces as follows.

Algorithm 7

Outline of the STLS algorithm for Hankel structured $[\mathbf{A}_{L\times K}\ \mathbf{b}_{L\times 1}]$

Given. $\mathbf{A}, \mathbf{b}, \mathbf{W}$, *structure on* $\mathbf{X}_{L\times M} = [\mathbf{A}_{L\times K}\ \mathbf{b}_{L\times 1}]$, *tolerance* ϵ.

Step 1: Initialization. Choose $\mathbf{u}$ *and* $\mathbf{v}$ *(say* $\mathbf{u} := \mathbf{u}_L$, $\mathbf{v} := \mathbf{v}_M$ *from the SVD of* $\mathbf{X}$*), construct* $\mathbf{D}_v$ *from* $\mathbf{v}$ *according to (5.17), normalize* $\mathbf{u}$ *via* $\mathbf{u} := \mathbf{u}/\sqrt{\mathbf{u}^H\mathbf{D}_v\mathbf{u}}$, *construct* $\mathbf{D}_u$ *from* $\mathbf{u}$ *according to (5.17), and normalize* $\mathbf{v}$ *via* $\mathbf{v} := \mathbf{v}/\sqrt{\mathbf{v}^H\mathbf{D}_u\mathbf{v}}$.

Step 2: Iteration. Repeat

(a) $\check{\mathbf{u}}_{Q1} := \mathbf{R}^{-H}\mathbf{D}_u\mathbf{v}$, $\check{\mathbf{u}}_{Q2} := -(\mathbf{Q}_2^H\mathbf{D}_v\mathbf{Q}_2)^{-1}(\mathbf{Q}_2^H\mathbf{D}_v\mathbf{Q}_1)\check{\mathbf{u}}_{Q1}$,
$\mathbf{u} := \mathbf{Q}_1\check{\mathbf{u}}_{Q1} + \mathbf{Q}_2\check{\mathbf{u}}_{Q2}$, $\mathbf{u} := \mathbf{u}/\sqrt{\mathbf{u}^H\mathbf{D}_v\mathbf{u}}$,
construct $\mathbf{D}_u$ *from* $\mathbf{u}$.

(b) $\mathbf{v} := \mathbf{R}^{-1}\mathbf{Q}_1^H\mathbf{D}_v\mathbf{u}$, $\tau := 1/\sqrt{\mathbf{v}^H\mathbf{D}_u\mathbf{v}}$, $\mathbf{v} := \mathbf{v}\tau$,
construct $\mathbf{D}_v$ *from* $\mathbf{v}$.

Until $(\|\Delta \mathbf{v}\| \leq \epsilon)$

Step 3: Computation of $\hat{x}$ *and* $\hat{\mathbf{X}}$. *Compute* $\hat{x}_n$ *from* $\hat{x}_n = x_n - (\mathbf{u}^H \mathbf{X}_n \mathbf{v})^* r$, *and construct* $\hat{\mathbf{X}}$ *from* $\hat{x}_n$.

5.2.3 Discussion

In contrast to STLS, the STLN algorithm can handle multiple-right-hand-side problems, $\mathbf{AY} \approx \mathbf{B}$, and allows for other norms than $\| \ \|_2$, e.g., $\| \ \|_1$, $\| \ \|_\infty$ (see [69] for details). The following comparisons and discussion in this subsection are made for single-right-hand-side problems, $\mathbf{Ay} \approx \mathbf{b}$.

When $\mathbf{W} = \begin{bmatrix} \mathbf{D}_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta \end{bmatrix}^2$, STLN and STLS actually minimize the same object function but in different ways:

- STLN forces $\mathbf{b} + \Delta\mathbf{b}$ to lie in the column space of $\mathbf{A} + \Delta\mathbf{A}$, while STLS assures rank deficiency of $[\mathbf{A} + \Delta\mathbf{A} \ \ \mathbf{b} + \Delta\mathbf{b}]$.
- STLN imposes structure on $[\Delta\mathbf{A} \ \ \Delta\mathbf{b}]$, while STLS imposes structure on $[\mathbf{A} + \Delta\mathbf{A} \ \ \mathbf{b} + \Delta\mathbf{b}]$.
- STLN uses a constrained optimization approach, while STLS uses a Riemannian SVD approach.

Once convergence is reached, it is proven below that the optimality conditions for the optimization problem (5.1) are satisfied by the two STLN algorithms.

STLN Algorithm-1

In the STLN Algorithm-1, the optimality condition for a local optimum of the penalty function $P(\mathbf{z})$ in (5.6) is the vanishing of the gradients :

$$\frac{\partial P(\mathbf{z})}{\partial \mathbf{z}} = \begin{bmatrix} \frac{\partial P(\mathbf{z})}{\partial \alpha} \\ \frac{\partial P(\mathbf{z})}{\partial \beta_0} \\ \frac{\partial P(\mathbf{z})}{\partial \mathbf{y}} \end{bmatrix} = \begin{bmatrix} w^2(\mathbf{P}_\alpha - \check{\mathbf{Y}})^T \mathbf{r}^* + \mathbf{D}_\alpha^2 \alpha^* \\ w^2(\mathbf{P}_\beta)^T \mathbf{r}^* + \mathbf{D}_\beta^2 \beta^* \\ -w^2(\mathbf{A} + \Delta\mathbf{A})^T \mathbf{r}^* \end{bmatrix} = \mathbf{0}$$

This is equivalent to the convergence of the LS subproblems in Step 2(a) of STLN Algorithm-1, since the LS solution $\Delta\mathbf{z} = [\Delta\alpha^T, \Delta\beta_0^T, \Delta\mathbf{y}]^T$ in Step 2(a)

satisfies

$$(\mathbf{M}^H\mathbf{M})^{-1} \begin{bmatrix} \Delta\alpha \\ \Delta\beta_0 \\ \Delta\mathbf{y} \end{bmatrix} = -\mathbf{M}^H \begin{bmatrix} w\mathbf{r} \\ \mathbf{D}_\alpha \alpha \\ \mathbf{D}_\beta \beta_0 \end{bmatrix}$$

$$= -\begin{bmatrix} w^2(\mathbf{P}_\alpha - \check{\mathbf{Y}})^H \mathbf{r} + \mathbf{D}_\alpha^2 \alpha \\ w^2(\mathbf{P}_\beta)^H \mathbf{r} + \mathbf{D}_\beta^2 \beta \\ -w^2(\mathbf{A} + \Delta\mathbf{A})^H \mathbf{r} \end{bmatrix} = -\begin{bmatrix} \frac{\partial P(\mathbf{z})}{\partial \alpha} \\ \frac{\partial P(\mathbf{z})}{\partial \beta_0} \\ \frac{\partial P(\mathbf{z})}{\partial \mathbf{y}} \end{bmatrix}^*$$

where

$$\mathbf{M} = \begin{bmatrix} w(\mathbf{P}_\alpha - \check{\mathbf{Y}}) & w\mathbf{P}_\beta & -w(\mathbf{A} + \Delta\mathbf{A}) \\ \mathbf{D}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_\beta & \mathbf{0} \end{bmatrix} \tag{5.18}$$

As long as matrix $\mathbf{M}$ has full rank, $\Delta\mathbf{z}$ has a unique solution, which will be zero if, and only if, the right-hand-side gradients are zero.

STLN Algorithm-2

For our problem formulated as (5.10) in STLN Algorithm-2, the optimality condition for a local minimum is the vanishing of the gradients of the Lagrange function $\mathcal{L}(\mathbf{z}) = f(\mathbf{z}) + \lambda^T \mathbf{r}(\mathbf{z})$:

$$\frac{\partial \mathcal{L}(\mathbf{z})}{\partial \mathbf{z}} = 0 \quad \Rightarrow \quad \hat{\mathbf{A}}^T \hat{\mathbf{b}}^* + \hat{\mathbf{B}}^T \lambda = 0$$
$$\frac{\partial \mathcal{L}(\mathbf{z})}{\partial \lambda} = 0 \quad \Rightarrow \quad \hat{\mathbf{B}} \Delta\mathbf{z} - \hat{\mathbf{d}} = 0$$

When the solution $\Delta\check{\mathbf{z}} = \mathbf{0}$ and hence $\Delta\mathbf{z} = \mathbf{0}$ is obtained using the GSVD in Step 2(b) of STLN Algorithm-2, the above two equalities of optimality conditions are satisfied as shown below.

Referring(5.11) and introducing $\mathbf{U}_A = [\mathbf{U}_{A1}, \ \mathbf{U}_{A2}] = [\mathbf{u}_{A1}, \mathbf{u}_{A2}, \ldots, \mathbf{u}_{A\hat{m}}]$, $\mathbf{U}_{A1} \in \mathbb{C}^{\hat{m} \times (\hat{n} - \hat{p})}$, $\mathbf{U}_{A2} \in \mathbb{C}^{\hat{m} \times (\hat{m} + \hat{p} - \hat{n})}$, $\mathbf{u}_{Ai} \in \mathbb{C}^{\hat{m} \times 1}, i = 1, 2, \ldots, \hat{m}$, $\mathbf{U}_B = [\mathbf{u}_{B1}, \mathbf{u}_{B2}, \ldots, \mathbf{u}_{B\hat{p}}]$, $\mathbf{u}_{Bi} \in \mathbb{C}^{\hat{p} \times 1}, i = 1, 2, \ldots, \hat{p}$, and $\mathbf{P}_A = \begin{bmatrix} \mathbf{P}_{A1} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{A2} \end{bmatrix}$, with $\mathbf{P}_{A1} = \text{diag}\,(\rho_{A1}, \rho_{A2}, \ldots, \rho_{A(\hat{n} - \hat{p})})$, $\mathbf{P}_{A2} = \text{diag}\,(\rho_{A(\hat{n} - \hat{p} + 1)}, \rho_{A(\hat{n} - \hat{p} + 2)}, \ldots, \rho_{A\hat{m}})$. Then the solution of $\Delta\check{\mathbf{z}}$ in (5.13) can be written as

$$\Delta\check{z}_i = \begin{cases} \mathbf{u}_{Ai}^H \hat{\mathbf{b}} / \rho_{Ai}, & i = 1, 2, \ldots, (\hat{n} - \hat{p}) \\ \mathbf{u}_{B(i + \hat{p} - \hat{n})}^H \hat{\mathbf{d}} / \rho_{B(i + \hat{p} - \hat{n})}, & i = \hat{n} - \hat{p} + 1, \hat{n} - \hat{p} + 2, \ldots, \hat{n} \end{cases}$$

This means that

$$\Delta\check{\mathbf{z}} = \begin{bmatrix} \mathbf{P}_{A1}^{-1}\mathbf{U}_{A1}^H\hat{\mathbf{b}} \\ \mathbf{P}_B^{-1}\mathbf{U}_B^H\hat{\mathbf{d}} \end{bmatrix}$$

It follows immediately that

$$\text{if } \Delta\check{\mathbf{z}} = \mathbf{0} \text{ then } \Rightarrow \quad \mathbf{U}_{A1}^H\hat{\mathbf{b}} = \mathbf{0}, \quad \hat{\mathbf{d}} = \mathbf{0}, \text{and } \hat{\mathbf{B}}\Delta\mathbf{z} - \hat{\mathbf{d}} = \mathbf{0} \qquad (5.19)$$

Thus the second equality is proven. In the following, we show that the first equality with the complex Lagrange multipliers vector, $\lambda = -(\hat{\mathbf{b}}^H\mathbf{U}_A\mathbf{D}_A\mathbf{D}_B^\dagger\mathbf{U}_B^H)^T$, is satisfied for $\Delta\check{\mathbf{z}} = \mathbf{0}$:

$$\begin{aligned}
\hat{\mathbf{A}}^T\hat{\mathbf{b}}^* \;+\; & \hat{\mathbf{B}}^T\lambda = \mathbf{Z}_{AB}^T(\mathbf{D}_A^T\mathbf{U}_A^T\hat{\mathbf{b}}^* + \mathbf{D}_B^T\mathbf{U}_B^T\lambda) \\
= \; & \mathbf{Z}_{AB}^T(\mathbf{I}_{\hat{n}\times\hat{n}} - (\mathbf{D}_B^\dagger\mathbf{D}_B)^T)\mathbf{D}_A^T\mathbf{U}_A^T\hat{\mathbf{b}}^* \\
= \; & \mathbf{Z}_{AB}^T\begin{bmatrix} \mathbf{I}_{(\hat{n}-\hat{p})\times(\hat{n}-\hat{p})} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}\begin{bmatrix} \mathbf{P}_{A1} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{A2} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\begin{bmatrix} \mathbf{U}_{A1}^T \\ \mathbf{U}_{A2}^T \end{bmatrix}\hat{\mathbf{b}}^* \\
= \; & \mathbf{Z}_{AB}^T\begin{bmatrix} \mathbf{P}_{A1}\mathbf{U}_{A1}^T\hat{\mathbf{b}}^* \\ \mathbf{0}_{\hat{p}\times 1} \end{bmatrix}
\end{aligned}$$

where † denotes the pseudo-inverse and $\mathbf{D}_B^\dagger = \begin{bmatrix} \mathbf{0} \\ \mathbf{P}_B^{-1} \end{bmatrix}$. Since $\mathbf{U}_{A1}^T\hat{\mathbf{b}}^* = (\mathbf{U}_{A1}^H\hat{\mathbf{b}})^* = \mathbf{0}$ from (5.19), $\frac{\partial\mathcal{L}(\mathbf{z})}{\partial\mathbf{z}} = 0$ is satisfied for the Lagrange multiplier vector $\lambda = -(\hat{\mathbf{b}}^H\mathbf{U}_A\mathbf{D}_A\mathbf{D}_B^\dagger\mathbf{U}_B^H)^T$.

However, the optimization problem (5.1) is not convex. There is no guarantee that these two algorithms converge to a global minimum. In fact, they will converge to the closest local minimum when the residual $\mathbf{r}(\mathbf{z})$ is sufficiently small. In the rest of this thesis, we will mainly use STLN Algorithm-2, because STLN Algorithm-1 is non-automatic (in the choice of w) and less accurate, though faster, than STLN Algorithm-2 due to numerical problems (ill-conditioning when w is large, see Example 2 of Section 5.4.1) [76].

5.3 New parameter estimator HTLN

If we apply the STLN technique to our noise-corrupted data matrix $[\mathbf{A}\ \mathbf{B}]$ in (2.1), we end up with a rank-K Hankel matrix approximation $[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]$ and a solution $\mathbf{Y}$. From $[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]$, a "cleaned-up" signal is obtained, given by the first column and last row of the matrix $[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]$. Obviously, such a "cleaned-up" (though may be deviated) signal can be written

as an exact sum of K exponentially damped sinusoids, or an impulse response of order K. This enables several ways of simplification in the subsequent parameter estimator when "cleaned-up" data are used, as is the case of the MV or Cadzow enhancement when convergence is reached.

Refer to p. 74 in Chapter 4. When the STLN enhancement is fully convergent, one way of increasing efficiency is to reduce the number of data samples used in the pole-estimation algorithm to as small as $2K-1$ without affecting the resulting estimates. In particular, we can simply arrange the $2K-1$ "cleaned-up" data into a matrix $\mathbf{X}$ of size $K \times K$, and compute nonlinear parameters from eigenvalues of the solution $\tilde{\mathbf{Z}}$ in $\mathbf{X}^\uparrow = \mathbf{X}_\downarrow\tilde{\mathbf{Z}}$, without performing the SVD.

However, the enhancement is typically *not fully* convergent due to the limit of numerical precision, and the most efficient and accurate estimator, HTLS, is suggested for the enhanced data. Such an STLN-HTLS estimator, unlike cMV-HTLS or cCA-HTLS, can also be simplified and improved in efficiency by using the computed STLN solution $\mathbf{Y}$. This implies that we do not need to run the HTLS algorithm from the beginning after STLN enhancement. In this way, STLN-HTLS reduces to HTLN (stands for Hankel Total Least Norm), as shown below.

Denote the SVD of $\mathbf{X}$ by $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^H, \mathbf{V} = [\mathbf{V}_s, \mathbf{V}_n] = [\mathbf{V}_K, \overline{\mathbf{V}}_K]$, where the subscripts $_s$ and $_n$ denote the signal and null space of $\mathbf{X}$, respectively. Consider the QR decomposition of the STLN solution $\begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix} = [\mathbf{Q}_n\ \ \mathbf{Q}_s]\begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_n\mathbf{R}$, which satisfies $\mathbf{X}\begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix} = \mathbf{0}$. Then, it is easy to see that $\mathbf{Q}_n$ and $\mathbf{V}_n$ span the same null space, while $\mathbf{Q}_s$ and $\mathbf{V}_s$ span the same signal space of $\mathbf{X}$. Therefore, the SVD of $\mathbf{X}$ in the HTLS estimator can be replaced by the QR decomposition of $\begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix}$, to find the orthonormal signal subspace basis with the shift-invariance property.

Algorithm 8

Outline of the HTLN algorithm

Given. A model-order estimate K, and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L\times M$ Hankel matrix $[\mathbf{A}_{L\times K}\ \ \mathbf{B}_{L\times(M-K)}]$ as in (2.1).

Step 1. Apply the STLN algorithm to $[\mathbf{A}\ \mathbf{B}]$ in order to find the Hankel cor-

rection matrix $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$ *and STLN solution* $\mathbf{Y}$ *such that*

$$[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]\begin{bmatrix}\mathbf{Y}\\ -\mathbf{I}\end{bmatrix}=\mathbf{0}$$

Step 2. Compute a QR decomposition of

$$\begin{bmatrix}\mathbf{Y}\\ -\mathbf{I}\end{bmatrix}=[\mathbf{Q}_n\ \ \mathbf{Q}_s]\begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix}$$

where $[\mathbf{Q}_n\ \ \mathbf{Q}_s]$ *is unitary, and* $\mathbf{R}$ *is upper triangular.*

Step 3. Compute the TLS solution $\mathbf{Z}^{(Q)}$ *of the following equation:*

$$\mathbf{Q}_s^{*\uparrow}\approx\mathbf{Q}_{s\downarrow}^{*}\mathbf{Z}^{(Q)}$$

and retrieve the eigenvalues z_k *of* $\mathbf{Z}^{(Q)}$*. The* f_k, d_k *are obtained straightforwardly from* z_k*, using the relation* $z_k = e^{(-d_k+j2\pi f_k)/f_s}$.

Step 4. Apply the linear-parameter-estimation part (Step 3) of the HTLS Algorithm 1, to the original data x_n, $n=0,1,\cdots,N-1$, *to find* a_k *and* φ_k, $k=1,\cdots,K$.

The improved efficiency of HTLN, compared to STLN-HTLS, stems from the use of the QR decomposition of $\begin{bmatrix}\mathbf{Y}\\ -\mathbf{I}\end{bmatrix}$, instead of the SVD of $[\mathbf{A}\ \mathbf{B}]$, for finding the orthonormal signal subspace basis. This is more accurate and faster than solving the set of equation $\mathbf{X}^{\uparrow}=\mathbf{X}_{\downarrow}\tilde{\mathbf{Z}}$ mentioned earlier, where $\mathbf{X}_{K\times K}$ is composed of $2K-1$ enhanced data.

Actually, an even more efficient way of finding the poles z_k after application of the STLN algorithm is to use the first column of the STLN solution $\mathbf{Y}$ as LP filter coefficients so that the z_k are simply obtained by rooting the corresponding LP polynomial of order K. This works well when the matrix $[\mathbf{A}+\Delta\mathbf{A}\ \ \mathbf{B}+\Delta\mathbf{B}]$ is perfectly of rank K. When STLN is not fully convergent, e.g., when the tolerance ϵ in the stop criterion of the STLN iterations in (5.9) or (5.14) is not small enough (say, $\epsilon<10^{-8}$ in fitting *Signal-1*), such a rooting method is numerically not as reliable as HTLN. Moreover, the time saving is not significant compared to the computation time of STLN. Therefore we focus on the HTLN algorithm when estimating parameters in this chapter.

5.4 Numerical comparison of different algorithms

5.4.1 Optimality of the STLN, STLS solutions

In this subsection, we use two examples, for which the global minimum can be found analytically, and STLN, STLS converge to the optimal solutions while Cadzow's method does not. These examples arise in the problems of approximating a given data sequence by the impulse response of a finite-dimensional Linear Time-Invariant (LTI) system of a given order, such that the weighted sum of squared residuals is minimized, i.e.,

Approximate a given data sequence $x_n\in\mathbb{C}$ *by* $\hat{x}_n\in\mathbb{C}$, $n=0,1,\ldots,N-1$ *so as to minimize:*

$$\sum_{n=0}^{N-1}|x_n-\hat{x}_n|^2w_n\quad\text{subject to}\quad\hat{\mathbf{X}}_{L\times M}\begin{bmatrix}\mathbf{Y}_{K\times(M-K)}\\ -\mathbf{I}_{(M-K)\times(M-K)}\end{bmatrix}=\mathbf{0}\qquad(5.20)$$

where $\hat{\mathbf{X}}$ is an $L\times M$ Hankel matrix, $L\geq M$, constructed from the elements $\hat{x}_n$, and w_n are appropriate weights. The constraint imposing that $\hat{\mathbf{X}}$ has at most rank K ensures that $\hat{x}_n$ is the impulse response of a finite-dimensional linear system of order K at most. If the sequence x_n is itself the impulse response of a higher dimensional system, our problem corresponds to model reduction. If the sequence x_n is a given data sequence which is not an impulse response, but for instance, a noise-corrupted one, one might consider this problem as a noisy realization problem, or a signal enhancement problem. If additionally all $w_n=1$, we get model reduction in the L_2 norm.

Suppose the weights w_n account for the repetition of elements $\hat{x}_n$ in the Hankel matrix $\hat{\mathbf{X}}$, then

$$\begin{aligned}\mathbf{W}&=\operatorname{diag}(w_0,w_1,\ldots,w_{N-1})\\ &=\begin{bmatrix}\mathbf{D}_\alpha^2 & \mathbf{0}\\ \mathbf{0} & \mathbf{D}_\beta^2\end{bmatrix}=\operatorname{diag}(1,2,\ldots,M,\ldots,M,\ldots,2,1)\end{aligned}$$

and (5.20) reduces to a STLN optimization problem with $(M-K)$ right-hand-sides as we have discussed earlier. If $M=K+1$, the problem can also be solved by the STLS algorithm presented in the previous section.

As proven in [64], there is one special case for which the global optimum of (5.20) can be found analytically, i.e.

Approximate a given data sequence $x_n\in\mathbb{C}$ *by a first order LTI impulse response, which can be parameterized as* $\hat{x}_n=\gamma\eta^n\in\mathbb{C}$, $n=0,1,\ldots,N-1$.

Since a Hankel matrix $\hat{\mathbf{X}}_{L\times M}$, composed of the elements $\hat{x}_n=\gamma\eta^n$, is always a rank-1 matrix, we obtain a minimization problem without the rank

deficiency constraint:

$$\min_{\gamma,\eta} \sum_{n=0}^{N-1} |x_n - \gamma\eta^n|^2 w_n = \min_{\gamma,\eta} f(\gamma,\eta) \tag{5.21}$$

The global minimum of (5.21) can be obtained by setting the derivatives with respect to γ and η to zero:

$$\frac{\partial f}{\partial \gamma} = 0 \quad \Rightarrow \quad \sum_{n=0}^{N-1} w_n(x_n - \gamma\eta^n)(-\eta^n) = 0,$$

$$\frac{\partial f}{\partial \eta} = 0 \quad \Rightarrow \quad \sum_{n=1}^{N-1} w_n(x_n - \gamma\eta^n)(-n\eta^{n-1}) = 0.$$

Eliminating γ from the first equation and substituting it in the second, we get the polynomial in η:

$$\left(\sum_{n=1}^{N-1} n w_n x_n \eta^{n-1}\right)\left(\sum_{n=0}^{N-1} w_n \eta^{2n}\right) - \left(\sum_{n=0}^{N-1} w_n x_n \eta^n\right)\left(\sum_{n=1}^{N-1} n w_n \eta^{2n-1}\right) = 0 \tag{5.22}$$

This is a polynomial of degree $3(N-1)-2$ in η. One has to select the root which gives the minimum in (5.21).

Example 1

Consider the same data sequence $\mathbf{x} = [6\ 5\ 4\ 3\ 2\ 1]^T$, as given in [64], which can be modeled as the impulse response of the following second order LTI system:

$$\begin{aligned} s_{n+1} &= \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} s_n \quad \text{with initial state} \quad s_0 = \begin{bmatrix} 6 \\ 1 \end{bmatrix} \\ x_n &= [1\ 0]\, s_n \end{aligned}$$

We arrange the data $x_n, n = 0, 1, \ldots, N-1$ in the $L \times M = 5 \times 2$ Hankel matrix $\mathbf{X}$, and find the best rank-1 structure-preserving approximation $\hat{\mathbf{X}}$ to $\mathbf{X}$

$$\mathbf{X} = \begin{bmatrix} x_0 & x_1 \\ x_1 & x_2 \\ x_2 & x_3 \\ x_3 & x_4 \\ x_4 & x_5 \end{bmatrix} = \begin{bmatrix} 6 & 5 \\ 5 & 4 \\ 4 & 3 \\ 3 & 2 \\ 2 & 1 \end{bmatrix}, \quad \hat{\mathbf{X}} = \begin{bmatrix} \hat{x}_0 & \hat{x}_1 \\ \hat{x}_1 & \hat{x}_2 \\ \hat{x}_2 & \hat{x}_3 \\ \hat{x}_3 & \hat{x}_4 \\ \hat{x}_4 & \hat{x}_5 \end{bmatrix}$$

method	$w_n = 1,2,2,2,2,1$		
	# iterations	# flops	η
Rooting polynomial			0.7629230150743218
STLN Algorithm-1	14	108358	0.7629230150743225
STLN Algorithm-2	14	472887	0.7629230150743224
STLS	9	17202	0.7629230150743225
Cadzow	87	34365	0.7627806087028032
method	$w_n = 1,1,1,1,1,1$		
	# iterations	# flops	η
Rooting polynomial			0.7602263542172322
STLN Algorithm-1	15	116159	0.7602263542172321
STLN Algorithm-2	14	417775	0.7602263542172322
STLS	8	15309	0.7602263542172323

Table 5.1: Example 1: number of iterations, number of flops and values of η at convergence. The tolerance is $\epsilon = 10^{-15}$ for all the methods. The best weight, after several tests, is used in STLN Algorithm-1, which is $w = 10^{10}$ in this case.

by solving the corresponding STLN problem with weighting, $w_n = 1,2,2,2,2,1$, and without weighting, $w_n = 1,1,1,1,1,1$, for $n = 0,1,2,3,4,5$. The results of computation are listed in Table 5.1. The true values of η which give the minimum in (5.21) are obtained by finding the roots of (5.22) with $L = 5, K = 1, M = 2$ using MATLAB version 4.2a. The same minima are also obtained by the STLN Algorithm-1, STLN Algorithm-2 and the STLS algorithm, up to 14 digits, from each ratio $\hat{x}_{n+1}/\hat{x}_n \ \ n = 0,1,\ldots,4$. This shows that the algorithms of STLN and STLS converge to the optimal solution in this example, though both STLN Algorithms, especially STLN Algorithm-2, needs large number of flops. On the other hand, the suboptimal methods such as Cadzow's algorithm does not converge to the true solution.

Example 2

Consider the following fourth order LTI system:

$$s_{n+1} = \begin{bmatrix} 0.4 & 0 & 0 & 0 \\ 0 & 0.3 & 0 & 0 \\ 0 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix} s_n \quad \text{with initial state} \quad s_0 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

method	$w_n = 1,2,3,4,4,3,2,1$		
	# iterations	# flops	η
Rooting polynomial			0.2602566142908348
STLN Algorithm-1[a]	58	1867699	0.2602566142908066
STLN Algorithm-2	10	1434083	0.2602566142908349
Cadzow	82	116166	0.2602455147443435
methods	$w_n = 1,1,1,1,1,1,1$		
	# iterations	# flops	η
Rooting polynomial			0.2568639874063329
STLN Algorithm-1[b]	103	3320939	0.2568639874063330
STLN Algorithm-2	8	1064917	0.2568639874063330

[a] The best weight is $w = 10^6$. A larger w causes numerical problems.
[b] The best weight is $w = 10^7$. A larger w causes numerical problems.

Table 5.2: Example 2: number of iterations, number of flops and values of η at convergence. The tolerance is $\epsilon = 10^{-15}$ for both algorithms.

$$x_n = [1\ 1\ 1\ 1]\, s_n$$

Arrange the first 8 samples of the impulse response, $\mathbf{x} = [4\ 1\ 0.3\ 0.1\ 0.0354\ 0.013\ 0.00489\ 0.00187]^T$, in an $L \times M = 5 \times 4$ Hankel matrix $\mathbf{X}$ as done in Example 1. Table 5.2 shows the results of computing the best rank-1 structure-preserving approximation $\hat{\mathbf{X}}$ to $\mathbf{X}$ using different methods. Again, STLN Algorithm-2 reaches the known optimal solution for η up to at least 14 digits, while Cadzow's approach converges to suboptimal solutions. Since this problem has $M - K = 3$ right-hand-sides, the STLS method, which only reduce the rank of $\mathbf{X}$ by one, can not be considered here.

Note that when w is large and the number of right-hand-sides increases, there tend to be more ill-conditioning or non-convergent problems with STLN Algorithm-1. Typically, ill-conditioning occurs when $\mathbf{M}$ in (5.18) becomes rank deficient. A lower weight, w, can reduce such problems, but it leads to inaccurate results. Therefore, *we stick to STLN Algorithm-2 in the rest of this chapter.*

Similar results have been obtained for various other sequences x_n and other weights w_n for the problem (5.21).

5.4.2 Simulation results

In this section, several enhancement algorithms, Cadzow's, MV, STLS, STLN, and the corresponding enhanced parameter estimators CA-HTLS, MV-HTLS, STLS-HTLS, and HTLN are compared in terms of resolution, accuracy, and computational efficiency, based on computer simulations.

Simulation procedure

The complex signal, *Signal-1*, with $N = 25$ data points is processed by different algorithms. The maximum number of iterations of the iterative enhancement methods is set to 100. The tolerance ϵ in the stop criterion of iterative methods is $\epsilon = 10^{-10}$. If not specified, the iterative methods converge within 100 iterations. Different techniques such as CA, MV, STLS, and STLN, which solve the incompatible set $\mathbf{AY} \approx \mathbf{B}$ by adding different Hankel correction matrices $[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]$ to the noisy, full-rank Hankel matrix $[\mathbf{A}\ \ \mathbf{B}]$ so that the set $(\mathbf{A} + \Delta\mathbf{A})\mathbf{Y} = (\mathbf{B} + \Delta\mathbf{B})$ is exactly solvable, are first applied to signal enhancement and compared in Frobenius norm $\|[\Delta\mathbf{A}\ \ \Delta\mathbf{B}]\|_F$ averaged over 200 noise realizations. Finally, the performance of different enhanced estimation methods CA-HTLS, MV-HTLS, STLS-HTLS, and HTLN are studied, in terms of respectively resolution and parameter accuracy, by comparing the number of failures and the RMSE. The size $L \times M$ of the Hankel data matrix used in Cadzow's enhancement procedure and the HTLS estimator is 13×13, since a square matrix produces the best results for HTLS and Cadzow's method takes the same dimension as that of the applied estimator. The size $L \times M$ used in the MV, STLS and STLN enhancement procedures is 23×3, unless specified otherwise. STLN Algorithm-2 is used in STLN and HTLN methods.

Enhancement results

Fig. 5.1 shows the spectrum of a noisy ($\sigma_\nu = 0.1257$ or SNR=15dB) simulated *signal-1*, the signal enhanced by STLN, and its residue. Here, $N = 81$ is chosen for both STLN enhancement and FFT spectrum so that a nice resolution in the spectrum is achieved. Clearly, the STLN-enhanced signal looks smooth and "clean". The signal spectra enhanced by other methods such as CA, MV and STLS are similar and hence not shown. The difference between the considered enhancement methods is not clear from their residue spectra.

A better comparison is obtained by comparing the corrections applied by each method to the data matrices or data vectors when solving $\mathbf{AY} \approx \mathbf{B}$. Table 5.3 compares the corrections applied by the enhancement methods studied, as

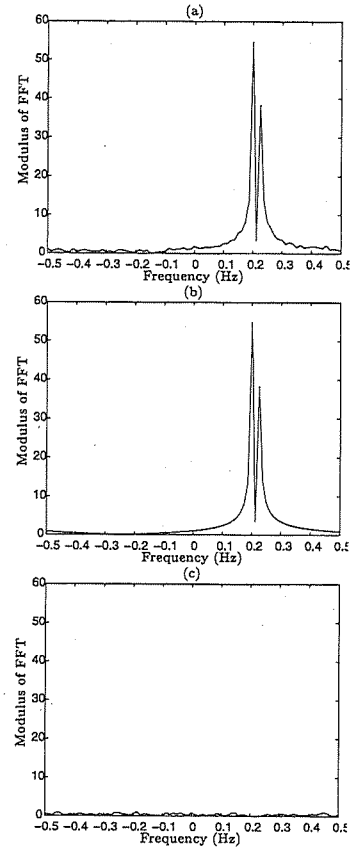


Figure 5.1: $N = 81$, SNR=15dB, $\sigma_\nu = 0.1257$, $M = K + 1$. (a) Original noisy signal spectrum; (b) STLN-enhanced signal spectrum; (c) residual spectrum.

SNR(dB)	method	M	# iteration[a]	corr_M	corr_v
25	TRUE[b]	3		$4.627e-01$	$2.781e-01$
	Cadzow	13	32	$6.693e-01$	$2.628e-01$
	MV[c]	3	100	$4.087e-01$	$2.476e-01$
	STLS	3	14	$4.272e-01$	$2.581e-01$
	STLN	3	10	$4.272e-01$	$2.581e-01$
	LS	3		$4.297e-01$	
	TLS	3		$2.650e-01$	
25	TRUE	4		$5.218e-01$	$2.792e-01$
	MV	4	100	$4.758e-01$	$2.563e-01$
	STLN	4	17	$4.815e-01$	$2.593e-01$
	LS	4		$6.837e-01$	
	TLS	4		$3.651e-01$	

[a] Averaged number of iterations over convergent cases.
[b] True correction or simulated noise.
[c] In all 200 runs, the MV method does not converge within 100 iterations.

Table 5.3: Expected (averaged over 200 runs) Frobenius norm, corr_M and corr_v, of correction matrices and correction vectors given by the first column and the last row of Hankel data matrices.

well as those of other techniques such as LS and TLS that solves $\mathbf{AY} \approx \mathbf{B}$. In all runs, the MV method did not converge within 100 iterations. All other iterative methods converge within 100 iterations, if not indicated otherwise. In Table 5.3, corr_M is the expected Frobenius norm of the correction matrix $[\Delta\mathbf{A} \;\; \Delta\mathbf{B}]$. If $[\Delta\mathbf{A} \;\; \Delta\mathbf{B}]$ is Hankel, then corr_v represents the expected Frobenius norm of the corresponding correction vector given by the first column and the last row of $[\Delta\mathbf{A} \;\; \Delta\mathbf{B}]$. Note that the corrections of the TLS method are always minimal, which can be easily proven. The corrections applied by STLN and STLS are about the same, larger than the correction applied by TLS which does not take the Hankel structure into account but lower than the true correction (or simulated noise) and the LS correction. Although MV has lower corrections, it does not successfully reduce the rank of the Hankel matrix $[\mathbf{A} \;\; \mathbf{B}]$ to $K = 2$ since it fails to converge within 100 iterations. Hence the correction applied by STLN and STLS is the lowest among the methods that preserve the Hankel structure and reduce the rank to K.

Parameter estimation results

All enhancement methods give very smooth signals when convergence occurs. However, as pointed out in Chapter 4, the convergence of the MV and Cadzow's algorithms is redundant when followed by an SVD-based estimator, such as HTLS, since the first iteration is the most decisive one. Therefore, the MV and Cadzow's method perform only one iteration when used to enhance the parameter estimates in the following simulation.

In STLS enhancement, the matrix size can only be chosen as $L \times M = 23 \times 3$ here for *Signal-1* due to the restriction of $M = K + 1$. In MV and STLN enhancement, more choices of the matrix size is possible, as long as $M \geq K + 1 = 3$. However, simulation results show that the MV enhancement gives comparable resolution and accuracy for *Signal-1* when M is equal to or a bit larger than $K+1$. However, the STLN enhancement performs best for this signal at $M = K + 1$. The reason is perhaps the larger number of constraints $\mathbf{r}(\mathbf{z})$ than the number of the unknowns $\mathbf{z}$ in (5.5) when $M - K$ is larger than 1 for this signal. Therefore, we will use the same size, $L \times M = 23 \times 3$, in the STLS, STLN and MV enhancement.

Table 5.4 (a) and (b) list the results obtained with the different methods. At SNR's as high as 25 dB, no failure in convergence or failure in resolving the two peaks has been observed. HTLN and STLS-HTLS produce almost the same results. These methods obtain lower RMSE values and hence are more accurate than HTLS, CA-HTLS and MV-HTLS. At SNR=20 dB, the iterative methods HTLN and STLS-HTLS still converge well, typically within 30 iterations. But there are some failures for all the methods. The iterative methods have fewer failures than the non-iterative methods HTLS, CA-HTLS and MV-HTLS, and show lower RMSE results. When the SNR is lowered to 15 dB, there are several runs in which HTLN and STLS-HTLS fail to converge within 100 iterations. Among the runs in which convergence occurred, there are still failures. At this SNR, MV-HTLS can better resolve the two interfering peaks since it produces the least number of failures. The RMSE of MV-HTLS and the iterative methods are comparable, all of which are lower than HTLS and CA-HTLS. Looking at the bias and standard deviation, we can see that HTLN and STLS-HTLS reduce the bias for d_1 with a factor 2 to 4 compared to HTLS at all the SNR's. In addition, HTLN and STLS-HTLS have lower standard deviation for both f_1 and d_1 at almost all the SNR's.

Computational efficiency

Although HTLN and STLS-HTLS are more accurate than other methods,

SNR	method	nc[a]	fail.	RMSE f_1	Bias±std. f_1
25dB	HTLS		0	0.00239	-0.00022 ± 0.00239
	CA-HTLS		0	0.00228	-0.00014 ± 0.00228
	MV-HTLS		0	0.00220	-0.00005 ± 0.00221
	STLS-HTLS	0	0	0.00208	-0.00013 ± 0.00208
	HTLN	0	0	0.00208	-0.00013 ± 0.00208
20dB	HTLS		11	0.00392	-0.00050 ± 0.00389
	CA-HTLS		9	0.00374	-0.00033 ± 0.00374
	MV-HTLS		11	0.00352	-0.00028 ± 0.00352
	STLS-HTLS	0	3	0.00353	-0.00060 ± 0.00348
	HTLN	0	3	0.00355	-0.00056 ± 0.00352
15dB	HTLS		96	0.00506	-0.00027 ± 0.00508
	CA-HTLS		62	0.00491	-0.00047 ± 0.00491
	MV-HTLS		46	0.00464	-0.00033 ± 0.00464
	STLS-HTLS	12	71	0.00427	-0.00062 ± 0.00425
	HTLN	2	86	0.00453	-0.00043 ± 0.00453

[a] Total number of runs, out of 200, in which the iterative method fails to converge after 100 iterations.

Table 5.4: (a): Number of non-convergence and failures, RMSE, bias and standard deviation of f_1.

they are less efficient as can be seen from the number of flops in Table 5.5. Therefore, one must trade off between the efficiency and accuracy requirement when choosing among optimal and suboptimal, iterative and non-iterative methods. The computational cost of these methods gets larger as the number of data samples N is larger or the SNR is lower. Moreover, the memory requirement gets larger when N is increased. This hampers the practical application of these methods in NMR spectroscopy.

5.5 Conclusion

In this chapter, the STLN and STLS algorithms are studied and applied to signal enhancement and parameter estimation of complex exponentially damped sinusoids. As a matter of fact, STLN and STLS can be applied in many other applications, wherever TLS is applied to structured problems and ML estimates are desired.

SNR	method	RMSE d_1	Bias±std. d_1
25dB	HTLS	0.01489	-0.00214 ± 0.01477
	CA-HTLS	0.01431	-0.00154 ± 0.01426
	MV-HTLS	0.01360	0.00022 ± 0.01363
	STLS-HTLS	0.01252	-0.00055 ± 0.01254
	HTLN	0.01252	-0.00055 ± 0.01254
20dB	HTLS	0.02714	0.00086 ± 0.02720
	CA-HTLS	0.02877	-0.00053 ± 0.02884
	MV-HTLS	0.02357	0.00390 ± 0.02330
	STLS-HTLS	0.02420	0.00045 ± 0.02426
	HTLN	0.02414	0.00046 ± 0.02419
15dB	HTLS	0.05980	-0.00411 ± 0.05995
	CA-HTLS	0.05517	-0.00239 ± 0.05532
	MV-HTLS	0.04255	0.00469 ± 0.04243
	STLS-HTLS	0.04157	0.00000 ± 0.04175
	HTLN	0.04550	0.00185 ± 0.04567

Table 5.4: (cont.) (b): RMSE, bias and standard deviation of d_1.

These two algorithms are closely related in that for single-right-hand-side problems, $\mathbf{Ay} \approx \mathbf{b}$, they minimize the same object function but in different ways

- STLN imposes structure on $[\Delta\mathbf{A} \;\; \Delta\mathbf{b}]$, while STLS imposes structure on $[\mathbf{A} + \Delta\mathbf{A} \;\; \mathbf{b} + \Delta\mathbf{b}]$.
- STLN uses a constrained optimization approach, while STLS transforms the problem to the computation of a Riemannian SVD via an inverse iteration based algorithm.
- Both STLN and STLS achieve optimal solutions.
- The results of enhancement by STLN are similar to those of STLS, both remove corruption noise effectively and improve resolution and parameter accuracy, especially, the standard deviations of the estimated parameters.

Comparison of the STLN and STLS algorithms with Cadzow's algorithms shows that the STLN and STLS algorithms can provide optimal solutions while Cadzow's algorithms can not. This partly explains the better accuracy performance in our parameter estimation, especially the lower bias and standard

methods	SNR=25dB	SNR=20dB	SNR=15dB
HTLS	1.19e+05	1.21e+05	1.23e+05
CA-HTLS	2.23e+05	2.25e+05	2.27e+05
MV-HTLS	1.24e+05	1.24e+05	1.25e+05
STLS-HTLS	8.43e+06	1.38e+07	2.60e+07
HTLN	4.86e+07	6.55e+07	1.15e+08

Table 5.5: Number of flops required by different methods using Matlab averaged over 200 runs at SNR=25dB, 20dB, and 15dB.

deviation of the estimated parameters by HTLN and STLS-HTLS. However, similar to Cadzow's algorithms, STLN and STLS also introduce colored noise and the obtained K exponentials are not the exact noise-free signals but deviated ones.

A common problem with STLN and STLS is that they are computationally expensive. A more efficient STLN algorithm has been proposed in [77] based on the Karush-Kuhn-Tucker equations, which is merely Newton's method applied to the Lagrange function of our optimization problem with nonlinear constraints. It is implemented using the sequential quadratic programming approach and line search [73]. In fact, STLN Algorithm2 is a special case of this new algorithm when the Hessian of constraints is not taken into account, in which case the object functions and constraints of the two algorithms are the same. By including the second order constraint terms, which is important for obtaining superlinear convergence rate for nonlinear constraints [73], the new method achieves better convergence rate at similar accuracy. Nevertheless, this is still not sufficient for processing real-world NMR spectroscopic signals, which have typically large number of data samples and low SNR's. Further work needs to be done in decreasing the computational cost of the STLN and STLS algorithms, which is crucial for real-world applications.

Chapter 6

Incorporation of prior knowledge of known signal poles

Up till now, we have presented several prefiltered and enhanced parameter estimation methods, which improve the resolution and parameter accuracy to different degree. These new methods are based on the *techniques* of filtering, minimum variance estimation, STLN and STLS, etc. Intuitively, if some *prior knowledge* is available and incorporated, the resolution and accuracy of the estimates can be further improved. For example, in NMR spectroscopy and other applications, some of the signal poles might be known *a priori*. The formulation of such prior knowledge and its incorporation into the TLS-FLP method, briefly mentioned in Section 2.2 of Chapter 2, has been described by Dowling et al. in [5][79], where the known signal poles are incorporated via linear constraints. The resulting method, using the linearly constrained TLS (LCTLS) technique and called LCTLS-FLP, produces better results than the TLS-FLP method, especially when the known signal poles are close to the unknown ones in the spectrum. In this chapter, we incorporate the same prior knowledge into more accurate estimators, such as TLS-BLP, HTLS and HTLN, and show that the resulting methods, LCTLS-BLP, HTLS(PK) and HTLN(PK), outperform the LCTLS-FLP method.

The rest of this chapter is arranged as follows. Section 1 reviews briefly some of the above-mentioned parameter estimators. Section 2 describes the corresponding prior-knowledge-incorporating estimators. In Section 3, results

of computer simulations using different estimators are presented. Conclusions are given in Section 4. More analysis and applications of the most appropriate estimator, HTLS(PK), will be given in the next chapter.

6.1 Brief review of some parameter estimators

Before incorporating prior knowledge of known signal poles into parameter estimators like TLS-FLP, TLS-BLP, HTLS and HTLN, we first review the theory of these estimation methods for clarity of exposition. Since the HTLS and HTLN algorithms have been described in Chapter 2 and 5 respectively, we only introduce the TLS-FLP and TLS-BLP algorithms, i.e., the forward and backward linear prediction methods based on the TLS technique.

Assume that an $L \times M$ Hankel matrix $\mathbf{X}$, defined in (2.1), consisting of N uniformly sampled data points $x_n, n = 0, 1, \ldots, N-1$, and a model-order estimate K are given.

TLS-FLP [20][25]

$$\text{Let} \quad [\mathbf{X}^{(FLP)}_{L\times(M-1)} \quad \mathbf{x}^{(FLP)}_{L\times 1}] = \mathbf{X}, \qquad L \geq K, M \geq K+1, N = L+M-1$$

where $\mathbf{X}^{(FLP)}$ consists of the first $M-1$ columns of $\mathbf{X}$ and $\mathbf{x}^{(FLP)}$ is the last column of $\mathbf{X}$.

If the data are noise-free, then the following forward prediction equations are satisfied:

$$\mathbf{X}^{(FLP)}\mathbf{b}^{(FLP)} = \mathbf{x}^{(FLP)} \quad \text{or} \quad \mathbf{X}\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} = \mathbf{0}$$

where $\mathbf{b}^{(FLP)} = [b^{(FLP)}_{M-1}, b^{(FLP)}_{M-2}, \ldots, b^{(FLP)}_1]^T$ contains the forward prediction coefficients, and $z_k, k = 1, 2, \cdots, K$ are the K signal zeros (see below) out of the $M-1$ zeros of the prediction-error filter polynomial

$$B^{(FLP)}(z) = 1 - b^{(FLP)}_1 z^{-1} - \cdots - b^{(FLP)}_{M-1} z^{-(M-1)} \tag{6.1}$$

For noisy data, the exact solution $\mathbf{b}^{(FLP)}$ to $\mathbf{X}^{(FLP)}\mathbf{b}^{(FLP)} = \mathbf{x}^{(FLP)}$ does not exist and must be estimated. The computation of $\mathbf{b}^{(FLP)}$ is based on the minimum norm TLS solution (see e.g. [24] p. 59) of the set $\mathbf{X}\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} \approx \mathbf{0}$

and is derived from the SVD of $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$ in the following way:

$$\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} = \frac{-\overline{\mathbf{V}}_K \overline{\mathbf{V}}_K^H \mathbf{e}_M}{\mathbf{e}_M^T \overline{\mathbf{V}}_K \overline{\mathbf{V}}_K^H \mathbf{e}_M} \tag{6.2}$$

where $\overline{\mathbf{V}}_K$ is the basis of the orthogonal complement of the space spanned by the first K right singular vectors $\mathbf{V}_K$ as defined in Chapter 2, i.e., $\mathbf{V} = [\mathbf{V}_K \ \overline{\mathbf{V}}_K]$, and $\mathbf{e}_M = [0, 0, \ldots, 0, 1]^T \in \mathbb{R}^{M\times 1}$. Note that if $K < M - K$ it is more efficient to work with $\mathbf{V}_K$ via the replacement $\overline{\mathbf{V}}_K \overline{\mathbf{V}}_K^H = \mathbf{I} - \mathbf{V}_K \mathbf{V}_K^H$.

Once the $M-1$ poles are determined from the $M-1$ zeros of the forward prediction error filter polynomial (6.1), the K true poles are separated from the remaining $M-1-K$ extraneous poles. A popular approach for pole selection is to choose K poles based on their location with respect to the unit circle. For example, one can choose the K poles closest to the unit circle if it is known that the poles are undamped or the K poles with smallest moduli if it is known that the poles are damped. However, a more reliable way is to classify poles as true or extraneous based on the energy of the corresponding mode, as used in our computer simulation in Section 6.3.

TLS-BLP [20][21][25]

$$\text{Let} \quad [\mathbf{x}_{L\times 1}^{(BLP)} \ \ \mathbf{X}_{L\times(M-1)}^{(BLP)}] = \mathbf{X}^*, \qquad L \geq K, M \geq K+1, N = L+M-1 \tag{6.3}$$

where $\mathbf{x}_{L\times 1}^{(BLP)}$ is the first column of $\mathbf{X}^*$ and $\mathbf{X}_{L\times(M-1)}^{(BLP)}$ consists of the last $M-1$ columns of $\mathbf{X}^*$.

If the data are noise-free, then the following backward prediction equations are satisfied:

$$\mathbf{X}^{(BLP)}\mathbf{b}^{(BLP)} = \mathbf{x}^{(BLP)} \quad \text{or} \quad [\mathbf{X}^{(BLP)} \ \mathbf{x}^{(BLP)}] \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} = \mathbf{0}$$

where $\mathbf{b}^{(BLP)} = [b_1^{(BLP)}, b_2^{(BLP)}, \ldots, b_{M-1}^{(BLP)}]^T$ contains the backward prediction coefficients, and $(z_k^*)^{-1} = \exp[d_k \Delta t + j(2\pi f_k \Delta t)]$, $k = 1, 2, \cdots, K$ are the K signal zeros out of the $M-1$ zeros of the prediction-error filter polynomial

$$B^{(BLP)}(z) = 1 - b_1^{(BLP)} z^{-1} - \cdots - b_{M-1}^{(BLP)} z^{-(M-1)} \tag{6.4}$$

The inverse in $(z_k^*)^{-1}$ comes from the reversion of prediction coefficients and the conjugation $*$ in $(z_k^*)^{-1}$ comes from the conjugation of data. Note that

$(z_k^*)^{-1} = \exp[d_k \Delta t + j(2\pi f_k \Delta t)]$ is different from z_k only in the sign of the damping factor d_k.

For noisy data, the computation of $\mathbf{b}^{(BLP)}$ is similar to that of $\mathbf{b}^{(FLP)}$ in TLS-FLP, that is, by taking the minimum norm TLS solution of the set $[\mathbf{X}^{(BLP)} \ \mathbf{x}^{(BLP)}] \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} \approx \mathbf{0}$.

Once the $M-1$ poles are determined from the $M-1$ zeros of the backward prediction error filter polynomial (6.4), the poles with negative damping factors are eliminated and the true poles are separated from the extraneous poles by selecting the K peaks with the largest peak energy.

6.2 Prior-knowledge-incorporating estimators

In this section, the incorporation of known signal poles into the TLS-FLP method, known as the LCTLS-FLP method, is first reviewed [79]. Incorporating the same prior knowledge into the other parameter estimators described in Section 2 results in new and better estimators, LCTLS-BLP, HTLS(PK) and HTLN(PK), which are presented thereafter. The abbreviation of PK stands for prior knowledge of known signal poles. Note that the LCTLS-FLP method will be derived in a different way, compared to [5][79], in order to motivate the HTLS(PK) method.

Without loss of generality, we index the p known poles as the first p poles, that is, $z_k, k = 1, \ldots, p$ are known and $z_k, k = p+1, \ldots, K$ are unknown.

LCTLS-FLP [5]

Since $z_k, k = 1, \ldots, p$ are *a priori* exactly known signal poles, they are also the zeros of the forward prediction-error filter polynomial in (6.1), which means

$$z^{M-1} B^{(FLP)}(z)|_{z=z_k} = 0, \quad k = 1, \ldots, p$$

Stacking these p equations together results in a set of p linear constraints which expresses the assumed prior knowledge in terms of known signal poles and forward prediction coefficients:

$$\begin{bmatrix} 1 & z_1 & \cdots & z_1^{M-2} \\ 1 & z_2 & \cdots & z_2^{M-2} \\ \vdots & & & \vdots \\ 1 & z_p & \cdots & z_p^{M-2} \end{bmatrix} \begin{bmatrix} b_{M-1}^{(FLP)} \\ b_{M-2}^{(FLP)} \\ \vdots \\ b_1^{(FLP)} \end{bmatrix} = \begin{bmatrix} z_1^{M-1} \\ z_2^{M-1} \\ \vdots \\ z_p^{M-1} \end{bmatrix}$$

or

$$\mathbf{T}_p^T \begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} = \mathbf{0} \tag{6.5}$$

where $\mathbf{T}_p \in \mathbb{C}^{M\times p}$ is a Vandermonde matrix as defined in (2.7) but composed of only the first p known signal poles.

The incorporation of this prior knowledge into the TLS-FLP method is accomplished by using the above equations as linear constraints. Therefore, the TLS solution $\mathbf{b}^{(FLP)}$ of $\mathbf{X}^{(FLP)}\mathbf{b}^{(FLP)} \approx \mathbf{x}^{(FLP)}$ in the TLS-FLP algorithm is replaced by the linearly constrained TLS, or LCTLS, solution of:

$$\begin{aligned} \mathbf{X}\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} &\approx \mathbf{0} \\ \text{subject to } \mathbf{T}_p^T\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix} &= \mathbf{0} \end{aligned}$$

resulting in the LCTLS-FLP algorithm.

The outline of the LCTLS-FLP algorithm is as follows. First perform a QR decomposition of $\mathbf{T}_p = [\mathbf{Q}_1 \;\; \mathbf{Q}_2]\begin{bmatrix}\mathbf{R}\\ \mathbf{0}\end{bmatrix}$, where $\mathbf{R} \in \mathbb{C}^{p\times p}$ is upper triangular, to find an orthonormal basis $\mathbf{Q}_2 \in \mathbb{C}^{M\times(M-p)}$ of the null space of $\mathbf{T}_p^H$, such that $\mathbf{T}_p^H\mathbf{Q}_2 = \mathbf{0}$. Let $\begin{bmatrix} \mathbf{b}^{(FLP)} \\ -1 \end{bmatrix}^T = \gamma\mathbf{Q}_2^*\hat{\mathbf{b}}^{(FLP)}$, where γ is a nonzero scaling factor that forces the last element to be -1 and $\hat{\mathbf{b}}^{(FLP)}$ is a column vector of dimension $M-p$. Then the constraints hold if and only if $[\mathbf{b}^{(BLP)T} \;\; -1]^T = \gamma\hat{\mathbf{Q}}_2\hat{\mathbf{b}}^{(BLP)}$ for some $\hat{\mathbf{b}}^{(BLP)} \in \mathbb{C}^{(M-p)\times 1}$. In this way, the LCTLS problem is reduced to a classical TLS problem: $(\mathbf{X}\mathbf{Q}_2^*)\hat{\mathbf{b}}^{(FLP)} \approx \mathbf{0}$. Once the TLS solution $\hat{\mathbf{b}}^{(FLP)}$ is computed, the forward prediction coefficients $\mathbf{b}^{(FLP)}$ are obtained, needed for the rooting of polynomial (6.1).

The TLS solution of $(\mathbf{X}\mathbf{Q}_2^*)\hat{\mathbf{b}}^{(FLP)} \approx \mathbf{0}$ can be computed from the SVD. Since $\mathbf{X}\mathbf{Q}_2^*$ has rank $K-p$ if noise-free data are used (refer to Eq. (2.7)):

$$\begin{aligned} \mathbf{X}\mathbf{Q}_2^* &= [\mathbf{S}_p \;\; \mathbf{S}_{K-p}]\begin{bmatrix}\mathbf{C}_p & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_{K-p}\end{bmatrix}[\mathbf{T}_p \;\; \mathbf{T}_{K-p}]^T\mathbf{Q}_2^* \\ &= \mathbf{S}_{K-p}\mathbf{C}_{K-p}\mathbf{T}_{K-p}^T\mathbf{Q}_2^* \end{aligned} \tag{6.6}$$

the null space of $(\mathbf{X}\mathbf{Q}_2^*)$ has rank $(M-p)-(K-p) = M-K$. Typically, the rank of the null space is $M-K > 1$ due to the fact that the LP method is preferably used at M much larger than K, e.g., around $M \approx \frac{3}{4}N$ [21]. Hence for noisy data, taking the SVD of $\tilde{\mathbf{X}} = (\mathbf{X}\mathbf{Q}_2^*) = \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^H$, the solution $\hat{\mathbf{b}}^{(FLP)}$

can be computed as the minimum norm TLS solution

$$\hat{\mathbf{b}}^{(FLP)} = \frac{-\overline{\tilde{\mathbf{V}}}_{K-p}\overline{\tilde{\mathbf{V}}}_{K-p}^H\mathbf{e}_{M-p}}{\mathbf{e}_{M-p}^T\overline{\tilde{\mathbf{V}}}_{K-p}\overline{\tilde{\mathbf{V}}}_{K-p}^H\mathbf{e}_{M-p}} \tag{6.7}$$

where $\overline{\tilde{\mathbf{V}}}_{K-p}$ is the basis of the orthogonal complement of the space spanned by the first $K-p$ right singular vectors $\tilde{\mathbf{V}}_{K-p}$, i.e., $\tilde{\mathbf{V}} = [\tilde{\mathbf{V}}_{K-p} \;\; \overline{\tilde{\mathbf{V}}}_{K-p}]$, and $\mathbf{e}_{M-p} = [0,0,\ldots,0,1]^T \in \mathbb{R}^{(M-p)\times 1}$. If $K-p < M-K$, it is more efficient to work with $\tilde{\mathbf{V}}_{K-p}$ via the replacement $\overline{\tilde{\mathbf{V}}}_{K-p}\overline{\tilde{\mathbf{V}}}_{K-p}^H = \mathbf{I} - \tilde{\mathbf{V}}_{K-p}\tilde{\mathbf{V}}_{K-p}^H$.

Algorithm 9

Outline of the LCTLS-FLP algorithm

Given. *A model-order estimate K, p known signal poles arranged in an $M \times p$ Vandermonde matrix $\mathbf{T}_p$ as in (6.5), and N uniformly sampled data points x_n, $n = 0,\cdots,N-1$, arranged in an $L \times M$ Hankel matrix $\mathbf{X}$ as in (2.1).*

Step 1. *Perform a QR decomposition of $\mathbf{T}_p = [\mathbf{Q}_1 \;\; \mathbf{Q}_2][\mathbf{R}^T \;\; \mathbf{0}]^T$, where $[\mathbf{Q}_1 \;\; \mathbf{Q}_2]$ is unitary, $\mathbf{Q}_1 \in \mathbb{C}^{M\times p}$, $\mathbf{Q}_2 \in \mathbb{C}^{M\times(M-p)}$ and $\mathbf{R} \in \mathbb{C}^{p\times p}$ is upper triangular.*

Step 2. *Compute the minimum norm TLS solution $\hat{\mathbf{b}}^{(FLP)}$ of*

$$(\mathbf{X}\mathbf{Q}_2^*)\hat{\mathbf{b}}^{(FLP)} \approx \mathbf{0}$$

via the SVD of $\mathbf{X}\mathbf{Q}_2^*$.

Step 3. *Compute the vector of forward prediction coefficients $\mathbf{b}^{(FLP)}$ from $[\mathbf{b}^{(FLP)} \;\; -1]^T = \gamma\mathbf{Q}_2^*\hat{\mathbf{b}}^{(FLP)}$ by calculating the appropriate scaling factor γ.*

Step 4. *Root the polynomial (6.1) using the coefficients $\mathbf{b}^{(FLP)}$ in order to find the $M-1$ poles, and extract the signal poles $z_k, k = 1,2,\ldots,K$ with the largest peak energy. The nonlinear parameters f_k, d_k, $k = 1,\cdots,K$ are found straight-forwardly using the relation $z_k = e^{(-d_k+j2\pi f_k)/f_s}$.*

Step 5. *Apply the linear-parameter-estimation part, Step 3, of the HTLS Algorithm 1 to the original data x_n, $n = 0,\cdots,N-1$ using the extracted K signal poles, to find the corresponding a_k and φ_k, $k = 1,\cdots,K$.*

LCTLS-BLP

The LCTLS-BLP algorithm is developed similarly to the LCTLS-FLP algorithm. The exactly known signal poles $z_k, k = 1, \ldots, p$ are the zeros of the backward-prediction-error filter polynomial in (6.4), which means:

$$B^{(BLP)}(z)|_{z=(z_k^*)^{-1}} = 0, \quad k = 1, \ldots, p$$

Stacking these p equations together results in a set of p linear constraints which expresses the assumed prior knowledge in terms of known signal poles and backward prediction coefficients:

$$\begin{bmatrix} z_1^* & z_1^{*2} & \cdots & z_1^{*(M-1)} \\ z_2^* & z_2^{*2} & \cdots & z_2^{*(M-1)} \\ \vdots & & & \vdots \\ z_p^* & z_p^{*2} & \cdots & z_p^{*(M-1)} \end{bmatrix} \begin{bmatrix} b_1^{(BLP)} \\ b_2^{(BLP)} \\ \vdots \\ b_{M-1}^{(BLP)} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

or

$$\hat{\mathbf{T}}_p^H \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} = \mathbf{0} \tag{6.8}$$

where $\hat{\mathbf{T}}_p$ permutes the first row of $\mathbf{T}_p$ in (6.5) to the bottom row.

This prior knowledge is incorporated into the TLS-BLP method by using the above equations as constraints. This is done by replacing the TLS solution $\mathbf{b}^{(BLP)}$ of $\mathbf{X}^{(BLP)}\mathbf{b}^{(BLP)} \approx \mathbf{x}^{(BLP)}$ in the TLS-BLP algorithm by the LCTLS solution of:

$$\begin{bmatrix} \mathbf{X}^{(BLP)} & \mathbf{x}^{(BLP)} \end{bmatrix} \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} \approx \mathbf{0} \tag{6.9}$$

$$\text{subject to } \hat{\mathbf{T}}_p^H \begin{bmatrix} \mathbf{b}^{(BLP)} \\ -1 \end{bmatrix} = \mathbf{0} \tag{6.10}$$

resulting in the LCTLS-BLP algorithm.

Algorithm 10

Outline of the LCTLS-BLP algorithm

Given. *A model-order estimate K, p known signal poles arranged in an $M \times p$ matrix $\hat{\mathbf{T}}_p$ as in (6.8), and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel matrix $[\mathbf{x}^{(BLP)} \ \mathbf{X}^{(BLP)}]$ as in (6.3) and permuted to $[\mathbf{X}^{(BLP)} \ \mathbf{x}^{(BLP)}]$.*

Step 1. *Perform a QR decomposition of $\hat{\mathbf{T}}_p = [\hat{\mathbf{Q}}_1 \ \hat{\mathbf{Q}}_2][\hat{\mathbf{R}}^T \ \mathbf{0}]^T$, where $[\hat{\mathbf{Q}}_1 \ \hat{\mathbf{Q}}_2]$ is unitary, $\hat{\mathbf{Q}}_1 \in \mathbb{C}^{M \times p}$, $\hat{\mathbf{Q}}_2 \in \mathbb{C}^{M \times (M-p)}$ and $\hat{\mathbf{R}} \in \mathbb{C}^{p \times p}$ is upper triangular.*

Step 2. *Compute the minimum norm TLS solution $\hat{\mathbf{b}}^{(BLP)}$ of*

$$([\mathbf{X}^{(BLP)} \ \mathbf{x}^{(BLP)}]\hat{\mathbf{Q}}_2)\hat{\mathbf{b}}^{(BLP)} \approx \mathbf{0}$$

via the SVD of $([\mathbf{X}^{(BLP)} \ \mathbf{x}^{(BLP)}]\hat{\mathbf{Q}}_2)$.

Step 3. *Compute the vector of backward prediction coefficients $\mathbf{b}^{(BLP)}$ from $[\mathbf{b}^{(BLP)} \ -1]^T = \gamma \hat{\mathbf{Q}}_2 \hat{\mathbf{b}}^{(BLP)}$ by calculating the appropriate scaling factor γ.*

Step 4. *Root the polynomial (6.4) using the coefficients $\mathbf{b}^{(BLP)}$ in order to find the $M-1$ poles, and extract the signal poles z_k with positive damping factors d_k, $k = 1, \cdots, K$. The nonlinear parameters f_k, d_k, $k = 1, \cdots, K$ are found straight-forwardly using the relation $z_k = e^{(-d_k + j2\pi f_k)/f_s}$.*

Step 5. *Apply the linear-parameter-estimation part, Step 3, of the HTLS Algorithm 1 to the original data x_n, $n = 0, \cdots, N-1$ using the extracted K signal poles, to find the corresponding a_k and φ_k, $k = 1, \cdots, K$.*

HTLS(PK)

For noise-free data, we can remove the known part in the Vandermonde decomposition (2.7) of the original Hankel data matrix $\mathbf{X}$ by projecting the rows of $\mathbf{X}$ onto the orthogonal complement of the column space of $\mathbf{T}_p$, i.e.,

$$\begin{aligned} \tilde{\mathbf{X}} &= \mathbf{X}\mathbf{Q}_2^* \\ &= (\mathbf{S}_p\mathbf{C}_p\mathbf{T}_p^T + \mathbf{S}_{K-p}\mathbf{C}_{K-p}\mathbf{T}_{K-p}^T)\mathbf{Q}_2^* \\ &= \mathbf{S}_{K-p}\mathbf{C}_{K-p}\mathbf{T}_{K-p}^T\mathbf{Q}_2^* \end{aligned}$$

as shown in (6.6). The multiplication of $\mathbf{X}$ with $\mathbf{Q}_2^*$ from the right destroys the shift-invariant property in the row space but keeps the shift-invariant property in the column space, that is,

$$\mathbf{S}_{K-p}^{\uparrow} = \mathbf{S}_{K-p\downarrow}\mathbf{Z}_{K-p}$$

is still satisfied, where $\mathbf{Z}_{K-p}$ is a diagonal matrix composed of the $K-p$ unknown signal poles. On the other hand, we have $\tilde{\mathbf{X}} = \tilde{\mathbf{U}}_{K-p}\tilde{\mathbf{\Sigma}}_{K-p}\tilde{\mathbf{V}}_{K-p}^H$,

where $\tilde{\mathbf{U}}_{K-p}$ contains the first $K-p$ left singular vectors of $\tilde{\mathbf{X}}$. Hence,

$$\tilde{\mathbf{U}}^{\uparrow}_{K-p} = \tilde{\mathbf{U}}_{(K-p)\downarrow}\mathbf{Z}^{(\tilde{U})}_{K-p}$$

where $\mathbf{Z}^{(\tilde{U})}_{K-p}$ is equal to $\mathbf{Z}_{K-p}$ up to a similarity transform. Therefore, the $K-p$ unknown poles are the eigenvalues of $\mathbf{Z}^{(\tilde{U})}_{K-p}$.

Algorithm 11

Outline of the HTLS(PK) algorithm

Given. A model-order estimate K, p known signal poles arranged in an $M \times p$ Vandermonde matrix $\mathbf{T}_p$ as in (6.5), and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel matrix $\mathbf{X}$ as in (2.1).

Step 1: (QRD). Perform a QR decomposition of $\mathbf{T}_p = [\mathbf{Q}_1 \;\; \mathbf{Q}_2][\mathbf{R}^T \;\; 0]^T$, where $[\mathbf{Q}_1 \;\; \mathbf{Q}_2]$ is unitary, $\mathbf{Q}_1 \in \mathbb{C}^{M\times p}$, $\mathbf{Q}_2 \in \mathbb{C}^{M\times(M-p)}$ and $\mathbf{R} \in \mathbb{C}^{p\times p}$ is upper triangular.

Step 2: (SVD). Compute a truncated SVD of $\mathbf{X}\mathbf{Q}_2^* = \tilde{\mathbf{U}}_{K-p}\tilde{\mathbf{\Sigma}}_{K-p}\tilde{\mathbf{V}}^H_{K-p}$.

Step 3: (TLS). Compute the TLS solution $\mathbf{Z}^{(\tilde{U})}_{K-p}$ of $\tilde{\mathbf{U}}^{\uparrow}_{K-p} \approx \tilde{\mathbf{U}}_{(K-p)\downarrow}\mathbf{Z}^{(\tilde{U})}_{K-p}$ and extract the signal poles $z_k, k = p+1, p+2, \ldots, K$ from the eigenvalues of $\mathbf{Z}^{(\tilde{U})}_{K-p}$ to find f_k and d_k, $k = p+1, p+2, \cdots, K$.

Step 4: (LS). Apply the linear-parameter-estimation part, Step 3, of the HTLS Algorithm 1 to the original data x_n, $n = 0, \cdots, N-1$, using the known and estimated signal poles $z_k, k = 1, 2, \ldots, p, p+1, \ldots, K$ to find a_k and φ_k, $k = 1, 2, \cdots, K$.

An alternative way is to find an orthogonal basis $\mathbf{Q}_2$ from the QR decomposition of $\mathbf{S}_p$ in (6.6), i.e., $\mathbf{S}_p = [\mathbf{Q}_1 \;\; \mathbf{Q}_2]\begin{bmatrix}\mathbf{R}\\ 0\end{bmatrix}$. Then the known poles are removed via left multiplication of $\mathbf{Q}_2^H$ with $\mathbf{X}$, i.e., $\tilde{\mathbf{X}} = \mathbf{Q}_2^H\mathbf{X}$. In this case, the shift-invariant property in the row space

$$\tilde{\mathbf{V}}^{*\uparrow}_{K-p} = \tilde{\mathbf{V}}^{*}_{(K-p)\downarrow}\mathbf{Z}^{(\tilde{V})}_{K-p}$$

can be used, and the unknown signal poles are the eigenvalues of $\mathbf{Z}^{(\tilde{V})}_{K-p}$.

HTLN(PK)

Prior knowledge is incorporated by adding constraints to the STLN problem $\mathbf{AY} \approx \mathbf{B}$, i.e.,

$$\begin{aligned} [\mathbf{A} \;\; \mathbf{B}]\begin{bmatrix}\mathbf{Y}\\ -\mathbf{I}\end{bmatrix} &\approx \mathbf{0} \\ \text{subject to} \quad \mathbf{T}_p^T\begin{bmatrix}\mathbf{y}_1\\ -1\end{bmatrix} &= \mathbf{0} \end{aligned} \tag{6.11}$$

where $\mathbf{y}_1$ is the forward prediction coefficient vector, or, the first column of $\mathbf{Y}$.

When $M = K+1$, i.e., $\mathbf{B}$ has only one column, $\mathbf{b}$, the constrained problem is simply

$$\begin{bmatrix}\mathbf{A} \;\; \mathbf{b}\\ \mathbf{T}_p^T\end{bmatrix}\begin{bmatrix}\mathbf{y}\\ -1\end{bmatrix} \approx \mathbf{0}$$

where $[\mathbf{A} \;\; \mathbf{b}]$ is affected by errors and has a Hankel structure, while $\mathbf{T}_p$ is error-free and fixed. This can be solved using the STLN algorithm so that the correction matrix applied to $[\mathbf{A} \;\; \mathbf{b}]$ is Hankel and the correction matrix applied to $\mathbf{T}_p^T$ is $\mathbf{0}$.

Recall that in Chapter 5 the two STLN algorithms for solving Hankel structured problems are described. However, the data matrix here is not purely Hankel structured. An easy solution is to add the constraints $\mathbf{T}_p^T\begin{bmatrix}\mathbf{y}\\ -1\end{bmatrix} = \mathbf{0}$ to the constraints $\mathbf{r}(\mathbf{z}+\Delta\mathbf{z}) = \mathbf{0}$ or $\hat{\mathbf{B}}\Delta\mathbf{z} - \hat{\mathbf{d}} = \mathbf{0}$ in (5.10) so that the STLN Algorithm-2 can be used after minor modifications of Step 2(a) as shown below:

Modified STLN Algorithm-2

Apply STLN Algorithm-2 in which Step 2(a) is modified as follows:

Step 2 (a).

$$\hat{\mathbf{A}} := \begin{bmatrix}\mathbf{D}_\alpha & 0 & 0\\ 0 & \mathbf{D}_\beta & 0\end{bmatrix}, \hat{\mathbf{B}} := \begin{bmatrix}\mathbf{P}_\alpha - \check{\mathbf{Y}} & \mathbf{P}_\beta & -(\mathbf{A}+\Delta\mathbf{A})\\ 0 & 0 & \mathbf{P}_p\end{bmatrix},$$

$$\Delta\mathbf{z} := \begin{bmatrix}\Delta\alpha\\ \Delta\beta_0\\ \Delta\mathbf{y}\end{bmatrix}, \hat{\mathbf{b}} := -\begin{bmatrix}\mathbf{D}_\alpha\alpha\\ \mathbf{D}_\beta\beta_0\end{bmatrix} \hat{\mathbf{d}} = \begin{bmatrix}-\mathbf{r}\\ \mathbf{p}_p\end{bmatrix}.$$

$\mathbf{P}_p$ and $\mathbf{p}_p$ are, respectively, the first $M-1$ columns and the last column of

$\mathbf{T}_p^T$, i.e., $\mathbf{T}_p^T = [\mathbf{P}_p, \mathbf{p}_p]$.

When $M > K+1$, the constrained problem can be written in the following form via the vectorization as done in (5.15):

$$\begin{bmatrix} \mathbf{A} & 0 & \cdots & 0 & \mathbf{b}_1 \\ 0 & \mathbf{A} & \cdots & 0 & \mathbf{b}_2 \\ \vdots & & \ddots & & \vdots \\ 0 & \cdots & 0 & \mathbf{A} & \mathbf{b}_{(M-K)} \\ \mathbf{P}_p & & \mathbf{0} & & \mathbf{p}_p \end{bmatrix} \begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \\ \vdots \\ \mathbf{y}_{(M-K)} \\ -1 \end{bmatrix} \approx \mathbf{0}$$

In this way, the problem can also be solved by the STLN algorithm which preserves the special structure of the data matrix.

Algorithm 12

Outline of the HTLN(PK) algorithm

Given. *A model-order estimate K, p known signal poles arranged in an $M \times p$ Vandermonde matrix $\mathbf{T}_p$ as in (6.5), and N uniformly sampled data points x_n, $n = 0, \cdots, N-1$, arranged in an $L \times M$ Hankel matrix $\mathbf{X} = [\mathbf{A}_{L\times K} \;\; \mathbf{B}_{L\times(M-K)}]$.*

Step 1. *Apply the modified STLN Algorithm-2 to (6.11) in order to find the Hankel correction matrix $[\mathbf{E} \;\; \mathbf{F}]$ and STLN solution $\mathbf{Y}$ such that*

$$[\mathbf{A}+\mathbf{E} \;\; \mathbf{B}+\mathbf{F}] \begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix} = \mathbf{0} \text{ and } \mathbf{T}_p^T \begin{bmatrix} \mathbf{y}_1 \\ -1 \end{bmatrix} = \mathbf{0}$$

Step 2. *Compute a QR decomposition of*

$$\begin{bmatrix} \mathbf{Y} \\ -\mathbf{I} \end{bmatrix} = [\mathbf{Q}_n \;\; \mathbf{Q}_s] \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}$$

where $[\mathbf{Q}_n \;\; \mathbf{Q}_s]$ is unitary, $\mathbf{R}$ is upper triangular.

Step 3. *Compute the TLS solution $\mathbf{Z}^{(Q)}$ of the following set of equations:*

$$\mathbf{Q}_s^{*\uparrow} \approx \mathbf{Q}_{s\downarrow}^{*} \mathbf{Z}^{(Q)}$$

and retrieve the eigenvalues z_k of $\mathbf{Z}^{(Q)}$. The nonlinear parameters f_k, d_k, $k = 1, \cdots, K$ are found straight-forwardly using the relation $z_k = e^{(-d_k + j2\pi f_k)/f_s}$.

Step 4. *Apply the linear-parameter-estimation part, Step 3, of the HTLS Algorithm 1 to the original data x_n, $n = 0, \cdots, N-1$ using the extracted signal poles, to find the corresponding a_k and φ_k, $k = 1, 2, \cdots, K$.*

All the prior-knowledge-incorporating estimators, namely, LCTLS-FLP, LCTLS-BLP, HTLS(PK) and HTLN(PK) presented above, are computationally more efficient than their corresponding estimator TLS-FLP, TLS-BLP, HTLS and HTLN, respectively. This difference in computational efficiency is more pronounced as the number p of *a priori* known poles increases. This is because the former estimators only need to compute the SVD of smaller sized matrices, as compared to the latter estimators, and the SVD is computationally the most time-consuming part of each method. For example, the LCTLS-FLP method is more efficient than the TLS-FLP method since the SVD in LCTLS-FLP is performed on the smaller matrix $\mathbf{X}\mathbf{Q}_2^*$ which has p columns fewer than the matrix $\mathbf{X}$ as used in the TLS-FLP method. The savings for the LCTLS-BLP and HTLS methods are similar. As for HTLN(PK), the constraints expressing prior knowledge reduce the number of iterations required for convergence.

6.3 Simulation results

We study and compare the performance of different estimation algorithms, HTLS, LCTLS-FLP, LCTLS-BLP, HTLS(PK) and HTLN(PK), in terms of respectively resolution and parameter accuracy. In order to obtain good parameter estimates, the number of columns M of the data matrix used in the LP, HTLS and HTLN estimators is respectively $M \approx \frac{2}{3}N$, $M \approx \frac{1}{2}N$ and $M = K+1$, unless specified otherwise. The extraction of the signal peaks in the LP methods is done by selecting the K peaks with the largest peak energy, which is defined as $\sum_{n=0}^{N-1} a_k^2 |z_k|^{2n} = a_k^2(1-|z_k|^{2N})/(1-|z_k|^2)$ for the k-th peak [80]. The maximum number of iterations in the iterative method HTLN is set to 100. The tolerance ϵ in the stop criterion of the STLN iterations (5.14) for HTLN is $\epsilon = 10^{-10}$. If not specified, all the HTLN runs are convergent within 100 iterations.

Example 1: *Signal-1*

In each algorithm, $N = 25$ data samples are taken, and the parameters $f_1 = 0.2$ Hz and $d_1 = 0.01$ rad/s of the first peak of *Signal-1* are assumed to be known *a priori*.

method	Bias±standard deviation	
	f_2 (Hz)	d_2 (rad/s)
HTLS	-0.0001 ± 0.0058	-0.0049 ± 0.0689
LCTLS-FLP	-0.0001 ± 0.0025	0.0060 ± 0.0147
LCTLS-BLP	0.0002 ± 0.0026	0.0040 ± 0.0130
HTLS(PK)	0.0000 ± 0.0020	0.0013 ± 0.0124
HTLN(PK)	0.0000 ± 0.0018	0.0011 ± 0.0120
method	Bias±standard deviation	
	a_2 (a.u.)	ψ_2 (°)
HTLS	0.1704 ± 0.7293	-2.2126 ± 46.7598
LCTLS-FLP	0.0944 ± 0.2254	-2.2586 ± 11.6442
LCTLS-BLP	0.0464 ± 0.1730	-2.2695 ± 9.9867
HTLS(PK)	0.0195 ± 0.1532	-0.7168 ± 8.1088
HTLN(PK)	0.0147 ± 0.1491	-0.8432 ± 7.6712

Table 6.1: Example 1: Bias ± standard deviation estimates of the parameters of the unknown peak at SNR=15dB, $N = 25$. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $M = 13, 17, 3$ is used respectively for the HTLS method, HTLS(PK) method, LP-based methods and the HTLN(PK) method.

Fig. 6.1(a) shows the failure rate of each method. Since the two peaks of this simulated signal are so close that their spectra overlap, all the methods incorporating prior knowledge of f_1, d_1 of the first peak have much fewer failures than the HTLS method. The LCTLS-FLP and LCTLS-BLP methods are comparable in resolution in this example and reduce the failure rate of HTLS with a factor 4. The HTLS(PK) and HTLN(PK) methods in turn reduce the failure rate of LCTLS-FLP by a factor of at least 10, since both methods have only one failure at SNR= 10dB ($\sigma_\nu \approx 0.2236$) and no failures at higher SNR's.

In each of the accuracy plots, Fig. 6.1(b) and (c), two full lines are added: the lower and upper one representing respectively the CR lower bounds with and without prior knowledge of f_1 and d_1 [81]. It can be clearly seen that the HTLS method produces the largest RMSE values which closely follow the

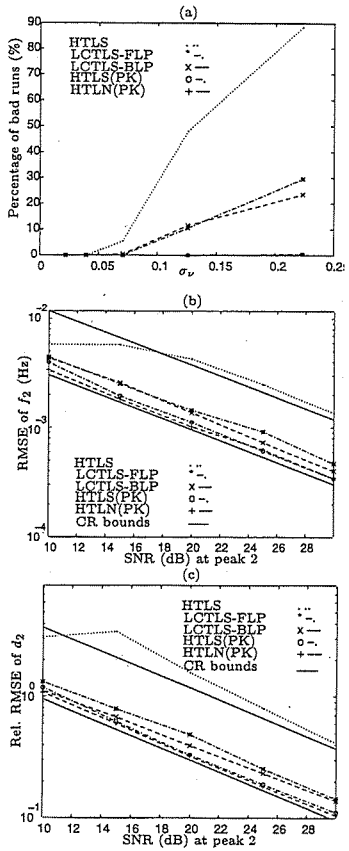


Figure 6.1: Example 1: (a) Failure rate versus σ_ν. (b) RMSE of $f_2 = 0.22$ Hz and (c) Relative RMSE of $d_2 = 0.02$ rad/s versus the SNR at this peak. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$. $M = 13, 17, 3$ is used respectively for the HTLS method, HTLS(PK) method, LP-based methods and the HTLN(PK) method. HTLN(PK) has 2 non-convergent cases within 100 iterations of the STLN Algorithm-2 at SNR=10dB.

upper full line. The prior-knowledge-incorporating method, LCTLS-FLP, reduces the RMSE values of the HTLS frequency and damping factor estimates with a factor 3 to 4. The accuracy of these LCTLS-FLP estimates is further improved with a factor 1.2, 1.5 and 1.6, respectively, by using the LCTLS-BLP, HTLS(PK) and HTLN(PK) methods. Among these, only the HTLS(PK) method and the HTLN(PK) method approach the lower full line. A more careful examination indicates that the HTLN(PK) method achieves the lowest RMSE values, and especially, the lowest standard deviations as listed in Table 6.1.

methods	SNR=25dB	SNR=20dB	SNR=15dB
HTLS	1.19e+05	1.21e+05	1.23e+05
LCTLS-FLP	9.06e+05	9.06e+05	9.00e+05
LCTLS-BLP	4.29e+05	4.29e+05	4.30e+05
HTLS(PK)	1.19e+05	1.19e+05	1.20e+05
HTLN(PK)	3.45e+07	4.07e+07	5.08e+07

Table 6.2: Example 1: Number of flops required by different methods using Matlab averaged over 200 runs. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$. $M = 13, 17, 3$ is used respectively for the HTLS method, the HTLS(PK) method, LP-based methods and the HTLN(PK) method.

Table 6.2 compares the computational efficiency of the proposed algorithms in terms of the number of flops averaged over 200 runs required by each method when implemented using Matlab. Obviously, the iterative method, HTLN(PK), requires the largest number of flops compared to the other SVD-based methods while HTLS(PK) needs the smallest number of flops. Observe that the improvement in computational efficiency of HTLS(PK) compared to HTLS is not large because only prior knowledge of *one* peak is incorporated. The number of flops required by the HTLN(PK) method increases when the SNR is lowered, because the number of required iterations up to convergence increases. In all examples we studied, the improvement in accuracy of the HTLN(PK) estimates compared to that of HTLS(PK) estimates is not large enough to justify the large increase in computational load, especially when the number of data samples is large, and therefore the HTLN(PK) method is not considered in the following study.

Fig. 6.2 shows the failure rate of each method versus varying number of columns M of the data matrix. In LP methods, $M - 1$ is actually the prediction length. It is clear from this failure rate plot that the prior-knowledge-

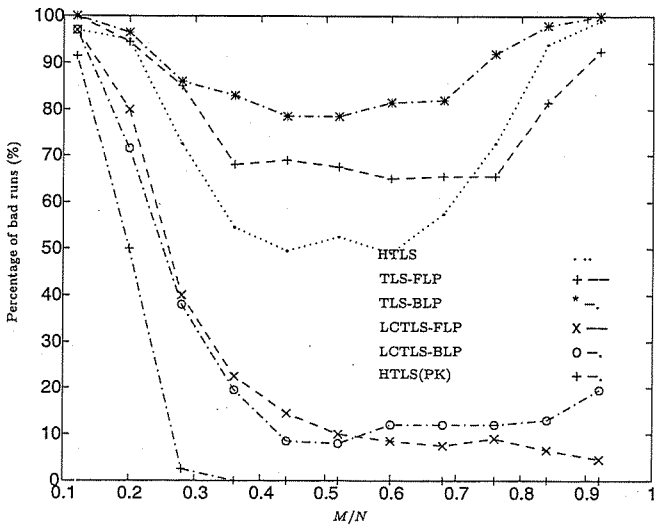


Figure 6.2: Example 1: Failure rate of each method versus M/N of the data matrix. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$, SNR=15dB or $\sigma_\nu = 0.1257$.

incorporating methods, LCTLS-FLP, LCTLS-BLP and HTLS(PK), give a wider minimum than the corresponding no-prior-knowledge-incorporating methods, TLS-FLP, TLS-BLP and HTLS, respectively. The same observation holds for the accuracy plots, which are not included. Hence the prior-knowledge-incorporating methods are not sensitive to the choice of M in this example. In particular, the region of constant failure rate and accuracy is as wide as $0.35 \leq M/N \leq 0.9$ for HTLS(PK) in contrast to $0.4 \leq M/N \leq 0.6$ for HTLS. In the following study, we will use $M \approx \frac{2}{3}N$ for the LCTLS-FLP and LCTLS-BLP methods, and $M \approx \frac{1}{2}N$ for the HTLS and HTLS(PK) methods, unless otherwise specified.

If, instead, the pole of the second peak is known *a priori*, the results are very similar and therefore omitted here.

Example 2: *Signal-2*

In each algorithm, $N = 128$ data samples are taken, and the parameters $f_4 = 353$ Hz and $d_4 = 117$ rad/s of the fourth peak of *Signal-2* are assumed to be known *a priori*.

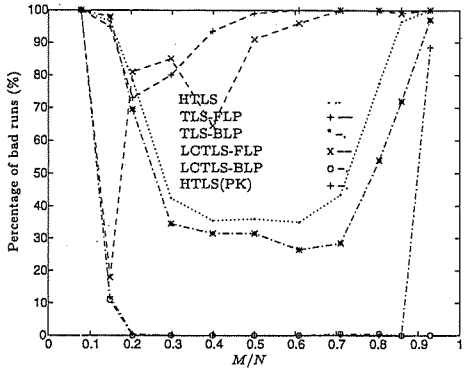


Figure 6.3: Example 2: Failure rate of each method versus M/N of the data matrix. $f_4 = 353$ Hz, $d_4 = 117$ rad/s are known *a priori*. $N = 128$, $\sigma_\nu = 2$.

If we vary the number of columns M used in each method as done for *Signal-1*, the results shown in Fig. 6.3 are similar to Fig. 6.2, except that the TLS-FLP and LCTLS-FLP methods have sharp minima around $M = 20$, instead of $M \approx \frac{2}{3}N \approx 85$. The reason is the following. When the length of the prediction filter is very large, say, around 85, and the noise is wildly fluctuating, some noise peaks are fitted as extraneous peaks with large peak energy. Thus the extraction of $K = 5$ signal poles from 85 fitted poles is rather difficult. Therefore, $M = 20$ will be used in the LCTLS-FLP method for this signal. However, $M \approx \frac{2}{3}N$ remains unchanged in the LCTLS-BLP method, since it computes all the extraneous peaks with negative damping factors, which are easily identified and eliminated.

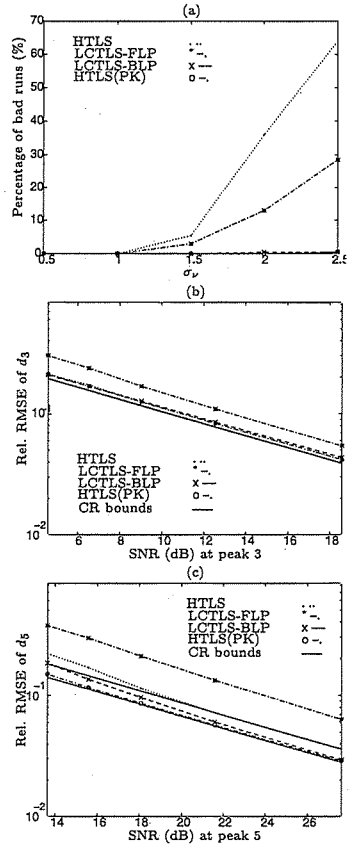


Figure 6.4: Example 2: (a) Failure rate versus σ_ν. Relative RMSE of the estimates of (b) $d_3 = 197$ rad/s and (c) $d_5 = 808$ rad/s versus the peak SNR. $f_4 = 353$ Hz, $d_4 = 117$ rad/s are known *a priori*. $N = 128$. $M = 64, 20, 85$ is used respectively for the HTLS, HTLS(PK), LCTLS-FLP and LCTLS-BLP methods.

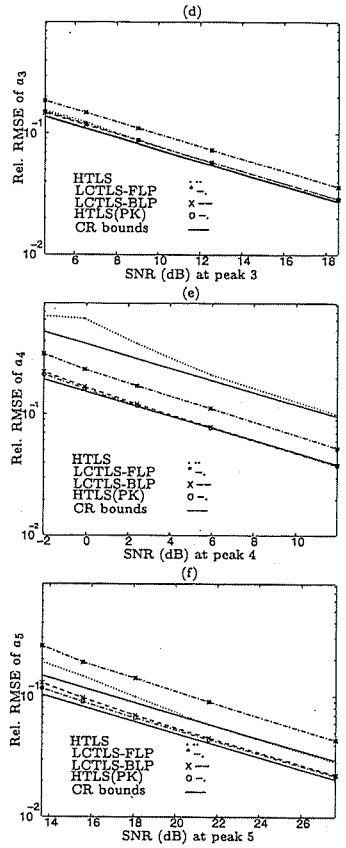


Figure 6.4: (cont.) Example 2: Relative RMSE of (d) $a_3 = 6.0$ a.u., (e) $a_4 = 2.8$ a.u., and (f) $a_5 = 17.0$ a.u. versus the peak SNR.

Fig. 6.4(a) gives the plot of the failure rate of the HTLS, LCTLS-FLP, LCTLS-BLP and HTLS(PK) methods versus the noise level σ_ν, which is similar to that of Example 1 (Fig. 6.1(a)), except that the LCTLS-BLP and HTLS(PK) methods are now comparable, with hardly any failure. However, the accuracy plots as shown in Fig. 6.4(b)~(f) are different from those of Example 1. Although $M = 20$ is used in LCTLS-FLP, the computed RMSE values are still too high due to un-eliminated extraneous peaks. The best accuracy is obtained by the HTLS(PK) and LCTLS-BLP methods, both approaching the CR lower bounds with prior knowledge. Actually, the accuracy of the HTLS(PK) estimates is slightly better than that of the LCTLS-BLP estimates.

Note that the gap between the CR lower bounds with and without prior knowledge is largest for a_4, obvious for a_5 and d_5, but hardly observable for a_3 and d_3. This is because peak 4 is the peak with known poles, peak 3 is far away from this known peak, but peak 5 is overlapping with it. The gap between the two CR bounds depends mainly on this difference in frequency. Hence, the incorporation of prior knowledge of known signal poles only improves resolution and parameter accuracy when the corresponding peaks are close to or even overlapping with the unknown peaks.

6.4 Conclusion

In parameter estimation of closely-spaced sinusoids, prior knowledge of some known signal poles can be incorporated, as done in the LCTLS-FLP algorithm proposed by Dowling et al. and based on forward prediction and total least squares. In this chapter, three better algorithms are presented, namely LCTLS-BLP, HTLS(PK) and HTLN(PK), derived respectively from the TLS-BLP, HTLS, and HTLN algorithms. These methods incorporate the same prior knowledge but clearly outperform the LCTLS-FLP method in resolution and parameter accuracy. Computer simulations indicate that the new methods, especially HTLS(PK) and HTLN(PK), improve the LCTLS-FLP method significantly, with a factor of at least 40 in resolution and 1.2 in accuracy. The accuracy of the HTLN(PK) method is slightly better than that of the HTLS(PK) method since it gives rise to the lowest RMSE values, and especially, the lowest standard deviations of the estimated parameters. However, HTLS(PK) is shown to be computationally the most efficient method. The improvement in accuracy of the HTLN(PK) estimates compared to that of HTLS(PK) estimates is not large enough to justify the large increase in computational load, especially when the number of data samples is large. Therefore, the HTLS(PK)

method is strongly recommended for exponential data fitting in the presence of known signal components. More analysis of the HTLS(PK) method and its application to NMR spectroscopy is given in the next chapter.

Chapter 7

More analysis of HTLS(PK) with application to NMR

The HTLS(PK) estimation method has been presented in the previous chapter and shown to be superior to other prior-knowledge-incorporating estimation techniques. In this chapter, we interpret the incorporation of prior knowledge in the HTLS(PK) method in terms of a prefiltering operation discussed in Chapter 3, which enables an intuitive analysis in the frequency domain. Subsequently, more results of quantification of simulated signals using HTLS(PK) are presented, including a performance comparison of HTLS(PK) and prefiltered HTLS, enhancement of the HTLS(PK) method, a stepwise efficiency comparison of HTLS(PK) and HTLS, and a sensitivity study of HTLS(PK) with respect to the accuracy of the given poles. Finally, the HTLS(PK) method is applied to a real-world NMR signal and compared with the HTLS estimator.

7.1 HTLS(PK) and prefiltering

Comparing $\bar{\mathbf{X}} = \mathbf{X}\mathbf{Q}_2^*$ performed in the HTLS(PK) method and $\bar{\mathbf{X}} = \mathbf{X}\mathbf{H}^T$ performed in the prefiltered HTLS method in Section 3.3, it is easy to see that $\mathbf{Q}_2^H$ is functioning as a filter matrix $\mathbf{H}$ discussed in Chapter 3. This enables us to examine the incorporation of prior knowledge in terms of a prefiltering operation.

A characteristic of this prefiltering operation is that it annihilates the p Vandermonde vectors, $\mathbf{t}_i, i = 1, 2, \ldots, p$, which consist of known signal poles $z_i, i = 1, 2 \ldots, p$ as defined in Definition 1. By annihilation, we mean that the filtered Vandermonde vectors $\mathbf{t}_i$ of p exponentials are all zeros, that is,

$$\mathbf{t}_i^T \mathbf{Q}_2^* = \mathbf{0}, \qquad i = 1, 2, \ldots, p$$

or

$$\mathbf{T}_p^T \mathbf{Q}_2^* = \mathbf{0}$$

As pointed out in [5], this implies a filtering operation via an FIR filter with p zeros at known signal poles. Let us examine such an FIR filtering in terms of a prefiltering operation discussed in Chapter 3. Suppose the z-transfer function of this FIR filter is

$$\begin{aligned} \mathsf{H}(\mathsf{z}) &= \mathsf{z}^{-p}(\mathsf{z} - z_1)(\mathsf{z} - z_2)\cdots(\mathsf{z} - z_p) \\ &= h_1 + h_2\mathsf{z}^{-1} + h_3\mathsf{z}^{-2} + \ldots + h_{p+1}\mathsf{z}^{-p} \end{aligned}$$

Let $q = p + 1$. We can perform a discrete convolution of the input data $x_1, x_2, \ldots, x_M, 0, \ldots, 0$ and the filter impulse response $h_1, h_2, \ldots, h_q, 0, \ldots, 0$ with zero-padding. As discussed in Section 3.2 of Chapter 3, several structures of the filter matrix are available. Here we take the form of filter matrix in the inner most box of (3.2) without distortion, i.e.

$$\mathbf{H}_{\mathrm{PK}} = \begin{bmatrix} h_q & h_{q-1} & \cdots & \cdots & h_1 & 0 & 0 \\ 0 & \ddots & \ddots & & & \ddots & 0 \\ 0 & 0 & h_q & h_{q-1} & \cdots & \cdots & h_1 \end{bmatrix} \in \mathbb{C}^{(M-q+1)\times M}$$

$$\text{where } q = p+1 \qquad (7.1)$$

Such a filter matrix generates filtered output without distortion in the absence of noise and has the same dimension as $\mathbf{Q}_2^H$.

In this way, the filtered Vandermonde vectors, $\mathbf{t}_i, i = 1, 2, \ldots, p$, of the p exponentials via the filter matrix $\mathbf{H}_{\mathrm{PK}}$ are also zeros:

$$\mathbf{t}_i^T \mathbf{H}_{\mathrm{PK}}^T = [z_i^p \mathsf{H}(z_i),\ z_i^{p+1}\mathsf{H}(z_i),\ \cdots,\ z_i^{p+M-1}\mathsf{H}(z_i)] = \mathbf{0} \qquad i = 1, 2, \ldots, p$$

or

$$\mathbf{T}_p^T \mathbf{H}_{\mathrm{PK}}^T = \mathbf{0}$$

This implies that the filter matrix $\mathbf{H}_{\mathrm{PK}}$ is playing a similar role to the filter matrix $\mathbf{Q}_2^H$ in that

- $\mathbf{H}_{\mathrm{PK}}$ and $\mathbf{Q}_2^H$ have the same dimension, $(M - p) \times M$, and are both full rank;
- $\mathbf{H}_{\mathrm{PK}}$ and $\mathbf{Q}_2^H$ are both orthogonal to the p-dimensional column space of $\mathbf{T}_p$;
- $\mathbf{H}_{\mathrm{PK}}$ *and* $\mathbf{Q}_2^H$ *have the same range, i.e.,* $\mathcal{R}(\mathbf{H}_{\mathrm{PK}}) = \mathcal{R}(\mathbf{Q}_2^H)$.

Therefore, the incorporation of prior knowledge through the filter matrix $\mathbf{Q}_2^H$ is actually an orthogonalized version of prefiltering through the FIR filter matrix $\mathbf{H}_{\mathrm{PK}}$. This relationship facilitates our interpretation of prior-knowledge-incorporation in the frequency domain.

Fig. 7.1 shows the modulus of frequency responses of the FIR filters designed to cancel the signal pole, z_1, of the two-peak signal, *Signal-1*, which has a rational z-transform

$$\mathsf{X}(\mathsf{z}) = \frac{c_1}{1 - z_1\mathsf{z}^{-1}} + \frac{c_2}{1 - z_2\mathsf{z}^{-2}}$$

Let $z_{1\mathrm{PK}} = e^{(-d_{1\mathrm{PK}} + j2\pi f_{1\mathrm{PK}})\Delta t}$ be the given prior knowledge. Then, the filter z-transfer function is

$$\mathsf{H}(\mathsf{z}) = 1 - z_{1\mathrm{PK}}\mathsf{z}^{-1}$$

When the given prior knowledge is exact, i.e., $z_{1\mathrm{PK}} = z_1$ as is the case of full curves in Fig. 7.1, the first pole of *Signal-1* is canceled or suppressed completely, and the problem of estimating poles of 2 exponentials is simplified to that of only 1 exponential after such prefiltering. Naturally, the decrease of the number of unknowns results in better resolution and accuracy, especially when the known and unknown exponentials are closely spaced, overlapping, and thus difficult to resolve.

When the given prior knowledge, $f_{1\mathrm{PK}}$ or $d_{1\mathrm{PK}}$ of $z_{1\mathrm{PK}}$, is perturbed, the designed filters are also perturbed as shown by the dashed lines in Fig. 7.1. By multiplying the frequency response of an FIR filter with the original signal spectrum, we can get the spectrum of a filtered signal. In Fig. 7.1 (a), $d_{1\mathrm{PK}}$ is fixed at d_1. The filter frequency response shifts due to a perturbation of the known frequency, $f_{1\mathrm{PK}} = f_1 + \Delta f$. This affects both filtered peaks, namely, their positions (frequencies) and shapes (damping factors). Intuitively, a positive Δf has a stronger influence than a negative Δf for the same absolute $|\Delta f|$, since $f_{1\mathrm{PK}}$ is closer to f_2 in the former case and further away from f_2 in the latter case when $f_1 < f_2$. In Fig. 7.1 (b), $f_{1\mathrm{PK}}$ is fixed at f_1 but $d_{1\mathrm{PK}}$ is perturbed. The series of the filter frequency responses corresponding to a perturbation of the known damping factor, $d_{1\mathrm{PK}} = d_1 + \Delta d$, is always centered

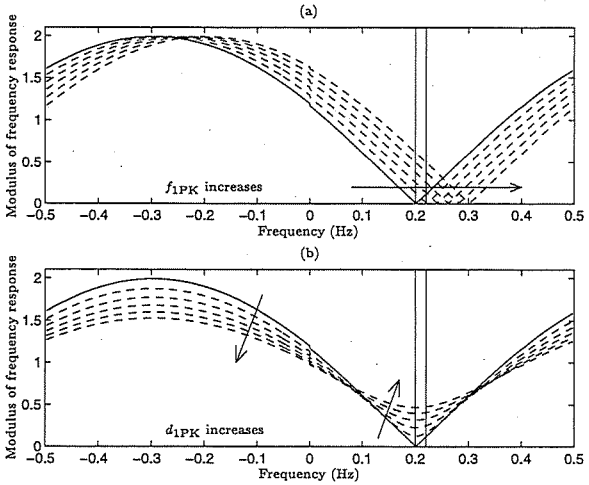


Figure 7.1: Frequency response of FIR filters H(z) for *Signal-1*, when the given prior knowledge is (a) $f_{1PK} = f_1 + \Delta f$, $d_{1PK} = d_1$, and (b) $f_{1PK} = f_1$, $d_{1PK} = d_1 + \Delta d$. The parallel vertical full lines denote the position of the two sinusoids in *Signal-1*. When Δf increases from 0 Hz till 0.1 Hz or Δd increases from 0 rad/s till $0.1 \times 2\pi$ rad/s, the filter frequency response shifts in the direction of arrows.

at $f_1 = 0.2$ Hz, the frequency of the first peak, which is supposed to be annihilated. The change in Δd determines the degree of suppression of the first peak. The larger Δd is, the less the first peak is suppressed. The un-suppressed part of the filtered first peak affects the parameter estimation of the second peak, especially when Δd is positive, instead of negative. This is the exponential effect in the damping factor. Take Fig. 7.1 (b) for example. As d_{1PK} changes from 0.01 rad/s to $0.01 + 0.1 \times 2\pi$ rad/s, the minimum modulus of the filter frequency response changes from 0.01 to 0.47. However, if d_{1PK} changes from 0 rad/s to 0.01 rad/s, this minimum only changes from 0 to 0.01, which is hardly visible. Of course, such frequency-domain analysis is rough. But it gives some

idea about the effect of perturbations in the known signal poles.

Though the prefiltering operation using $\mathbf{H}_{PK}$ of an FIR filter can be helpful in interpretation, it is less accurate in our parameter estimation than the HTLS(PK) method using the orthogonal filter matrix $\mathbf{Q}_2$. This is because, in HTLS(PK), $\mathbf{Q}_2^H$ has orthonormal rows due to $\mathbf{Q}_2^H\mathbf{Q}_2 = \mathbf{I}_{M-p}$. As a result, *a prefiltering operation $\bar{\mathbf{X}} = \mathbf{X}\mathbf{Q}_2^*$ does not change the original covariance matrix of white noise, which is, asymptotically, a scalar multiple of an identity matrix, in contrast to a prefiltering operation $\bar{\mathbf{X}} = \mathbf{X}\mathbf{H}_{PK}^T$*. This can be easily verified from

$$\mathbf{X} = \overset{\circ}{\mathbf{X}} + \mathbf{E}, \quad \overset{\circ}{\mathbf{X}}^H \mathbf{E} = 0, \quad \mathbf{E}^H\mathbf{E} = \mu^2\mathbf{I}_M = L\nu^2\mathbf{I}_M$$

$$\bar{\mathbf{X}} = (\overset{\circ}{\mathbf{X}} + \mathbf{E})\mathbf{Q}_2^* = \overset{\circ}{\bar{\mathbf{X}}} + \bar{\mathbf{E}}$$

where $\overset{\circ}{\bar{\mathbf{X}}} = \overset{\circ}{\mathbf{X}}\mathbf{Q}_2^*$, $\bar{\mathbf{E}} = \mathbf{E}\mathbf{Q}_2^*$, and

$$\bar{\mathbf{X}}^H\bar{\mathbf{X}} = \mathbf{Q}_2^T\,\overset{\circ}{\mathbf{X}}^H\overset{\circ}{\mathbf{X}}\,\mathbf{Q}_2^* + \mu^2\mathbf{Q}_2^T\mathbf{Q}_2^* = \mathbf{Q}_2^T\,\overset{\circ}{\mathbf{X}}^H\overset{\circ}{\mathbf{X}}\,\mathbf{Q}_2^* + \mu^2\mathbf{I}_{M-p}$$

Thus the requirement for the HTLS estimator that the noise covariance matrix be a scalar multiple of identity matrix is satisfied after prefiltering with $\mathbf{Q}_2^H$. The benefit of the orthogonal filter matrix $\mathbf{Q}_2^H$ compared to the FIR filter matrix $\mathbf{H}_{PK}$ is also confirmed by numerical results in the next section.

It is to be noted that the prior-knowledge-incorporating methods, LCTLS-FLP and LCTLS-BLP, can also be interpreted as such a prefiltering operation in a similar way.

7.2 Quantification of simulated signals

In this section, we study the HTLS(PK) method more extensively, and compare it to the prefiltered HTLS and HTLS method. Again, we work with the second-order simulated signal, *Signal-1*, and the fifth-order simulated signal, *Signal-2*. For *Signal-1*, $N = 25$ data samples are used, and the parameters $f_1 = 0.2$ Hz and $d_1 = 0.01$ rad/s of the first peak are assumed to be known *a priori*. For *Signal-2*, $N = 128$ data samples are used, and the parameters $f_4 = 353$ Hz and $d_4 = 117$ rad/s of the fourth peak are assumed to be known *a priori*.

7.2.1 Comparison of HTLS(PK) and prefiltered HTLS

Fig. 7.2 compares HTLS(PK) with prefiltered HTLS using prior knowledge, z_1, of *Signal-1*. $\mathbf{H}_{PK}$-HTLS stands for the prefiltered HTLS method via $\bar{\mathbf{X}} =$

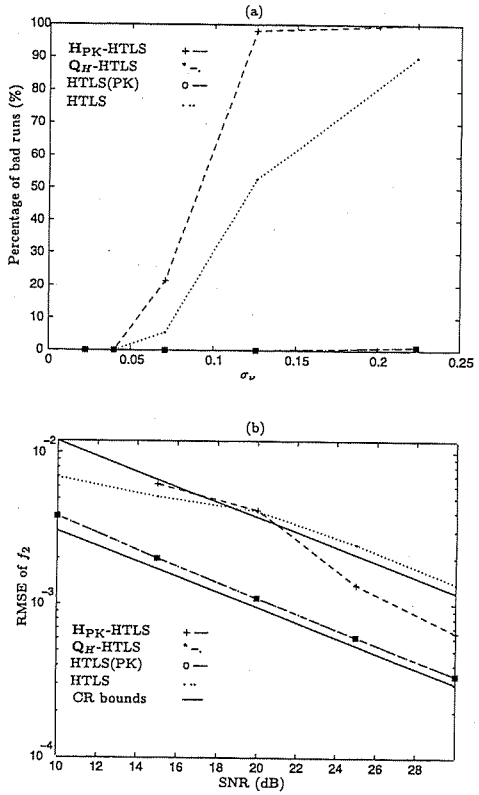


Figure 7.2: *Signal-1*: (a) Failure rate versus σ_ν, (b) RMSE of the estimates of f_2 versus SNR. Prior knowledge of $f_1 = 0.2$ Hz and $d_1 = 0.01$ rad/s is used in the HTLS(PK) method.

$\mathbf{X}\mathbf{H}_{PK}^T$. $\mathbf{Q}_H$-HTLS represents the prefiltered HTLS method via $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{Q}_H$, where $\mathbf{Q}_H$ is the orthogonalized $\mathbf{H}_{PK}^T$ obtained from the "skinny" QR decomposition:

$$\mathbf{H}_{PK}^T = \mathbf{Q}_H \mathbf{R}_H$$

As expected, $\mathbf{Q}_H$-HTLS and HTLS(PK) produce exactly the same results, while the results of $\mathbf{H}_{PK}$-HTLS are much worse. Thus, the importance of orthogonalization of the filter matrix $\mathbf{H}_{PK}$ is evident. At very high noise level, $\mathbf{H}_{PK}$-HTLS is even worse than the HTLS method without using any prior knowledge. The following is our explanation. On the one hand, the use of $\mathbf{H}_{PK}$ introduces colored noise. On the other hand, the smaller dimension of $\mathbf{H}_{PK}$ compared to $\mathbf{H}_{FIR_t}$ in (3.1) due to further truncation of filtered output leads to a severe decrease of the SNR as explained on p. 31, Section 3.1 of Chapter 3, since the part of the filtered damped signal with highest SNR has been discarded.

7.2.2 Enhancement of HTLS(PK)

When the SNR is not sufficiently high, enhancement procedures can be applied prior to the HTLS(PK) method. Fig. 7.3 compares the results of the HTLS(PK) method enhanced by the MV and STLN methods, respectively, with the non-enhanced HTLS(PK) and HTLS methods. Both enhancement procedures improve the resolution and parameter accuracy. HTLN(PK) gives better accuracy than MV-HTLS(PK) at higher SNR's, but MV-HTLS(PK) is superior to HTLN(PK) at low SNR's, due to the non-convergence problems. Hence, for enhancement at low SNR, the MV method is both efficient and accurate.

7.2.3 Computational efficiency of HTLS(PK)

Table 7.1 compares the computational efficiency of HTLS(PK) and HTLS step by step. Obviously, HTLS(PK) is computationally cheaper than HTLS. This difference in computational efficiency is more pronounced as the number p of *a priori* known signal poles increases and/or the Hankel data matrix gets more rectangular. This is because the computation of the SVD of an $\hat{L} \times \hat{M}$ matrix requires $\mathcal{O}[\hat{L}\hat{M}^2 + \hat{M}^3]$ floating point operations ($\mathcal{O}[\,]$ denotes the order of magnitude, as listed on page 66, Section 4.3 of Chapter 4, which is the computationally most intensive part of both algorithms. Even if HTLS and HTLS(PK) use a Hankel data matrix of the same dimension, $L \times M$, the HTLS(PK) estimator only needs to compute the SVD's of smaller sized matrices. That is, in the SVD step (Step 1 of the HTLS algorithm, Algorithm

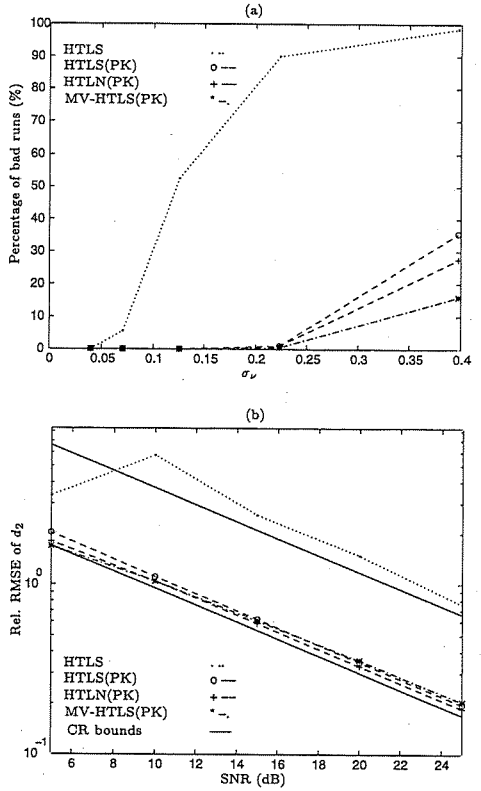


Figure 7.3: *Signal-1*: (a) Failure rate versus σ_ν, (b) Relative RMSE of the estimates of d_2 versus SNR. HTLN(PK) has 2 non-convergent cases in the STLN iterations at SNR=10dB, and 22 non-convergent cases at SNR=5dB. Prior knowledge used is $f_1 = 0.2$ Hz and $d_1 = 0.01$ rad/s.

flops	HTLS $M \approx 0.5N$
Step 1: SVD	1.13e+05
Step 2: TLS	7.98e+03
Step 3: LS	5.90e+03
Total	1.27e+05

flops	HTLS(PK) $M \approx 0.5N$	HTLS(PK) $M \approx 0.6N$	HTLS(PK) $M \approx 0.7N$	HTLS(PK) $M \approx 0.8N$
Step 1: QRD	2.98e+03	3.91e+03	5.56e+03	6.81e+03
Step 2: SVD	9.35e+04	9.08e+04	6.90e+04	5.32e+04
Step 3: TLS	1.31e+03	1.12e+03	8.31e+02	6.39e+02
Step 4: LS	5.90e+03	5.90e+03	5.90e+03	5.90e+03
Total	1.04e+05	1.02e+05	8.12e+04	6.66e+04

Table 7.1: Number of flops used in MATLAB in the HTLS and HTLS(PK) algorithms per step when quantifying the noisy (SNR=10 dB or $\sigma_\nu = 0.2236$) simulated signal, *Signal-1*. Steps 1 through 3 of the HTLS method correspond to the steps outlined in Algorithm 1, given in Chapter 2, and steps 1 through 4 of the HTLS(PK) method correspond to the steps outlined in Algorithm 11, given in Chapter 6. The LS step uses the SVD method.

1 and Step 2 of the HTLS(PK) algorithm, Algorithm 11) the dimension of the matrix used is reduced from $L \times M$ to $L \times (M - p)$; while in the TLS step (Step 2 of HTLS and Step 3 of HTLS(PK)), the dimension of the matrix taken is reduced from $L \times 2K$ to $L \times 2(K - p)$. The saving in computation time can be roughly verified using theoretical expressions in Section 4.3 of Chapter 4. Actually, the computational cost of HTLS(PK) can be further decreased. Referring to Fig. 6.2, the HTLS(PK) method has a wider range of minimum errors than the HTLS method. Therefore, a more rectangular ($0.35 \leq M/N \leq 0.9$, instead of $0.4 \leq M/N \leq 0.6$) Hankel matrix is allowed in the HTLS(PK) method.

7.2.4 Sensitivity of HTLS(PK) w.r.t. the accuracy of given poles

For *Signal-2*, we have shown the advantages of the HTLS(PK) method in Fig. 6.3 and Fig. 6.4 of Chapter 6, when the given prior knowledge, z_4, is exact.

If the pole of peak 4 is only approximately known, rather than exactly

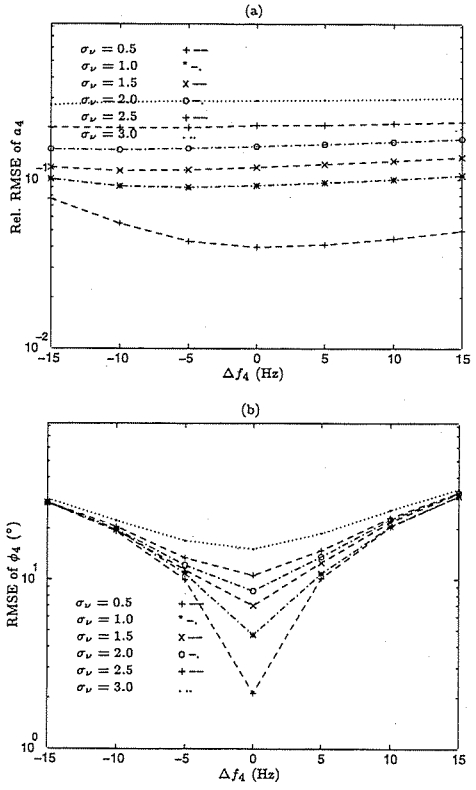


Figure 7.4: (Relative) RMSE of the estimates of (a) a_4 (b) ϕ_4 versus Δf_4, obtained by the HTLS(PK) method. The imposed prior knowledge is $f_4 = 353 + \Delta f_4$ Hz, $d_4 = 117$ rad/s.

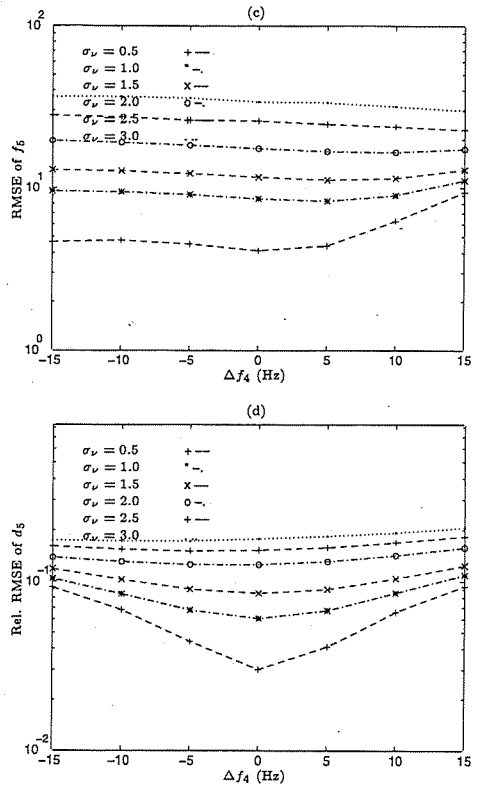


Figure 7.4: (cont.) (Relative) RMSE of the estimates of (c) f_5 (d) d_5 versus Δf_4, obtained by the HTLS(PK) method. The imposed prior knowledge is $f_4 = 353 + \Delta f_4$ Hz, $d_4 = 117$ rad/s.

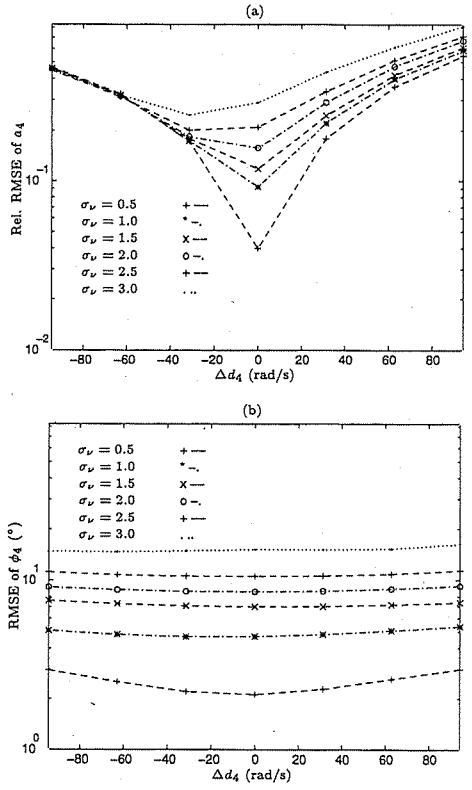


Figure 7.5: (Relative) RMSE of the estimates of (a) a_4 (b) ϕ_4 versus Δd_4, obtained by the HTLS(PK) method. The imposed prior knowledge is $f_4 = 353$ Hz, $d_4 = 117 + \Delta d_4$ rad/s.

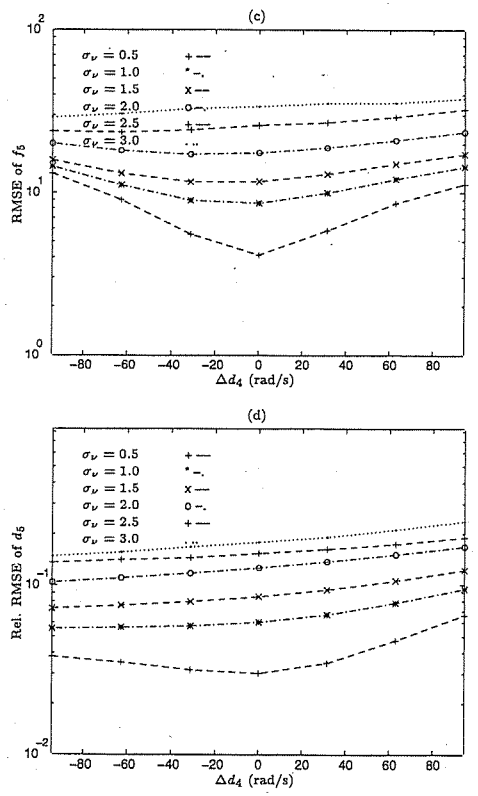


Figure 7.5: (cont.) (Relative) RMSE of the estimates of (c) f_5 (d) d_5 versus Δd_4, obtained by the HTLS(PK) method. The imposed prior knowledge is $f_4 = 353$ Hz, $d_4 = 117 + \Delta d_4$ rad/s.

known, the estimates of the HTLS(PK) method are affected depending on the precision of the known pole, the frequency of its spectral line, the SNR level of the noisy signal, etc. The sensitivity of the estimates of HTLS(PK) with respect to perturbations of the known pole is studied here via computer simulations. Only the estimates of peak 4, as well as the estimates of peak 5 which is overlapping with peak 4, are noticeably affected when a minor perturbation is applied. The smaller the noise level, the more visible this effect. Fig. 7.4 and Fig. 7.5 show the (relative) RMSE values of the most affected parameters versus Δf and Δd, respectively. The perturbation on f_4 affects mainly the estimates of ϕ_4, f_5 and d_5, while a perturbation on d_4 affects mainly a_4, f_5 and d_5. The minimum (relative) RMSE values are mostly obtained at non-perturbed f_4, d_4 values. These (relative) RMSE curves are not exactly symmetric for Δf or Δd. A positive Δf or Δd affects more the accuracy, as analyzed in the previous section in terms of filtering operations.

7.3 Quantification of a real-world NMR signal

Fig. 7.6 shows the signal to be processed, which is a ^{31}P NMR signal measured from the brain of a healthy volunteer. The measurement data are composed of 1024 samples in the time domain and assumed to be modeled by (1.4). To get this spectrum, the zero-order phase $\phi^{(0)} = 231.81°$, which can be determined from the spectrometer phase settings, is subtracted and 1009 time-domain data samples are taken for the Fourier transform starting from the 16th point (x_{16}) on (the first 15 are deleted in order to eliminate the broad hump that appears in the original spectrum). The same 1009 data samples, i.e., N=1009, will be used in the HTLS and HTLS(PK) estimation algorithms for the quantification of this signal. All the estimated parameters are extrapolated to the time origin and the phases are corrected. Among the peaks of interest, NAD is a difficult peak to resolve, because of its small amplitude and its position on the wing of α-ATP. Since the accuracy of the estimates of NAD is dependent on that of α-ATP, attention is focused on the fit of the cluster of NAD and α-ATP peaks, as circled in the spectrum in Fig. 7.6.

One way of fitting the cluster is to apply the HTLS algorithm using an overestimated model order estimate. In Fig. 7.7(a), the estimated values of the frequency, damping factor, amplitude and phase of the α-ATP peaks and NAD peak, if it is fitted, are plotted versus the model order K used. For model orders ranging from 10 to 19, only the dominant α-ATP is fitted as a doublet while NAD is not fitted at all. When the model order used is as high as 20 and above, α-ATP is still fitted as a doublet and NAD is fitted as one exponential.

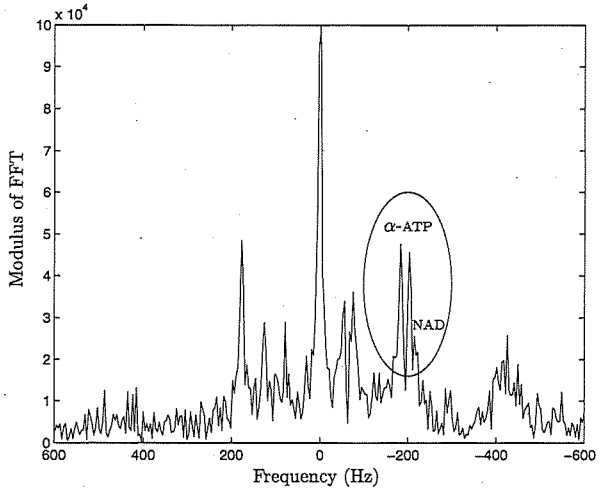


Figure 7.6: FFT spectrum of ^{31}P NMR signal of human brain (after omission of the first 15 data points and correction of zero-order phase $\phi^{(0)} = 231.81°$).

When HTLS fails to fit the NAD peak, the frequencies of the α-ATP doublet are the least affected parameters, as can be seen from the frequency plot of Fig. 7.7(a). However, the damping factor plot of Fig. 7.7(a) indicates that the two damping factors of the doublet are overestimated, when the NAD peak emerging from the wing of α-ATP is not fitted. A possible explanation is that the two estimated damping factors should be larger so that the two exponentials fit the whole cluster of α-ATP and NAD. Especially at frequencies around -203 Hz, the damping factor is seriously overestimated because of the neighboring NAD peak. Actually, the two damping factors of α-ATP should be equal according to physical disciplines and reality. Such prior knowledge, however, cannot be used in the HTLS method.

Another way of fitting the cluster is to apply the HTLS(PK) method with prior knowledge of α-ATP, which might be obtained from chemical analysis or from a pre-fit of the α-ATP doublet using the HTLS method as described

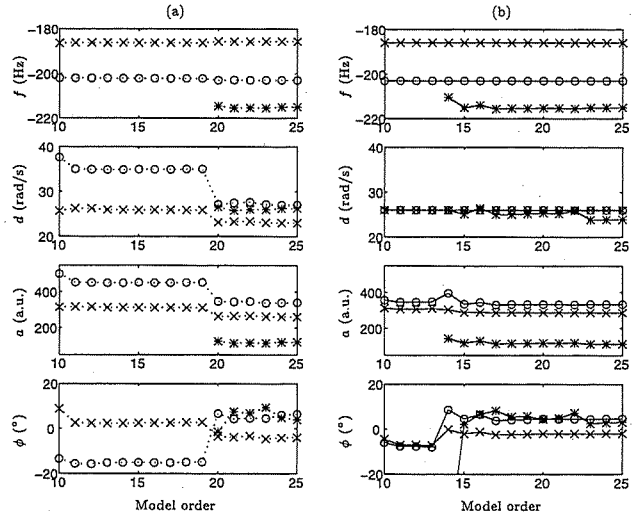


Figure 7.7: Estimates of f, d, a and ϕ of the cluster of NAD ($*$) and α-ATP (o and $\times$) by the (a) HTLS ($\cdots$) and (b) HTLS(PK) (—) methods versus the model order used. Prior knowledge used in the HTLS(PK) method is $f = -203$ Hz, $d = 26$ rad/s and $f = -186$ Hz, $d = 26$ rad/s for the α-ATP doublet.

above. Using prior knowledge that α-ATP is composed of two exponentials with signal poles $e^{-26\Delta t - j2\pi 203\Delta t}$ and $e^{-26\Delta t - j2\pi 186\Delta t}$ (obtained from a pre-fit with HTLS shown in Fig. 7.7(a) and based on physical reality), the HTLS(PK) method is able to fit the NAD peak from order $K = 14$ on, as shown in Fig. 7.7(b).

peak	method[a]	f (rad/s)	d (rad/s)	a (a.u.)	ϕ(°)
NAD	HTLS	-214.6±1.5	26.6±9.5	125.0±48.6	-1.5±22.3
	HTLS(PK)	-215.1±1.1	25.1±7.0	117.6±28.5	2.1±13.9
α-ATP	HTLS	-202.9±0.6	27.2±4.0	348.0±59.2	6.6± 9.7
		-185.7±0.5	23.1±2.9	264.0±30.2	-3.5± 6.6
	HTLS(PK)	-203.0±0.0	26.0±0.0	335.2±21.5	4.5± 3.7
		-186.0±0.0	26.0±0.0	288.2±16.1	-2.2± 3.2

[a] Results of HTLS and HTLS(PK) are obtained, respectively, at $K = 20$ and $K = 15$.

Table 7.2: Parameter estimates and their CR bounds of the cluster of NAD and α-ATP peaks of a real-world ^{31}P NMR signal measured from human brain.

Table 7.2 compares the results of these parameter estimation methods. The phase estimates are similar, both close to zero which is consistent with physical reality. But the estimates of the two damping factors of the α-ATP doublet as computed by HTLS are not satisfactory since their difference, which should be zero, is too large. This affects directly the accuracy of the NAD parameters, as well as the amplitudes of the α-ATP peaks. As a result, the amplitude-ratio of the two exponentials that fit the α-ATP doublet is $348/264 \approx 1.318$ for the HTLS method and $335.2/288.2 \approx 1.163$ for the HTLS(PK) method, the latter ratio of which is more close to the ratio of 1.159 known *a priori*.

The sensitivity of the parameters estimated by the HTLS(PK) method with respect to minor changes in the given poles of the α-ATP doublet can be observed in Fig. 7.8. In Fig. 7.8(b), the estimates of the cluster are plotted versus the perturbation in the damping factors of the two exponentials comprising the α-ATP doublet. It seems that the estimates are not very sensitive to the choice of the two equal damping factors assumed known, though the damping factor and amplitude of NAD decreases a little as the damping factors of the α-ATP doublet increase together. In particular, the amplitude ratio of the α-ATP doublet is within the narrow range of $1.144 \sim 1.173$, quite close to the known ratio 1.159. In Fig. 7.8(a), however, the NAD peak cannot be estimated when the frequencies of two known poles are increased together to as high as -202.8

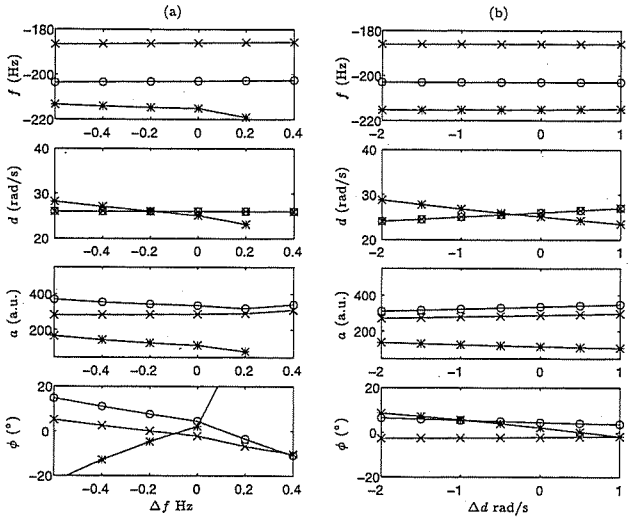


Figure 7.8: Estimates of f, d, a and ϕ of the cluster of NAD ($*$) and α-ATP (o and $\times$) by the HTLS(PK) methods versus (a) Δf, i.e., prior knowledge used is $f = -203 + \Delta f$ Hz, $d = 26$ rad/s and $f = -186 + \Delta f$ Hz, $d = 26$ rad/s for the α-ATP doublet; (b) Δd, i.e., prior knowledge used is $f = -203$ Hz, $d = 26 + \Delta d$ rad/s and $f = -186$ Hz, $d = 26 + \Delta d$ rad/s for the α-ATP doublet.

and -185.8 Hz, keeping a difference of 17 Hz in accordance with physical reality. The phase estimates of the cluster are a bit sensitive when the frequencies of the two known poles change. This sensitivity in phase, in some sense, may serve as a guidance in choosing a suitable values for the poles we want to fix.

7.4 Conclusion

The prior-knowledge-incorporating method, HTLS(PK), has been shown to belong to the class of prefiltered HTLS method. Using the concept of prefiltering, the effect of incorporating prior knowledge can be visualized in the frequency domain. For parameter estimation, however, HTLS(PK) is far superior to an un-orthogonalized prefiltered HTLS due to the use of an orthogonal filter matrix, which preserves the white-noise characteristic.

Actually, the prior-knowledge-incorporating methods, LCTLS-FLP and LCTLS-BLP, can also be interpreted as prefiltering methods in a similar way.

When the SNR is not sufficiently high, an MV enhancement procedure is suggested prior to the HTLS(PK) method.

The computational efficiency of the HTLS(PK) method compared to the HTLS method is examined step-wisely using MATLAB. The sensitivity of the HTLS(PK) method with respect to perturbations of the exactly known signal poles is also studied via computer simulations.

In application to real-world NMR signals, HTLS(PK) offers the possibility of imposing knowledge of selected poles obtained in a preceding round of HTLS. Such poles can be selected in difficult regions of a spectrum. By fixing poles of selected stronger peaks, HTLS-PK improves the quantification of weak peaks in the vicinity of these stronger peaks.

Although HTLS(PK) is ideal in both parameter accuracy and computational efficiency, it requires that both the frequency and damping factor of certain peaks are known *a priori*. In practice, the frequencies and/or damping factors are *not* very often known. A further development would be to adapt HTLS(PK) to estimation of an overall frequency shift of a spectrum, imposing *a priori* knowledge of the *relative* individual peak frequencies, as is often the case in NMR spectroscopy.

Chapter 8

General Conclusions

The subject of this thesis is accurate estimation of parameters of a sum of K exponentially damped sinusoids embedded in white noise. In the previous chapters, several variants of the HTLS estimation method have been presented and studied. Similarly to the HTLS estimation method, which is actually Kung et al.'s state-space-based method using the TLS principle, these variants of the HTLS method are also based on subspace separation techniques, etc. They improve the resolution and accuracy of the estimated parameters to different degrees, at different cost, and under different conditions.

8.1 Summary

In this concluding chapter, we summarize the main features of the presented variants of the HTLS method, as well as how and when they perform best. Referring to Table 8.1, these methods are classified into two categories according to whether prior knowledge is required.

Category 1: Enhanced HTLS estimators without using prior knowledge

The MV-HTLS, CA-HTLS, STLS-HTLS, HTLN (an efficient version of STLN-HTLS) methods and their possible combinations (e.g., MV-CA-HTLS etc., which are omitted here) belong to the class of enhanced HTLS estimators and require no prior knowledge except perhaps a model order estimate. These enhancement algorithms serve as preprocessing steps of "cleaning up" the data before estimating the values of the signal parameters.

method	prior knowledge requirement	computational cost ratio[a]
Category[b] 1:		
CA-HTLS	—	1.86
MV-HTLS	—	1.29
STLS-HTLS	—	202
HTLN	—	980
Category 2:		
$\mathbf{H}_{\mathrm{FIR}t}$-HTLS	approximate frequency region of interest	1.09
$\mathbf{H}_{\mathrm{IIR}}$-HTLS	approximate frequency region of interest	1.06
HTLS_F-HTLS	approximate frequency region of interest	1.80
HTLS(PK)	some poles	0.85
HTLN(PK)	some poles	401

[a] Ratio of the averaged number of flops of a method to that of the HTLS method when processing 200 noisy (SNR= 15 dB) realizations of *Signal-1* with 25 data points and 2 peaks under typical conditions simulated in the thesis.

[b] The two categories are classified according to whether prior knowledge is required.

Table 8.1: Variants of the HTLS method and some of their features.

Although HTLS itself can filter out part of the noise by truncating the SVD of the Hankel data matrix to rank K and discarding the non-significant singular values and vectors, such SVD filtering is not always satisfactory especially at low SNR. This is partly because the computed rank-K matrix, spanning the signal subspace of the original Hankel data matrix, does not preserve the Hankel structure anymore. The above-mentioned enhancement algorithms, however, aim at finding a matrix of both rank-K and Hankel structure using different numerical techniques so as to approximate the original data and enhance the parameter estimates.

The MV and Cadzow enhancement procedures are based on the minimum variance estimation and Cadzow's estimation method, respectively, in which the SVD plays a key role. The STLS and STLN enhancement procedures are based on the optimization-oriented approaches. Our main conclusions regarding these

enhancement methods are:

- The convergence of the MV and Cadzow's enhancement procedures is found to be unnecessary when followed by an SVD-based estimator such as HTLS, since the accuracy of the estimated parameters deteriorates despite a smoother spectrum. That is, a "cleaner" spectrum does not guarantee better parameter accuracy. Therefore, it is suggested to use only one or two iterations of the MV and/or Cadzow's enhancement procedure in the parameter estimation. In this way, the computational cost is drastically reduced.
- When nonoverlapping peaks are quantified, all the methods, MV-HTLS, CA-HTLS, STLS-HTLS and HTLN, are comparable to HTLS in resolution and parameter accuracy, but the HTLS method is the most efficient. Therefore, no enhancement is suggested in this case.
- When overlapping peaks are to be quantified, then the MV-HTLS method doubles the resolution of the CA-HTLS method independently of the noise variance at comparable parameter accuracy. The best parameter accuracy, especially the bias and standard deviation of the estimated parameters, is obtained with the STLS-HTLS and HTLN methods. However, the computation time required by STLS-HTLS and HTLN is enormous compared to the one-iteration methods, MV-HTLS and CA-HTLS. Actually, the MV-HTLS method is the cheapest since an MV iteration requires less than 10% of the time needed to execute a Cadzow's iteration, depending on the number of data points and the model order. Therefore, trade-off must be made between the MV-HTLS method and the STLS-HTLS, HTLN method, according to parameter accuracy and computational efficiency requirement.
- The SVD-based MV and Cadzow's iterations converge to suboptimal solutions, while the optimization-oriented STLS and STLN iterations can converge to optimal ones. This partly explains why the MV and Cadzow's enhancement procedures at convergence do not produce better results than after one iteration and the parameter accuracy, especially the bias and standard deviation, of the STLS-HTLS and HTLN methods is superior to that of the MV and Cadzow's methods.

Category 2: Prefiltered HTLS estimators using prior knowledge

The class of $\mathbf{H}_{\mathrm{FIR}t}$-HTLS, $\mathbf{H}_{\mathrm{IIR}}$-HTLS, HTLS$_F$-HTLS, HTLS(PK), and HTLN(PK) methods requires some kind of prior knowledge which is incorporated in a form of prefiltering operation prior to applying the HTLS estimation

algorithm. This enables better resolution and accuracy of the estimated parameters of interest.

Previously, the incorporation of prior knowledge is only possible in nonlinear least squares based techniques. In this thesis, we have incorporated some prior knowledge into HTLS, an SVD and subspace-based estimation method.

The prefiltering operation in $\mathbf{H}_{\mathrm{FIR}t}$-HTLS and $\mathbf{H}_{\mathrm{IIR}}$-HTLS methods is realized by constructing a full-rank square filter matrix $\mathbf{H}_{\mathrm{FIR}t}$ or $\mathbf{H}_{\mathrm{IIR}}$ from an FIR or IIR filter, and then multiplying it with the original Hankel data matrix. The HTLS$_F$ prefiltering operation is realized by first invoking HTLS to get a rough estimation of all K sinusoids, then subtracting the signal that fits the unwanted peaks from the original one, and finally estimating the parameters of wanted peaks from the residual signal. The prefiltering operation in HTLS(PK) is realized by first finding an orthogonal basis of the subspace of the Vandermonde vectors comprising known signal poles, and then projecting the original Hankel data matrix onto this orthogonal basis. Here follow our main conclusions regarding this class of algorithms:

- When all the K sinusoids are of interest and the frequency region of these sinusoids is known (*a priori* or via FFT) approximately, the $\mathbf{H}_{\mathrm{FIR}t}$-HTLS or $\mathbf{H}_{\mathrm{IIR}}$-HTLS method can be used as a kind of enhanced estimation. By designing suitable FIR and IIR bandpass filters that encompass all the K sinusoids, these two methods can mostly remove the corruption noise outside the filter passband and produce better parameter estimates than the HTLS method as long as the signal spectrum is not too wide.
- When only some of the sinusoids (fewer than K) are of interest with approximately known frequency region (*a priori* or via FFT) and the unwanted peaks are easily resolvable, the HTLS$_F$-HTLS method is suggested. A prerequisite of the HTLS$_F$-HTLS prefiltering is the accurate pre-estimation in the HTLS$_F$ prefiltering.
- When only some of the sinusoids (fewer than K) are of interest with approximately known frequency region (*a priori* or via FFT) and these wanted peaks are *not much overlapping* with the unwanted difficult peaks, the $\mathbf{H}_{\mathrm{FIR}t}$-HTLS or $\mathbf{H}_{\mathrm{IIR}}$-HTLS method, with an appropriately designed filter, can give better results than HTLS or HTLS$_F$-HTLS, especially at low SNR's. In this case, the filter should be designed to encompass only the wanted peaks and suppress the unwanted ones and the noise. When unwanted peaks are easily resolvable, the parameter accuracy can be improved through the combination of HTLS$_F$-HTLS prefiltering and FIR or IIR bandpass prefiltering (e.g., HTLS$_F$-$\mathbf{H}_{\mathrm{FIR}t}$-HTLS, HTLS$_F$-$\mathbf{H}_{\mathrm{IIR}}$-HTLS).

- When some of the signal poles (i.e., both frequencies and damping factors of the corresponding peaks) are known *a priori*, the HTLS(PK) and HTLN(PK) methods can be invoked to incorporate such prior knowledge. The accuracy of the HTLN(PK) method is slightly better than that of the HTLS(PK) method, since HTLN(PK) is actually the STLN enhanced HTLS(PK) method. However, the improvement in accuracy is not large enough to justify the large increase in computational load, especially when the number of data samples is large. Instead, when the SNR is low, the MV enhanced HTLS(PK) method, or MV-HTLS(PK), is suggested.

- Among these variants of HTLS, only the HTLS(PK) method preserves the original noise covariance matrix, which is the scalar multiple of an identity matrix, due to the use of an orthogonal filter matrix.

- The computational cost of $\mathbf{H}_{\mathrm{FIR}t}$-HTLS and $\mathbf{H}_{\mathrm{IIR}}$-HTLS are basically the same as that of HTLS, and the computational cost of HTLS_F-HTLS almost doubles that of HTLS, depending on the number of wanted peaks. The HTLS(PK) method, however, lowers the computational cost of HTLS.

8.2 Suggestions for further research

There are five areas in which further research should be fruitful.

1. There is a requirement for the HTLS estimator that the noise covariance matrix be the scalar multiple of an identity matrix, or the noise be asymptotically white. However, most of the variants of the HTLS method, such as CA-HTLS, MV-HTLS, $\mathbf{H}_{\mathrm{FIR}t}$-HTLS, $\mathbf{H}_{\mathrm{IIR}}$-HTLS, HTLS_F-HTLS, etc., have the disadvantage of coloring the white noise after preprocessing resulting in a noise covariance matrix, which is not a scalar multiple of an identity matrix, after prefiltering operation. This affects parameter accuracy, since the white-noise requirement in the succeeding HTLS estimator is not satisfied. Once this problem is solved, we are likely to obtain even better parameter estimates.

2. Both STLS and STLN techniques approximate full-rank Hankel data matrices by reduced-rank Hankel data matrices, preserving the matrix structure. The enhancement of STLS or STLN remove noise effectively and improve resolution and parameter accuracy, especially, the standard deviations of the estimated parameters. However, the main disadvantage of the STLS-HTLS, HTLN, and HTLN(PK) methods is their high computational cost. It is desirable to search for more efficient implementations of the STLS and STLN algorithms so that the superior accuracy of these variants of the HTLS method can be exploited in practical applications.

3. HTLS(PK) is ideal in both parameter accuracy and computational efficiency. However, it requires that both the frequency and damping factor of certain peaks are known *a priori*. In practice, the frequencies and/or damping factors are *not* very often known. Very often, as is the case of NMR spectroscopy, only the frequency difference or the damping factor ratio of certain peaks is known. Incorporation of such prior knowledge into the HTLS method will be very useful.

4. Since automated, batch quantification need to be further developed for clinical use, automatic detection and handling of failures is required.

5. The CR bound is a lower bound on the variance of an unbiased estimator. It provides a precision that can be achieved given the observations, independently of the particular method of estimation. The means that the obtained precision can not be better than the CR bound. In our Monte Carlo experiment, RMSE values are averaged over *good* runs only, *excluding failures*. As a result, the RMSE curves are sometimes "swapped" from above to below the line of CR bound when quite a number of failures are excluded. Such a "swapping" phenomenon ought to be studied, since it may constitutes the primary failure mechanism. In this thesis, only *estimated parameters of* **succeeded** quantifications are studied. As further research, the properties of those noise realizations that resulted in failures should also be studied. If one understands the failures encountered in a Monte Carlo experiment, it might become possible to somehow reduce their number.

Appendix A

Simulated signals

The following signals are simulated using the model function (1.4), which is repeated here:

$$x_n = \sum_{k=1}^{K} a_k e^{j\varphi_k} e^{(-d_k + j2\pi f_k)n\Delta t} + e_n \qquad n = 0, \ldots, N-1$$

Signal-1

f_s (Hz)	order K	peak k	f_k (Hz)	d_k (rad/s)	a_k (a.u.)[a]	ϕ_k(°)
1	2	1	0.2	0.01	1	0
		2	0.22	0.02	1	0

[a] a.u. means arbitrary units.

Table A.1: Exact parameter values of the simulated signal: *Signal-1*.

This signal is frequently used in the literature, see, e.g. [63, Example 2]. The decisive frequency intervals in Table A.2 are obtained from the CR bounds of each peak at $\sigma_\nu = 0.2$ or SNR$\approx$ 10.9691dB, $N = 25$, and $f_s = 1$ Hz. Typically, $N = 25$ data samples are used in the parameter estimation of this thesis.

peak k	exact f_k (Hz)	decisive	frequency	interval (Hz)
1	0.2		(0.1906	0.2094)
2	0.22		(0.2094	0.2306)

Table A.2: Decisive frequency intervals used for failure definition in parameter estimation of the simulated signal: *Signal-1*.

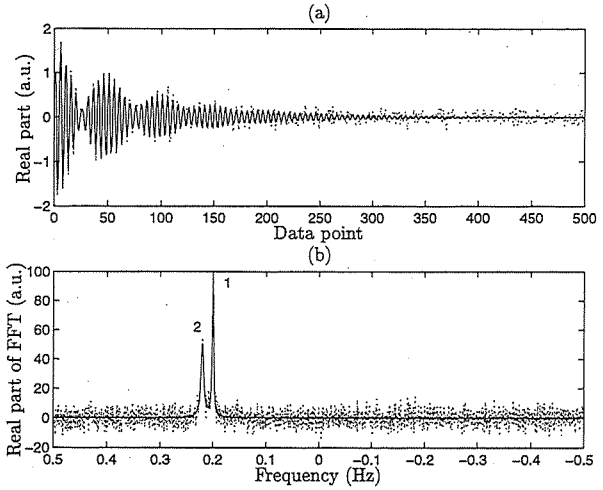


Figure A.1: Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.1$) signal, *Signal-1*, in the (a) time domain and (b) frequency domain.

Signal-2

f_s (Hz)	order K	peak k	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k(°)
$10K$	5	1	-1379	208	6.1	15
		2	-685	256	9.9	15
		3	-271	197	6.0	15
		4	353	117	2.8	15
		5	478	808	17.0	15

Table A.3: Exact parameter values of the simulated signal: *Signal-2*.

peak k	exact f_k (Hz)	decisive frequency interval (Hz)
1	-1379	(-1461 -1297)
2	-685	(-767 -603)
3	-271	(-353 -189)
4	353	(310 396)
5	478	(396 560)

Table A.4: Decisive frequency intervals used for failure definition in parameter estimation of the simulated signal: *Signal-2*.

This signal represents a ^{31}P NMR signal from a perfused rat liver. The decisive frequency intervals in Table A.4 are obtained based on the CR bounds of the 4th and 5th peak at $\sigma_\nu = 8.8$, $N = 128$, and $f_s = 10,000$ Hz. Typically, $N = 128$ data samples are used in the parameter estimation of this thesis.

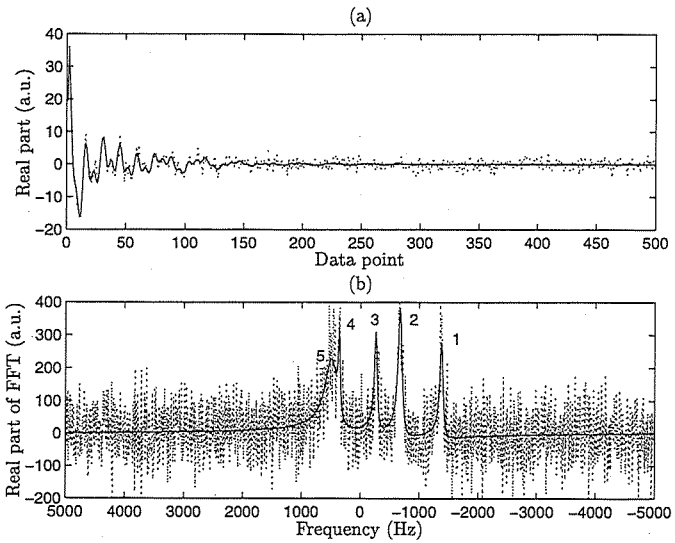


Figure A.2: Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 1.5$) NMR signal, *Signal-2*, in the (a) time domain and (b) frequency domain.

Signal-3

f_s (Hz)	order K	peak k	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k(°)
1	4	1	-0.175	0.02π	0.5	90
		2	-0.1	0.035π	0.5	90
		3	0.1	0.035π	0.5	-90
		4	0.175	0.035π	0.5	-90

Table A.5: Exact parameter values of the simulated signal: *Signal-3*.

peak k	exact f_k (Hz)	decisive	frequency	interval (Hz)
1	-0.175		(-0.2236	-0.1264)
2	-0.1		(-0.1264	-0.0736)
3	0.1		(0.0736	0.1264)
4	0.175		(0.1264	0.2236)

Table A.6: Decisive frequency intervals used for failure definition in parameter estimation of the simulated signal: *Signal-3*.

This signal comes from the literature [51]. Actually, it is a real-valued signal which is described by

$$\overset{\circ}{x}_n = e^{-0.02\pi n}\sin(0.2\pi n) + e^{-0.035\pi n}\sin(0.35\pi n)$$

The decisive frequency intervals in Table A.6 are obtained from the CR bounds of each peak at $\sigma_\nu = 2.475$, $N = 30$, and $f_s = 1$ Hz. Typically, $N = 30$ data samples are used in the parameter estimation of this thesis.

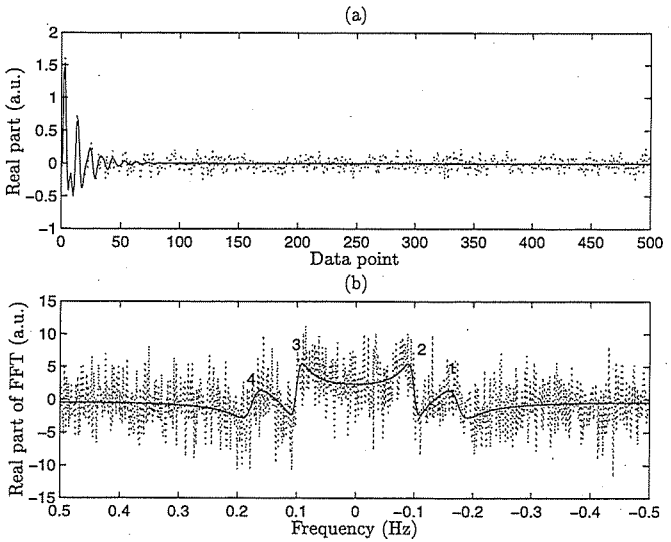


Figure A.3: Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.1$) NMR signal, *Signal-3*, in the (a) time domain and (b) frequency domain.

Signal-4

f_s (Hz)	order K	peak k	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k(°)
1	3	1	-0.2	0.02	0.5	0
		2	0.1	0.05	3	0
		3	0.12	0.1	5	0

Table A.7: Exact parameter values of the simulated signal: *Signal-4*.

peak k	exact f_k (Hz)	decisive frequency interval (Hz)
1	-0.2	(-0.23 -0.17)

Table A.8: Decisive frequency interval used for failure definition in parameter estimation of the simulated signal: *Signal-4*. Only the 1st peak of this signal is of interest.

This signal is used for frequency-selective estimation in the thesis, since only its 1st peak is of interest. The half frequency interval, 0.03 Hz, in Table A.8 is the CR bound of the 1st peak at $\sigma_\nu = 9.53$, $N = 95$, and $f_s = 1$ Hz. Typically, $N = 95$ data samples are used in the parameter estimation of this thesis.

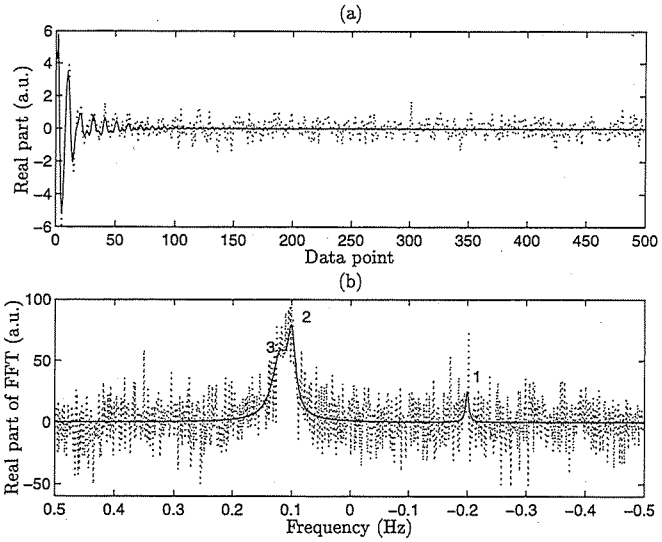


Figure A.4: Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.5$) signal, *Signal-4*, in the (a) time domain and (b) frequency domain.

Signal-5

f_s (Hz)	order K	peak k	f_k (Hz)	d_k (rad/s)	a_k (a.u.)	ϕ_k(°)
$10K$	11	1	-3880	70	214000	0
		2	-282	118	81000	0
		3	-249	768	53100	0
		4	516	50	10000	0
		5	671	64	40000	0
		6	727	215	177000	0
		7	826	164	32000	0
		8	879	90	44000	0
		9	1192	216	96410	0
		10	3250	700	62000	0
		11	3250	100	69000	0

Table A.9: Exact parameter values of the simulated signal: *Signal-5*.

peak k	exact f_k (Hz)	decisive frequency	interval (Hz)
10	3250	(3000	3500)
11	3250	(3000	3500)

Table A.10: Decisive frequency intervals used for failure definition in parameter estimation of the simulated signal: *Signal-5*.

This signal represents the glycogen signal in NMR spectroscopy, which is used in frequency-selective estimation in the thesis. Only the two glycogen peaks, peak 10 and 11, are of interest.

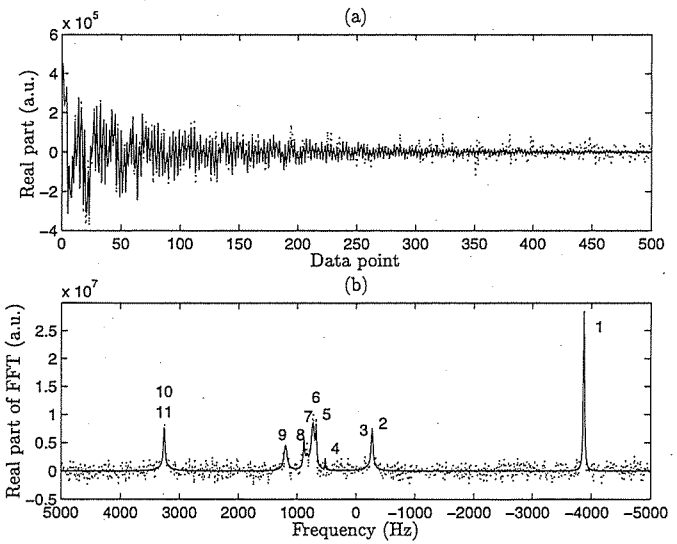


Figure A.5: Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 3 \times 10^4$) NMR signal, *Signal-5*, in the (a) time domain and (b) frequency domain.

Appendix B

Publications

Publications in international journals

[1] H. Chen, S. Van Huffel, C. Decanniere and P. Van Hecke, "A Signal Enhancement Algorithm for Time-Domain MR Data Quantification", *J. Magn. Reson.* **A 109**, pp.46-55, July 1994.

[2] S. Van Huffel, H. Chen, C. Decanniere and P. Van Hecke, "Algorithm for Time-Domain NMR Data Fitting Based on Total Least Squares", *J. Magn. Reson.* **A 110**, pp. 228-237, Sept. 1994.

[3] C. Decanniere, P. Van Hecke, F. Vanstapel, H. Chen, S. Van Huffel, C. van der Voort and D. van Ormondt, "Evaluation of Signal Processing Methods for the Quantification of Strongly Overlapping Peaks in ^{31}P NMR Spectra", *J. Magn. Reson.* **B 105**, pp.31-37, Nov. 1994.

[4] H. Chen, S. Van Huffel, and J. Vandewalle, "Bandpass filtering for exponential data fitting", *Signal Processing* Vol. 48, No. 2, pp. 135-154, 1996.

[5] K. Overloop, P. Van Hecke, F. Vanstapel, H. Chen, S. Van Huffel, A. Knijn and D. Van Ormondt, "Evaluation of signal processing methods for the quantification of a multi-exponential signal: the Glycogen ^{13}C-1 NMR signal", submitted to *NMR in Biomedicine*, 1996.

[6] H. Chen, S. Van Huffel, and J. Vandewalle, "Improved methods for exponential parameter estimation in the presence of known poles and noise", *ESAT-SISTA Report* TR-95-35, ESAT Lab., K. U. Leuven, Belgium, 1995, in review by *IEEE Trans. Signal Process.*

[7] H. Chen, S. Van Huffel, D. van Ormondt, and R. de Beer, "Parameter estimation with prior knowledge of known signal poles for the quantification of NMR data in the time domain", *ESAT-SISTA Report* TR-95-36, ESAT Lab., K. U. Leuven, Belgium, 1995, to appear in *J. Magn. Reson.*, **A**, Apr. 1996.

Publications in international conference proceedings

[1] S. Van Huffel, C. Decanniere, H. Chen, P. Van Hecke, "Resolution Improvement by Minimum Variance Signal Enhancement", *Proc. 3rd IMA Conference on Mathematics in Signal Processing*, University of Warwick, England, Dec. 15-17, 1992, J.G. McWhirter, ed., Oxford University Press, 1994, pp.197-207.

[2] S. Van Huffel, H. Chen, C. Decanniere, P. Van Hecke, "Improved Spectral Resolution of NMR Data by Minimum Variance Signal Enhancement", *Proc. 11th Annual Scientific Meeting of Society of Magnetic Resonance in Medicine*, Berlin, Germany, Aug. 8-14, 1992, p. 4272.

[3] S. Van Huffel, B. De Moor and H. Chen, "Relationships between structured and constrained TLS, with application to signal enhancement", *Proc. International Symposium on the Mathematical Theory of Networks and Systems (MTNS)*, Regensburg, Germany, Aug. 2-6, 1993, U. Helmke, R. Mennicken, J. Saurer, eds., Vol. II, Akademie Verlag, Berlin (Mathematical Research: Vol. 79), 1994, pp. 899-904.

[4] C. Decanniere, P. Van Hecke, H. Chen, S. Van Huffel, C. van der Voort and D. van Ormondt, "Comparison of Model Function Fitting algorithms for the Quantification of Strongly Overlapping Peaks in ^{31}P NMR Spectra", *Proc. 12th Annual Scientific meeting of Society of Magnetic Resonance in Medicine*, New York, USA, Aug. 14-20, 1993, vol. 2, p. 972.

[5] H. Chen, S. Van Huffel and J. Vandewalle, "Frequency-Selective Techniques Based on SVD, TLS and Bandpass Filtering", *Proc. SPIE Conference on Advanced Signal Processing: Algorithms, Architectures, and Implementations V*, San Diego, USA, July 24-27, 1994, Franklin T. Luk ed., Vol. 2296, pp. 601-611.

[6] K. Overloop, P. Van Hecke, H. Chen, S. Van Huffel, A. Knijn and D. Van Ormondt, " Evaluation of time- and frequency domain methods for the quantification of a multi-component line", *Proc. 13th Annual Scientific Meeting of the Society of Magnetic Resonance in Medicine*, San Francisco, USA, Aug. 6-12, 1994, p.1183.

[7] H. Chen, S. Van Huffel and J. Vandewalle, "Bandpass filtering for the HTLS estimation algorithm: design, evaluation and SVD analysis", *Proc. 3rd Int. Workshop on SVD and Signal Processing*, Leuven, Belgium, August 22-25, 1994, M. Moonen and B. De Moor eds., pp. 391-398.

[8] H. Chen, S. Van Huffel and J. Vandewalle, "Total Least Norm and its application to signal enhancement", *ESAT-SISTA Report* TR-94-12, ESAT Lab., K. U. Leuven, Belgium, 1995. *Proc. ProRISC/IEEE workshop*, Mierlo, The Netherlands, March 23-24, 1995. pp.45-50.

[9] H. Chen, S. Van Huffel and J. Vandewalle, "State-space based approximation methods for the harmonic retrieval problem in the presence of known signal poles", *ESAT-SISTA Report* TR-95-68, ESAT Lab., K. U. Leuven, Belgium, 1995. to appear in *ICASSP-96 proceedings.*

[10] H. Chen, S. Van Huffel, and J. Vandewalle, "Application of structured total least norm technique in spectral estimation", *ESAT-SISTA Report* TR-95-69, ESAT Lab., K. U. Leuven, Belgium, 1995, to appear in *Proc. EUSIPCO-96.*

List of Tables

3.1 Three ideal cases of filtering using a rectangular filter. 44

4.1 Parameter estimates and CR lower bound of the noisy ($\sigma_\nu = 2$, see Fig. 4.2) simulated ^{31}P NMR signal, *Signal-2*, using the HTLS, CA-HTLS and MV-HTLS algorithms. 77
4.2 The first $K+1(=6)$ singular values of the noisy ($\sigma_\nu = 2$, see Fig. 4.2) simulated ^{31}P NMR signal, *Signal-2*, processed by different enhancement procedures. 82
4.3 Bias±standard deviation values of all parameter estimates of the noisy ($\sigma_\nu = 2.5$) simulated ^{31}P NMR signal, *Signal-2* (see Fig. 4.3 through Fig. 4.4). 83
4.4 Typical numbers of iterations required by different enhancement procedures for the simulated noisy ($\sigma_\nu = 2$) ^{31}P NMR signal, *Signal-2*, with model order $K = 5$. 87
4.5 Real-world NMR signal-parameter estimates and their CR bounds[a] corresponding to the three exponentially damped sinusoidal components which fit the peaks PDE-P_i-PME. 90

5.1 Example 1: number of iterations, number of flops and values of η at convergence. The tolerance is $\epsilon = 10^{-15}$ for all the methods. The best weight, after several tests, is used in STLN Algorithm-1, which is $w = 10^{10}$ in this case. 111
5.2 Example 2: number of iterations, number of flops and values of η at convergence. The tolerance is $\epsilon = 10^{-15}$ for both algorithms. 112
5.3 Expected (averaged over 200 runs) Frobenius norm, corr_M and corr_v, of correction matrices and correction vectors given by the first column and the last row of Hankel data matrices. 115
5.4 (a): Number of non-convergence and failures, RMSE, bias and standard deviation of f_1. 117

5.4 (cont.) (b): RMSE, bias and standard deviation of d_1. 118
5.5 Number of flops required by different methods using Matlab averaged over 200 runs at SNR=25dB, 20dB, and 15dB. 119

6.1 Example 1: Bias ± standard deviation estimates of the parameters of the unknown peak at SNR=15dB, $N = 25$. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $M = 13, 17, 3$ is used respectively for the HTLS method, HTLS(PK) method, LP-based methods and the HTLN(PK) method. 132
6.2 Example 1: Number of flops required by different methods using Matlab averaged over 200 runs. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$. $M = 13, 17, 3$ is used respectively for the HTLS method, the HTLS(PK) method, LP-based methods and the HTLN(PK) method. 134

7.1 Number of flops used in MATLAB in the HTLS and HTLS(PK) algorithms per step when quantifying the noisy (SNR=10 dB or $\sigma_\nu = 0.2236$) simulated signal, *Signal-1*. Steps 1 through 3 of the HTLS method correspond to the steps outlined in Algorithm 1, given in Chapter 2, and steps 1 through 4 of the HTLS(PK) method correspond to the steps outlined in Algorithm 11, given in Chapter 6. The LS step uses the SVD method. 149
7.2 Parameter estimates and their CR bounds of the cluster of NAD and α-ATP peaks of a real-world ^{31}P NMR signal measured from human brain. 157

8.1 Variants of the HTLS method and some of their features. . . . 161

A.1 Exact parameter values of the simulated signal: *Signal-1*. . . . 166
A.2 Decisive frequency intervals used for failure definition in parameter estimation of the simulated signal: *Signal-1*. 167
A.3 Exact parameter values of the simulated signal: *Signal-2*. . . . 168
A.4 Decisive frequency intervals used for failure definition in parameter estimation of the simulated signal: *Signal-2*. 168
A.5 Exact parameter values of the simulated signal: *Signal-3*. . . . 170
A.6 Decisive frequency intervals used for failure definition in parameter estimation of the simulated signal: *Signal-3*. 170
A.7 Exact parameter values of the simulated signal: *Signal-4*. . . . 172
A.8 Decisive frequency interval used for failure definition in parameter estimation of the simulated signal: *Signal-4*. Only the 1st peak of this signal is of interest. 172

A.9 Exact parameter values of the simulated signal: *Signal-5*. . . . 174
A.10 Decisive frequency intervals used for failure definition in parameter estimation of the simulated signal: *Signal-5*. 174

List of Figures

1.1 An example of an *FID* signal measured from an NMR experiment. (a) Time-domain or measurement-domain signal, (b) Signal spectrum. a.u. means arbitrary unit. 5

1.2 Typical parameter estimation procedure. 7

1.3 Relationships between the new techniques developed in this thesis from an application's point of view. 12

3.1 An example of FIR filtering using the filter matrix (a) $\mathbf{H}_{\mathrm{FIR}c}$ and (b) $\mathbf{H}_{\mathrm{FIR}t}$. The unfiltered data $\mathbf{x}$ is the first sinusoid of *Signal-4*, i.e., $e^{-0.02n\Delta t - j2\pi 0.2n\Delta t}$. The filter is the bandpass rectangular filter as used in Example 2 in the simulation section of this chapter. 33

3.2 (a) $|\beta_{12}|$ and $|\tilde{\beta}_{12}|$ (b) $|\eta_{11}|$, $|\eta_{22}|$ and $|\eta_{12}|$ (c) σ_1, σ_2, $\tilde{\sigma}_1$ and $\tilde{\sigma}_2$ of a two-peak signal with $f_1 = -0.2$ Hz, f_2 variable, $d_1 = 0.03$ rad/s, $d_2 = 0.05$ rad/s, and $c_1 = c_2 = 1$. $f_l = f_1 - 0.1$ Hz, $f_u = f_1 + 0.1$ Hz, $q = 99$, $L = 62$, $M = 60$, $N = 121$. 43

3.3 Example 1: (a) Failure rate versus σ_ν. (b) RMSE of $f_3 = 0.1$ Hz and (c) Relative RMSE of $d_3 = 0.02\pi$ rad/s versus SNR at peak 3. 56

3.4 Example 2: (a) Noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.5$) spectra of *Signal-4*. (b) The frequency response of the filters used in the prefiltered HTLS method. 57

3.5 Example 2: (a) Failure rate versus σ_ν. (b) RMSE of f_1 and (c) Relative RMSE of d_1 versus SNR at peak 1. 58

3.6 Example 2: Failure rate versus varying filter center. $\sigma_\nu = 0.5$. . 59

3.7 Example 2: Failure rate versus varying amplitude $|c_1|$ when other signal parameters are fixed. $\sigma_\nu = 0.5$. 60

3.8 Example 2: (a) Failure rate, (b) RMSE of f_1 and (c) Relative RMSE of d_1 versus varying damping factor d_1 when other signal parameters are fixed. $\sigma_\nu = 0.5$. 61

3.9 Example 3: (a) Noise-free (full line) and noisy (dotted line, $\sigma_\nu = 3 \times 10^4$) spectra of *Signal-5*. (b) The frequency response of the filters used in the prefiltered HTLS method. 62

3.10 Example 3: (a) Failure rate versus σ_ν. Relative RMSE of (b) a_{10} and (c) a_{11} versus SNR at peak 10 and 11, respectively. . . 64

3.11 Example 3: Failure rate versus M/N. $\sigma_\nu = 3 \times 10^4$. 65

4.1 The singular values $\hat{\sigma}_{K-1}, \hat{\sigma}_K$ and $\hat{\sigma}_{K+1}$ (i.e., $\hat{\sigma}_4, \hat{\sigma}_5$ and $\hat{\sigma}_6$) of the $(129 - \hat{M}) \times \hat{M}$ Hankel matrix built up with 128 data points of a noisy ($\sigma_\nu = 0.1$) simulated ^{31}P NMR signal, *Signal-2*, with model order $K = 5$. 75

4.2 Real part of the FFT of the reconstructed simulated ^{31}P NMR signal, *Signal-2*, (dashed line) as estimated by (a) HTLS, (b) CA-HTLS, and (c) MV-HTLS from the same set of noisy data ($\sigma_\nu = 2$, dotted line). The full line represents the exact signal. . 78

4.3 Failure rate of HTLS, nCA-HTLS, nMV-HTLS, and MV-nCA-HTLS versus noise standard deviation. Here the MV method does not converge within 100 iterations. 79

4.4 Relative RMSE values of a_5 (averaged over good runs) as estimated by HTLS, nCA-HTLS, nMV-HTLS, and MV-nCA-HTLS. The thick full line is the CR bound. $\sigma_\nu = 2$ corresponds to SNR=15.58 dB at peak 5. 81

4.5 Failure rate of (a) CA-HTLS and (b) MV-HTLS versus σ_ν for different estimated model orders K_{est}. 84

4.6 "Cleaned-up" spectra (full line) of the same set of noisy data ($\sigma_\nu = 2$, dotted line) using the nMV, nCA and MV-nCA enhancement procedures. 86

4.7 (a) ^{31}P-NMR FID signal of perfused rat liver. (b) The corresponding NMR spectrum (after omission of the first five data points). 88

4.8 Spectra of the exponentially damped sinusoidal components that reconstruct the three overlapping peaks PDE-P_i-PME of the real-world NMR signal, as computed by (a) HTLS (b) CA-HTLS and (c) MV-HTLS. 89

5.1 $N = 81$, SNR=15dB, $\sigma_\nu = 0.1257$, $M = K + 1$. (a) Original noisy signal spectrum; (b) STLN-enhanced signal spectrum; (c) residual spectrum. 114

6.1 Example 1: (a) Failure rate versus σ_ν. (b) RMSE of $f_2 = 0.22$ Hz and (c) Relative RMSE of $d_2 = 0.02$ rad/s versus the SNR at this peak. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$. $M = 13, 17, 3$ is used respectively for the HTLS method, HTLS(PK) method, LP-based methods and the HTLN(PK) method. HTLN(PK) has 2 non-convergent cases within 100 iterations of the STLN Algorithm-2 at SNR=10dB. 133

6.2 Example 1: Failure rate of each method versus M/N of the data matrix. $f_1 = 0.2$ Hz, $d_1 = 0.01$ rad/s are known *a priori*. $N = 25$, SNR=15dB or $\sigma_\nu = 0.1257$. 135

6.3 Example 2: Failure rate of each method versus M/N of the data matrix. $f_4 = 353$ Hz, $d_4 = 117$ rad/s are known *a priori*. $N = 128$, $\sigma_\nu = 2$. 136

6.4 Example 2: (a) Failure rate versus σ_ν. Relative RMSE of the estimates of (b) $d_3 = 197$ rad/s and (c) $d_5 = 808$ rad/s versus the peak SNR. $f_4 = 353$ Hz, $d_4 = 117$ rad/s are known *a priori*. $N = 128$. $M = 64, 20, 85$ is used respectively for the HTLS, HTLS(PK), LCTLS-FLP and LCTLS-BLP methods. 137

6.4 (cont.) Example 2: Relative RMSE of (d) $a_3 = 6.0$ a.u., (e) $a_4 = 2.8$ a.u., and (f) $a_5 = 17.0$ a.u. versus the peak SNR. . . 138

7.1 Frequency response of FIR filters H(z) for *Signal-1*, when the given prior knowledge is (a) $f_{1\text{PK}} = f_1 + \Delta f$, $d_{1\text{PK}} = d_1$, and (b) $f_{1\text{PK}} = f_1$, $d_{1\text{PK}} = d_1 + \Delta d$. The parallel vertical full lines denote the position of the two sinusoids in *Signal-1*. When Δf increases from 0 Hz till 0.1 Hz or Δd increases from 0 rad/s till $0.1 \times 2\pi$ rad/s, the filter frequency response shifts in the direction of arrows. 144

7.2 *Signal-1*: (a) Failure rate versus σ_ν, (b) RMSE of the estimates of f_2 versus SNR. Prior knowledge of $f_1 = 0.2$ Hz and $d_1 = 0.01$ rad/s is used in the HTLS(PK) method. 146

7.3 *Signal-1*: (a) Failure rate versus σ_ν, (b) Relative RMSE of the estimates of d_2 versus SNR. HTLN(PK) has 2 non-convergent cases in the STLN iterations at SNR=10dB, and 22 non-convergent cases at SNR=5dB. Prior knowledge used is $f_1 = 0.2$ Hz and $d_1 = 0.01$ rad/s. 148

7.4 (Relative) RMSE of the estimates of (a) a_4 (b) ϕ_4 versus Δf_4, obtained by the HTLS(PK) method. The imposed prior knowledge is $f_4 = 353 + \Delta f_4$ Hz, $d_4 = 117$ rad/s. 150

7.4 (cont.) (Relative) RMSE of the estimates of (c) f_5 (d) d_5 versus Δf_4, obtained by the HTLS(PK) method. The imposed prior knowledge is $f_4 = 353 + \Delta f_4$ Hz, $d_4 = 117$ rad/s. 151

7.5 (Relative) RMSE of the estimates of (a) a_4 (b) ϕ_4 versus Δd_4, obtained by the HTLS(PK) method. The imposed prior knowledge is $f_4 = 353$ Hz, $d_4 = 117 + \Delta d_4$ rad/s. 152

7.5 (cont.) (Relative) RMSE of the estimates of (c) f_5 (d) d_5 versus Δd_4, obtained by the HTLS(PK) method. The imposed prior knowledge is $f_4 = 353$ Hz, $d_4 = 117 + \Delta d_4$ rad/s. 153

7.6 FFT spectrum of ^{31}P NMR signal of human brain (after omission of the first 15 data points and correction of zero-order phase $\phi^{(0)} = 231.81°$). 155

7.7 Estimates of f, d, a and ϕ of the cluster of NAD ($*$) and α-ATP (o and $\times$) by the (a) HTLS ($\cdots$) and (b) HTLS(PK) (—) methods versus the model order used. Prior knowledge used in the HTLS(PK) method is $f = -203$ Hz, $d = 26$ rad/s and $f = -186$ Hz, $d = 26$ rad/s for the α-ATP doublet. 156

7.8 Estimates of f, d, a and ϕ of the cluster of NAD ($*$) and α-ATP (o and $\times$) by the HTLS(PK) methods versus (a) Δf, i.e., prior knowledge used is $f = -203 + \Delta f$ Hz, $d = 26$ rad/s and $f = -186 + \Delta f$ Hz, $d = 26$ rad/s for the α-ATP doublet; (b) Δd, i.e., prior knowledge used is $f = -203$ Hz, $d = 26 + \Delta d$ rad/s and $f = -186$ Hz, $d = 26 + \Delta d$ rad/s for the α-ATP doublet. . 158

A.1 Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.1$) signal, *Signal-1*, in the (a) time domain and (b) frequency domain. 167

A.2 Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 1.5$) NMR signal, *Signal-2*, in the (a) time domain and (b) frequency domain. 169

A.3 Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.1$) NMR signal, *Signal-3*, in the (a) time domain and (b) frequency domain. 171

A.4 Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 0.5$) signal, *Signal-4*, in the (a) time domain and (b) frequency domain. 173

A.5 Simulated noise-free (full line) and noisy (dotted line, $\sigma_\nu = 3 \times 10^4$) NMR signal, *Signal-5*, in the (a) time domain and (b) frequency domain. 175

Bibliography

[1] R. Roy and T. Kailath, "ESPRIT — Estimation of Signal Parameters via Rotational Invariance Techniques", *Opt. Eng.*, Vol. 29, No.4 pp. 296-313, Apr. 1990.

[2] A. Swindlehurst, B. Ottersten, R. Roy and T. Kailath, "Multiple Invariance ESPRIT", *IEEE Trans. Signal Process.*, Vol. SP-40, No. 3, Mar. 1992.

[3] R. De Beer and D. Van Ormondt, *Analysis of NMR data using time-domain fitting procedures*, in "In-Vivo Magnetic Resonance Spectroscopy I: Probeheads, Radiofrequency Pulsed, Spectrum Analysis", (M. Rudin, Guest Ed.), NMR Basic Principles and Progress series, Vol. **26**, p. 201, Springer-Verlag, Berlin/Heidelberg, 1992.

[4] R. De Beer, D. Van Ormondt, W. W. F. Pijnappel and J. W. C. van der Veen, "Quantitative analysis of magnetic resonance signals in the time domain", *Israel J. Chem.* Vol. 28, pp.249-261, 1988.

[5] E. M. Dowling, R. D. DeGroat, and D. A. Linebarger, "Exponential parameter estimation in the presence of known components and noise", *IEEE Trans. Antennas and Propagat.*, Vol. AP-42, No. 5, pp. 590-599, May 1994.

[6] G. H. Golub and C. F. Van Loan, *Matrix computations*, (2nd ed.), The Johns Hopkins Univ. Press, Baltimore, 1989.

[7] M. Alonso, E. J. Finn, *Fundamental University Physics, Vol. III, Quantum and Statistical Physics*, Addison-Wesley, Reading Massachusetts, 1968.

[8] M. Alonso, E. J. Finn, *Fundamental University Physics, Vol. III, Quantum and Statistical Physics*, Addison-Wesley, Reading Massachusetts, 1968.

[9] D. G. Gadian, *Nuclear Magnetic Resonance and its applications to living systems*, Clarendon Press, Oxford, 1982.

[10] R. R. Ernst, G. Bodenhausen, A. Wokaun, *Principles of Nuclear Magnetic Resonance in One and Two Dimensions*, Clarendon Press, Oxford, 1987.

[11] R. E. Andrew, G. Bydder, J. Griffiths, R. Iles, P. Styles, *Clinical Magnetic Resonance: Imaging and Spectroscopy*, John Wiley & Sons, Chichester, 1990.

[12] A. van den Boogaart, *The use of signal processing algorithms to obtain biochemically relevant parameters from magnetic resonance data sets*, Ph. D. thesis, CRC Biomedical Magnetic Resonance Research Group, division of Biochemistry, Dept. Cellular and Molecular Sciences, University of London at St. George's Hospital Medical School, Jun. 1995.

[13] C. Decanniere, *In-vivo localized NMR spectroscopy at high field strength*, Ph. D. thesis, Biomedische NMR Eenheid, Faculteit Geneeskunde, K. U. Leuven, May 1993.

[14] G. Majda, W. F. Strauss, and M. Wei, "Computation of exponentials in transient data", *IEEE Trans. Antennas and Propagat.*, Vol. AP-37, No. 10, pp. 1284-1290, 1989.

[15] N. Gharasallah, E. J. Rothwell, K. M. Chen, and D. P. Nyquist, "Identification of the natural resonance frequencies of a conducting sphere from a measured transient response", *IEEE Trans. Antennas and Propagat.*, Vol. AP-38, pp. 141-143, Jan. 1990.

[16] J. W. C. Van der Veen, R. de Beer, P. R. Lyuten, and D. van Ormondt, "Accurate quantification of *in vivo* ^{31}P NMR signals using the variable projection method and prior knowledge", *Magn. Res. Med.* 6, pp.92-98, 1988.

[17] G. H. Golub and V. Pereyra, "The Differentiation of Pseudo-Inverses and Nonlinear LS Problems whose Variables Separate", *SIAM J. Numer. Anal.* Vol. 10, pp.413-432, Apr. 1973.

[18] H. Barkhuijsen, R. De Beer, A. C. Drogendijk, D. Van Ormondt, and J. W. C. van der Veen, in *Proc. International School of Physics 'Enrico Fermi' on the 'Physics of NMR Spectroscopy in Biology and Medicine'* (B. Maraviglia, Ed.) pp. 313, Italian Physical Society, 1988.

[19] Y. Bresler and A. Macovski, "Exact maximum likelihood parameter estimation of superimposed exponential signals in noise", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-34, No.5, pp. 1081-1089, Oct. 1986.

[20] D. W. Tufts and R. Kumaresan, "Singular value decomposition and improved frequency estimation using linear prediction," *IEEE Trans. Acoust. Speech Signal Process.*, Vol ASSP-30, No. 4, pp. 671-675, Aug. 1982.

[21] R. Kumaresan and D. W. Tufts, "Estimating the parameters of exponentially damped sinusoids and pole-zero modeling in noise," *IEEE Trans. Acoust. Speech Signal Process.*, Vol ASSP-30, No. 6, pp. 833-840, Dec. 1982.

[22] Y. Hua and T. K. Sarkar, "Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise," *IEEE Trans. Acoust. Speech Signal Process.*, Vol ASSP-38, No. 5, pp. 814-824, May 1990.

[23] S. Y. Kung, K.S. Arun, and D.V. Bhaskar Rao, "State-space and SVD-based approximation methods for the harmonic retrieval problem," *J. Opt. Soc. Amer.*, Vol. 73, No. 12, pp. 1799-1811, Dec. 1983.

[24] S. Van Huffel and J. Vandewalle, *The total least squares problem : computational aspects and analysis*, Frontiers in Applied Mathematics series, Vol. 9, SIAM, Philadelphia, 1991.

[25] M. A. Rahman and K. Yu, "Total least squares approach for frequency estimation using linear prediction," *IEEE Trans. Acoust. Speech Signal Process.*, Vol ASSP-35, No. 10, pp. 1440-1454, Oct. 1987.

[26] S. Van Huffel, L. Aerts, J. Bervoets, J. Vandewalle, C. Decanniere and P. Van Hecke, "Improved quantitative time-domain analysis of NMR data by total least squares," *Proc. EUSIPCO-92*, Vol. III, pp. 1721-1724, Aug. 24-27, 1992.

[27] S. Van Huffel, H. Chen, C. Decanniere and P. Van Hecke, "Algorithm for Time-Domain NMR Data Fitting Based on Total Least Squares," *J. Magn. Res.* A, Vol.110, pp. 228-237, Sept. 1994.

[28] Y. Hua and T. K. Sarkar, "On SVD for Estimating Generalized Eigenvalues of Singular Matrix Pencil in Noise", *IEEE Trans. Signal Process.* Vol. 39, No.4, pp.892-900, Apr. 1991.

[29] B. De Moor, "The singular value decomposition and long and short spaces of noisy matrices", *IEEE Trans. Signal Process.*, Vol. SP-41, No.9, pp.2826-2838, Sept. 1993.

[30] R. A. Horn, and C. R. Johnson, *Matrix analysis*, Cambridge University Press, New York, 1988.

[31] THE MATHWORKS, INC., "PRO-MATLAB for Unix Computers", Version 4.2c, South Natick, MA 01760, 1994.

[32] W. W. Pijnappel, A. Van Den Boogaart, R. de Beer and D. Van Ormondt, "SVD-based quantification of magnetic resonance signals", *J. Magn. Reson.*, **97**, pp. 122-134, 1992.

[33] H. Chen, S. Van Huffel, and J. Vandewalle, "Bandpass filtering for exponential data fitting with known frequency region of interest", *Signal Processing*, Vol. 48, No. 2, pp. 135-154, 1996.

[34] K. Overloop, P. Van Hecke, F. Vanstapel, H. Chen, S. Van Huffel, A. Knijn and D. Van Ormondt, "Evaluation of signal processing methods for the quantification of a multi-exponential signal: the Glycogen ^{13}C-1 NMR signal", submitted to *NMR in Biomedicine*, 1996.

[35] K. Overloop, P. Van Hecke, H. Chen, S. Van Huffel, A. Knijn and D. Van Ormondt, " Evaluation of time- and frequency domain methods for the quantification of a multi-component line", *Proc. 13th Annual Scientific Meeting of the Society of Magnetic Resonance in Medicine*, San Francisco, USA, Aug. 6-12, 1994, p.1183.

[36] H. Chen, S. Van Huffel and J. Vandewalle, "Bandpass filtering for the HTLS estimation algorithm: design, evaluation and SVD analysis", *Proc. 3rd Int. Workshop on SVD and Signal Processing*, Leuven, Belgium, Aug. 22-25, 1994, M. Moonen and B. De Moor eds., pp. 391-398.

[37] H. Chen, S. Van Huffel and J. Vandewalle, "Frequency-Selective Techniques Based on SVD, TLS and Bandpass Filtering", *Proc. SPIE Conference on Advanced Signal Processing: Algorithms, Architectures, and Implementations V*, San Diego, USA, Jul. 24-27, 1994, Franklin T. Luk ed., Vol. 2296, pp. 601-611.

[38] H. Chen, S. Van Huffel, C. Decanniere and P. Van Hecke, "A Signal Enhancement Algorithm for Time-Domain MR Data Quantification", *J. Magn. Reson.* **A 109**, pp.46-55, Jul. 1994.

[39] S. Van Huffel, C. Decanniere, H. Chen, P. Van Hecke, "Resolution Improvement by Minimum Variance Signal Enhancement", *Proc. 3rd IMA Conference on Mathematics in Signal Processing*, University of Warwick, England, Dec. 15-17, 1992, J.G. McWhirter, ed., Oxford University Press, 1994, pp.197-207.

[40] S. Van Huffel, H. Chen, C. Decanniere, P. Van Hecke, "Improved Spectral Resolution of NMR Data by Minimum Variance Signal Enhancement", *Proc. 11th Annual Scientific Meeting of Society of Magnetic Resonance in Medicine*, Berlin, Germany, Aug. 8-14, 1992, p. 4272.

[41] C. Decanniere, P. Van Hecke, F. Vanstapel, H. Chen, S. Van Huffel, C. van der Voort and D. van Ormondt, "Evaluation of Signal Processing Methods for the Quantification of Strongly Overlapping Peaks in ^{31}P NMR Spectra", *J. Magn. Reson.* **B 105**, pp.31-37, Nov. 1994.

[42] C. Decanniere, P. Van Hecke, H. Chen, S. Van Huffel, C. van der Voort and D. van Ormondt, "Comparison of Model Function Fitting algorithms for the Quantification of Strongly Overlapping Peaks in ^{31}P NMR Spectra", *Proc. 12th Annual Scientific meeting of Society of Magnetic Resonance in Medicine*, New York, USA, Aug. 14-20, 1993, Vol. 2, p. 972.

[43] S. Van Huffel, B. De Moor and H. Chen, "Relationships between structured and constrained TLS, with application to signal enhancement", *Proc. International Symposium on the Mathematical Theory of Networks and Systems (MTNS)*, Regensburg, Germany, Aug. 2-6, 1993, U. Helmke, R. Mennicken, J. Saurer, eds., Vol. II, Akademie Verlag, Berlin (Mathematical Research: Vol. 79), 1994, pp. 899-904.

[44] H. Chen, S. Van Huffel and J. Vandewalle, "Total Least Norm and its application to signal enhancement", *ESAT-SISTA Report* TR-94-12, ESAT Lab., K. U. Leuven, Belgium, 1995. *Proc. ProRISC/IEEE workshop*, Mierlo, The Netherlands, Mar. 23-24, 1995. pp.45-50.

[45] H. Chen, S. Van Huffel, and J. Vandewalle, "Application of structured total least norm technique in spectral estimation", *ESAT-SISTA Report* TR-95-69, ESAT Lab., K. U. Leuven, Belgium, 1995, to appear in *Proc. EUSIPCO-96.*

[46] H. Chen, S. Van Huffel, and J. Vandewalle, "Improved methods for exponential parameter estimation in the presence of known poles and noise", *ESAT-SISTA Report* TR-95-35, ESAT Lab., K. U. Leuven, Belgium, 1995, in review by *IEEE Trans. Signal Process.*

[47] H. Chen, S. Van Huffel and J. Vandewalle, "State-space based approximation methods for the harmonic retrieval problem in the presence of known signal poles", *ESAT-SISTA Report* TR-95-68, ESAT Lab., K. U. Leuven, Belgium, 1995, to appear in *Proc. ICASSP-96.*

[48] H. Chen, S. Van Huffel, D. van Ormondt, and R. de Beer, "Parameter estimation with prior knowledge of known signal poles for the quantification of NMR data in the time domain", *ESAT-SISTA Report* TR-95-36, ESAT Lab., K. U. Leuven, Belgium, 1995, to appear in *J. Magn. Reson.*, A, Apr. 1996.

[49] A. Knijn, R. de Beer, A. F. Mehlkopf, and D. van Ormondt, "Frequency-Selective Quantification in the Time Domain", *J. Magn. Res.* **97**, pp. 444-450, 1992.

[50] J. Tang and J. R. Norris, "LP-ZOOM, a linear prediction method for local spectral analysis of NMR signals", *J. Magn. Res.*, **79**, pp. 190-196, 1988.

[51] F. Hu, T. K. Sarkar and Y. Hua, "Utilization of Bandpass Filtering for the Matrix Pencil Method", *IEEE Trans. Signal Process.*, Vol. SP-41, Jan. 1993, pp. 442-446.

[52] E. Oran Brigham, *The Fast Fourier Transform and its Applications*, Englewood Cliffs, NJ: Prentice-Hall, 1988.

[53] W. W. F. Pijnappel, "Quantification of 1-D and 2-D magnetic resonance spectroscopic data", Ph. D. thesis, Feb. 1991, Technical University of Delft, Faculty of Applied Physics, The Netherlands.

[54] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.

[55] K. Overloop, A. Steegmans, M. Waebens, P. Van Hecke, F. Vanstapel and W. Stalmans, "In-vitro NMR visibility of the ^{13}C-1 of the glucosyl residues in hepatica glycogen", *S.M.R.M. 10th Annual Meeting Abstract Book*, p. 1194, 1991.

[56] J. A. Cadzow, "Signal Enhancement–A Composite Property Mapping Algorithms", *IEEE Trans. Acoust. Speech Signal Process.*, Vol. ASSP-36, No. 1, Jan. 1988.

[57] J. A. Cadzow and D. M. Wilkes, "Signal Enhancement and the SVD", *Proc. 2nd International workshop on SVD and Signal Process.*, Rhode Island, USA, Jun. 1990.

[58] J. A. Cadzow and D. M. Wilkes, "Enhanced sinusoidal and exponential data modeling", *SVD and signal process., II Algorithms, Analysis and Applications*, R. J. Vaccaro (Ed.), Elsevier Science Publishers B. V., 1991.

[59] A. Diop, A. Briguet and D. Graveron-Demilly, "Automatic *in vivo* NMR data processing based on an enhancement procedure and LP method", *Magn. Res. Med.*, **27**, pp.318-328, 1992.

[60] Yung-Ya Lin and Lian-Pin Hwang, "NMR signal enhancement based on matrix property mappings", *J. Magn. Res.*, **103**, pp.109-114, 1993.

[61] S. Van Huffel, "Enhanced resolution based on minimum variance estimation and exponential data modeling," *Signal Process.* 33, pp.333-355, 1993.

[62] H. Chen, *Enhancement algorithms for NMR signal processing*, master thesis, ESAT Lab., K. U. Leuven, Jul. 1992.

[63] D. Tufts, R. Kumaresan and I. Kirsteins, "Data adaptive signal estimation by singular value decomposition of a data matrix", *Proc. IEEE*, Vol. 70, No. 6, Jun. 1982, pp. 684-685.

[64] B. De Moor, " Total least squares for affinely structured matrices and the noisy realization problem", *ESAT-SISTA report* TR-93-21, ESAT Lab., K. U. Leuven, Belgium, 1993. Submitted to the IEEE Transactions on Signal Processing.

[65] I. Dologlou and G. Carayannis, "Physical interpretation of signal reconstruction from reduced rank matrices", *IEEE Trans. Signal Process.*, Vol. SP-39, No. 7, pp. 1681-1682, Jul. 1991.

[66] I. Dologlou and S. Van Huffel, submitted to *IEEE Trans. Signal Process.*, 1996.

[67] Theagenis J. Abatzoglou, Jerry M. Mendel and Gail A. Harada, "The constrained total least squares technique and its applications to harmonic super-resolution", *IEEE Trans. Signal Process.*, Vol. SP-39, No.5, pp. 1070-1087, May 1991.

[68] B. De Moor, "Structured total least squares and L_2 approximation problems", *ESAT-SISTA report* TR-92-33J, ESAT Lab., K. U. Leuven, Belgium, 1992. *Lin. Alg. Appl.* 188-189, pp. 163-207, Jul. 1993.

[69] J. B. Rosen, H. Park and J. Glick, "Total Least Norm formulation and solution for structured problems," *SIAM J. Matrix Anal. Appl*, vol. 17, No. 1, Jan. 1996.

[70] H. Park, J. B. Rosen, and S. Van Huffel, "Structure preserving total least squares method and its application to parameter estimation," *SIAM J. Matrix Anal. Appl.*, Vol. 17, Jan. 1996.

[71] S. Van Huffel, H. Park and J. Ben Rosen, "Formulation and solution of structured total least norm problems for parameter estimation", ESAT/SISTA report TR 95-08, Dept. of Electr. Eng., Katholieke Universiteit Leuven, Belgium, Feb. 1995, in review by *IEEE Trans. on Signal Process.*.

[72] P. Lemmerling, B. De Moor and S. Van Huffel, "On the equivalence of constrained total least squares and structured total least squares", *ESAT-SISTA report* TR-95-55, ESAT Lab., K. U. Leuven, Belgium, 1995. Submitted to *IEEE Trans. Signal Process.*

[73] R. Fletcher, *Practical methods of optimization,* John Wiley & Sons, New York, 1987.

[74] P. E. Gill, W. Murray and M. H. Wright, *Practical optimization,* Academic Press, London, 1981.

[75] A. Bjorck, *Least squares methods,* in "Handbook of numerical analysis", Vol. 1, Elsevier, North Holland, 1988.

[76] C. F. Van Loan, "On the method of weighting for equality constrained least squares problems", *SIAM J. Numer. Anal.*, Vol. 22, pp. 851-864, 1985.

[77] P. Lemmerling and S. Van Huffel, "Super-linear algorithms for solving the structured total least norm problem", *ESAT-SISTA report* TR-95-44, ESAT Lab., K. U. Leuven, Belgium, 1995. Submitted to *SIAM J. Matrix Anal. Appl.*

[78] S. Haykin, *Adaptive filter theory*, (2nd ed.), Prentice-Hall, inc., Englewood cliffs, New Jersey, 1991.

[79] E. M. Dowling, R. D. DeGroat, and D. A. Linebarger, "Total least squares with linear constraints," *Proc. ICASSP'92*, Vol. 5, pp. 341-344, 1992.

[80] R. Carriere and R. L. Moses, "High resolution radar target modeling using a modified Prony estimator," *IEEE Trans. Antennas Propagat.*, Vol. 40, pp. 13-18, Jan. 1992.

[81] B.P.O. van Tongeren, R. de Beer, A.F.Mehlkopf, and D. van Ormondt, "Prior knowledge modeling for MR signal processing," *Proc. IEEE Benelux & ProRISC Workshop on Circuits, Systems and Signal Processing*, pp.257-262, 1993.