KATHOLIEKE UNIVERSITEIT LEUVEN
FAKULTEIT DER TOEGEPASTE WETENSCHAPPEN
DEPARTEMENT ELEKTROTECHNIEK
Afdeling ESAT
Kardinaal Mercierlaan 94, B-3030 Leuven (Heverlee), België

ANALYSIS OF THE TOTAL LEAST SQUARES PROBLEM AND ITS USE IN PARAMETER ESTIMATION

Jury :
Prof. Dr. ir. J. Berlamont, voorzitter
(academisch secretaris)
Prof. Dr. ir. J. Vandewalle, promotor
Prof. Dr. ir. A. Haegemans, copromotor
Prof. Dr. ir. A. Bultheel
Dr. ir. A. Barbé
Dr. ir. P. Sas
Dr. ir. P. Van Dooren (PRLB)

PREFACE

Without the enthusiastic support of many people, this thesis would not exist today. First of all, I want to express my heartfelt thanks to my promotor Prof. J. Vandewalle. His careful guidance, suggestions and continuous stimulation have been invaluable for the successful completion of this research work. Working with him has broadened my perspective and strengthened my confidence in my own abilities. I shall always be grateful for the inspiration and personal education he gave me.

Furthermore, I wish to thank my copromotor Prof. A. Haegemans. Her valuable advice and interesting suggestions have contributed to the success of this thesis. At the same time, I am grateful to her and Prof. A. Bultheel for the careful review of this manuscript.

A strong multi-disciplinary interaction has been vital to the success of this research. I have been very pleased to enjoy such an interaction in the creative environment of the ESAT Laboratory at Leuven. The opportunity to work in the group of system theory has been very stimulating for me. Undoubtedly, the open and cordial atmosphere created by the people in that group Bart De Moor, Dirk Callaerts, Shaohua Tan, André Barbé, Jan Vanderschoot and Chen Jiande, has encouraged my enthusiasm to carry on to the end. I particularly would like to thank Bart De Moor for our interesting discussions. Especially his optimism and utmost willingness to answer my questions whensoever have been admirable.

Also important have been the contributions of several people from both the academic and industrial world to the obtained remarks. First of all, I am highly undebted to Prof. G.H. Golub from Stanford University for his valuable advice and stimulating remarkt. In this respect I also mention Prof. J.L. Willems of the department of medical informatics, as well as Prof. M. Ch. De Roo and A. Vandecruys of the department of nuclear medicine, at the Gasthuisberg University hospital in Leuven. I am very grateful for their in-depth insights in renography and medical diagnostics and for the valuable clinical data I received from them. In this respect, thanks are also due to Dr. ir. J. Leuridan of Leuven Measurement &

Systems for providing interesting data sets of measured frequency spectra.

Finally, I want to express my special feelings of thanks to my parents for orienting me in this direction. And last but not least, I am very grateful to my dear Johan for his endless patience and for his heartfelt encouragements during the realization of my life work.

Sabine Van Huffel

Dedicated

to

Johan,

little Eva

and ... God.

TABLE OF CONTENTS

Introduction 1

Outline of the thesis 7

Notations and preliminaries 10

CHAPTER 1 :
THE TOTAL LEAST SQUARES PROBLEM : FORMULATION, EXTENSIONS AND ALGORITHM 15

1.1. Introduction 15
1.2. Formulation and solution of the generic one-dimensional TLS problem with unique solution 18
1.3. A geometric comparison of the TLS problem with respect to the LS problem 22
1.3.1. Viewed from the column space 22
1.3.2. Viewed from the row space 23
1.4. The generic multidimensional TLS problem with unique solution 25
1.5. Non-uniqueness of the generic TLS solution 29
1.6. The nongeneric TLS problem and solution 32
1.6.1. The one-dimensional case 32
1.6.2. The multidimensional case 42
1.7. The TLS solution in case of exactly known columns 54
1.8. Outline of the extended TLS algorithm 55
1.8.1. No exactly known columns 55
1.8.2. With exactly known columns 60
1.9. Concluding remarks 61

CHAPTER 2 :
PROPERTIES OF THE TLS PROBLEM AND THEIR RELATION TO THOSE OF THE LS PROBLEM 63

2.1. Introduction 63
2.2. Solution properties 64
2.3. Residual properties 72

2.4. Approximation effort properties 77
2.5. Subspace properties 84
2.6. Concluding remarks 89

CHAPTER 3 :
THE SENSITIVITY OF THE TLS PROBLEM AND ITS RELATION TO THE LS PROBLEM 91

3.1. Introduction 91
3.2. Sensitivity properties of the solution 92
3.2.1. Condition of the problem 92
3.2.2. Upper bounds of the LS solution perturbation 95
3.2.3. Upper bounds of the TLS solution perturbation 102
3.3. Sensitivity properties of the SVD 105
3.4. Simulation results 115
3.5. Concluding remarks 121

CHAPTER 4 :
THE STATISTICAL PROPERTIES OF THE TLS PROBLEM 123

4.1. Introduction : The errors-in-variables model 123
4.2. Parameter estimators 126
4.3. Strong consistency properties 128
4.4. Asymptotic distributions 131
4.5. Practical relevance and simulation results 135
4.6. Concluding remarks 141

CHAPTER 5 :
RELATIONSHIPS BETWEEN TLS ESTIMATION AND CLASSICAL LINEAR REGRESSION 143

5.1. Introduction 143
5.2. Regression versus errors-in-variables model 144
5.3. Unified SVD approach of the classical regression techniques 146
5.3.1. Multicollinearities 146
5.3.2. Biased regression estimators 148
5.4. Algebraic comparison of the classical regression estimators w.r.t. the TLS estimator 151
5.5. Concluding remarks 156

CHAPTER 6 :
SUBSET SELECTION USING THE TLS APPROACH IN MULTICOLLINEARITY PROBLEMS WITH ERRORS IN THE VARIABLES 159

6.1. Introduction 159
6.2. Subset selection algorithms 162
6.3. Overall comparison of subset selection algorithms 165
6.4. Concluding remarks 178

CHAPTER 7 :
USE OF TLS IN IMPULSE RESPONSE ESTIMATION BY DISCRETE DECONVOLUTION 179

7.1. Introduction 179
7.2. Discrete deconvolution 181
7.3. Deconvolution of the renogram 183
7.3.1. The renal retention function 184
7.3.2. The use of TLS in renography 187
7.3.2.1. Noise level determination 188
7.3.2.2. Scaling 189
7.3.2.3. Determination of an appropriate length of the retention function 190
7.3.2.4. Data handling 191
7.3.3. Comparing the matrix algorithm and TLS 191
7.3.3.1. Simulation results and discussion 191
7.3.3.2. Clinical results and discussion 196
7.3.4. Evaluation of the matrix algorithm and TLS 199
7.4. Concluding remarks 202

CHAPTER 8 :
USE OF TLS IN PARAMETER ESTIMATION OF TRANSFER FUNCTION MODELS 203

8.1. Introduction 203
8.2. (Extended) instrumental variable methods 205
8.2.1. Noise-free input case 205
8.2.2. Noisy input case 208
8.3. Relationships between TLS and IV methods 209
8.4. Simulation results and discussion 213
8.5. Concluding remarks 222

CHAPTER 9 :
DIRECT SPEED IMPROVEMENT OF THE TLS COMPUTATIONS 225

9.1. Introduction 225
9.2. The classical SVD algorithm 227
9.3. The partial SVD algorithm PSVD 230
9.3.1. Delay initialization phase 230
9.3.2. Partial diagonalization 230
9.3.3. QL iterations 235
9.3.4. Outline of the PSVD algorithm 236
9.3.5. The case m < n 239
9.3.6. Computational considerations 239
9.3.7. Solution of a set of nonhomogeneous equations 241
9.4. Comparison of the classical SVD versus the partial SVD algorithm 243
9.4.1. Storage requirements 243
9.4.2. Operation counts 243
9.4.3. Computational results 250
9.5. The partial TLS algorithm PTLS 253
9.5.1. Outline of the PTLS algorithm 253
9.5.2. Comparison of the classical TLS versus the partial TLS algorithm 255
9.6. Concluding remarks 257

CHAPTER 10 :
ITERATIVE SPEED IMPROVEMENTS OF THE TLS COMPUTATIONS 259

10.1. Introduction 259
10.2. Matrix functions 261
10.3. Chebyshev iteration 264
10.3.1. Chebyshev polynomials 264
10.3.2. The Chebyshev iteration algorithm 266
10.3.3. Convergence properties and applicability 268
10.4. Inverse iteration 271
10.4.1. The inverse iteration algorithm 271
10.4.2. Convergence properties and applicability 275
10.5. Rayleigh quotient iteration 278
10.5.1. The Rayleigh quotient iteration algorithm 278
10.5.2. Convergence properties and applicability 281

10.6. Lanczos methods 282
10.6.1. Lanczos algorithms 282
10.6.2. Convergence properties and applicability 287
10.7. Overall comparison in efficiency of direct versus iterative methods w.r.t. the TLS problem 294
10.8. Efficiency of inverse iteration and of the partial and classical TLS algorithm w.r.t. the estimation of frequency response functions 302
10.8.1. Problem description 302
10.8.2. Method description 304
10.8.3. Computational results 308
10.9. Concluding remarks 315

CHAPTER 11 :
CONCLUSIONS 317

11.1. Review of the obtained results 317
11.2. Suggestions for future research 323

REFERENCES 327

APPENDICES

A. The singular value decomposition 339
B. Statistical properties of an estimator 343
C. Summary (in Dutch) 349
C.1. Inleiding 349
C.2. Het totale kleinste kwadratenprobleem : formulering, uitbreidingen en algoritme 359
C.3. Eigenschappen van het TKK probleem en hun verband met deze van het KK probleem 361
C.4. De gevoeligheid van het TKK probleem en zijn verband met het KK probleem 362
C.5. De statistische eigenschappen van het TKK probleem 364
C.6. Verbanden tussen TKK schatting en klassieke lineaire regressie 365
C.7. Selectie van een deelmodel door middel van de TKK benadering in problemen met multicollineariteiten en fouten in alle veranderlijken 366

C.8. Het gebruik van de TKK benadering in de schatting van impulsresponsies door middel van discrete deconvolutie 368
C.9. Het gebruik van de TKK benadering in de schatting van de parameters in overdrachtsfunctiemodellen 370
C.10. Rechtstreekse verbetering in rekensnelheid van de TKK berekeningen 371
C.11. Iteratieve verbetering in rekensnelheid van de TKK berekeningen 373
C.12. Besluiten 375
C.12.1. Overzicht van de voornaamste bijdragen 376
C.12.2. Suggesties voor verder onderzoek 377

* *

*

INTRODUCTION

In this thesis a deep study and analysis is made of the method of **total least squares** (TLS) which is one of several linear parameter estimation techniques that have been devised to compensate for data errors.

The problem of linear parameter estimation arises in a broad class of scientific disciplines such as signal processing, automatic control, system theory and in general engineering, statistics, physics, economics, biology, medicine, etc. It can be described by a linear equation:

$$a_1 x_1 + \ldots + a_n x_n = b \qquad \text{(I.1)}$$

where $a_1, \ldots, a_n$ and b denote the **variables** and $x = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ plays the role of a parameter vector that characterizes the special system. A basic problem of applied mathematics is to **determine an estimate of the true but unknown parameters from certain measurements of the variables.** This gives rise to an **overdetermined** set of m linear equations ($m > n$) :

$$Ax = b \qquad \text{(I.2)}$$

where the i-th row of the data matrix $A \in \mathbb{R}^{m \times n}$ and the vector of observations $b \in \mathbb{R}^m$ contain the measurements of the variables $a_1, \ldots, a_n$ and b respectively.

In the classical least squares (LS) approach the measurements A of the variables a_i (the left handside of our equations (I.2)) are assumed to be free of error and hence, all errors are confined to the observation vector b (the right handside of our equations (I.2)). However, this assumption is frequently unrealistic : sampling errors, human errors, modelling errors, and instrument errors may imply inaccuracies of the data matrix A. **TLS is one method of fitting** that is appropriate **when there are errors in both the observation vector b and the data matrix A.** It amounts to fitting a "best" subspace to the measurement points (A_i^T, b_i), $i = 1, \ldots, m$ where A_i^T is the i-th row of A.

The effect of the use of TLS in comparison with LS becomes clear when we consider the **simplest example of parameter estimation**, i.e. when only one parameter ($n = 1$) has to be estimated. Consider the following relation :

$$ax = b \qquad \text{(I.3)}$$

between two variables a and b. An estimate $\tilde{x}$ has to be found for parameter x. Assume that m measurements of the variables a and b are available :

$$\begin{aligned} \tilde{a}_i &= a_i + \Delta a_i \\ \tilde{b}_i &= b_i + \Delta b_i \qquad i = 1, \ldots, m \end{aligned} \qquad \text{(I.4)}$$

Δa_i and Δb_i represent the random errors on the values a_i and b_i of the crue variables a and b.
If a can be observed exactly, i.e. $\Delta a_i = 0$, all errors may be confined to the right handside b. Hence, the use of LS for solving (I.2) is appropriate. This method perturbs the m observations $\tilde{b}_i$ by a minimum amount r_i so that $\tilde{b}_i + r_i = \tilde{a}_i \tilde{x}$ can be predicted by the relation (I.3).

This is done by minimizing the sum of squared differences $\sum_{i=1}^{m} (\tilde{a}_i \tilde{x} - \tilde{b}_i)^2$.

The best estimate of x follows then immediately :

$$\tilde{x} = \sum_{i=1}^{m} \tilde{a}_i \tilde{b}_i \; / \; \sum_{i=1}^{m} \tilde{a}_i^2 \qquad \text{(I.5)}$$

This LS estimation has a nice geometric interpretation in fig. I.1.a.
If b can be measured without error, i.e. $\Delta b_i = 0$, the use of LS is again appropriate. Indeed we can write (I.3) as

$$b \frac{1}{x} = a \qquad \text{(I.6)}$$

and confine all errors to the right handside a. By minimizing the sum of squared differences between the measured values $\tilde{a}_i$ and the values $\tilde{b}_i / \tilde{x}$ predicted by (I.6), the best estimate of x follows immediately (see fig. I.1.b) :

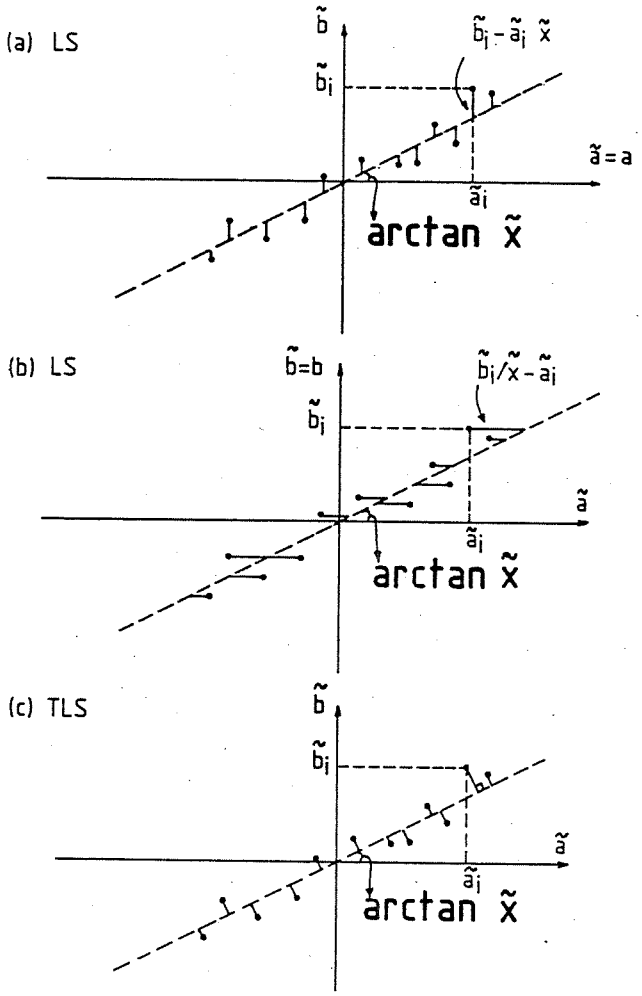


FIGURE I.1 : Geometric interpretation of one parameter estimation ax = b with errors on (a)b only (LS solution) (b)a only (LS solution) (c) both a and b (TLS solution).

$$\tilde{x} = \sum_{i=1}^{m} \tilde{b}_i^2 \; / \; \sum_{i=1}^{m} \tilde{a}_i \tilde{b}_i \qquad \text{(I.7)}$$

In many problems of physics, enigneering, biology, ... however, both variables are measured with errors, i.e. $\Delta a_i \neq 0$ and $\Delta b_i \neq 0$. If the errors on the variables are independently and identically distributed with zero mean and common variance σ^2, the best estimate is obtained by minimizing the sum of squared distances of the observed points from the fitted line. This is in fact the solution $\tilde{x}$ one obtains by solving (I.2) with TLS for n = 1. Fig. I.1.c illustrates the estimation. The deviations are orthogonal to the fitted line : it is the sum of squares of their lengths that is minimized. Therefore this estimation procedure is sometimes known as **orthogonal regression.**

This problem of parameter estimation where all variables involved contain errors has already a **long history.** Indeed the n = 1 case has been scrutinized since the turn of the century. Some well-known contributors are Adcock [3], Pearson [83], Koopmans [66], Madansky [73] and York [136] (see [7] for a list of references). The method of orthogonal regression has been discovered and rediscovered many times, often independently. Although this estimation technique for solving the univariate problem (n = 1), i.e. line fitting, is quite old, it has only been recently extended to the **multivariate problem.** In the field of numerical analysis, this problem was first studied by Golub and Van Loan [47]. Their analysis as well as their solution technique, called total least squares (TLS), was heavily based on the singular value decomposition (SVD). Independently, Gleser [41] studied the problem in the field of statistics and obtained the same estimate as the TLS technique. His method was based on an eigenvalue-eigenvector analysis and called multivariate "errors-in-variables" regression. Both authors also extended their estimation technique to problems with more than one observation vector in the right handside of equation (I.2).

Remember that **TLS is only one possible fitting technique** for estimating the parameters of a linear multivariate problem. It gives the "best" estimates (in a statistical sense) when all variables are subject to independently and identically distributed errors with zero mean and common covariance matrix of the form $\sigma^2 I$ (I is the identity matrix). Several

other and more general approaches to this problem have led to as many other fitting techniques for the linear as well as for the nonlinear case. Mostly, knowledge of the error covariance matrix is required (see [124-97-88] for a list of references). Recently, new insights in the identification of linear relations from noisy data led to the non-uniqueness principle. Based on minimal assumptions, a general solution for all linear relations present in the problem was proposed [60-25].

In this thesis, the **main emphasis** is put on the **linear algebraic approach** of the TLS technique. Brief attention is given to the viewpoint of the statistician.
A **first contribution** of this work is the **extension of the TLS technique** to all cases in which the TLS algorithm of Golub and Van Loan fails to compute a solution. Our analysis is essentially based on the properties of the powerful SVD and leads to the development of a more general algorithmic implementation of the TLS technique which includes the extensions (chapter 1).

A second important item in the analysis of the TLS technique is **the validity**, i.e. the domain of applicability, of its estimate. Hereto, all its properties must be fully analyzed for all possible conditions that (may) occur. A comparison with other existing estimation techniques can highly elucidate its applicability and significance w.r.t. reliability, sensitivity, ... Our **answer to this question of validity** forms our **second contribution** of this thesis. Not only a thorough theoretical analysis of the algebraic properties of the TLS technique in relation to the classical LS method is given (chapter 2), but also its statistical properties are presented (chapter 4) and evaluated w.r.t. the classical regression estimation techniques (chapter 5). Moreover, the sensitivity of the TLS technique in relation to the LS method is discussed for various perturbation effects on the data. This is done by performing a theoretical sensitivity analysis and confirming our results with simulation tests (chapter 3).

A third important aspect that needs to be analyzed is **the practical applicability of TLS for solving various problems** in different domains such as signal processing, system theory, medicine, ... This requires enough insight into the particular needs and specific problems of each

domain. Based on this information, the best strategy for solving the problem with TLS can be set up. In this thesis, such procedures are developed for solving different practical problems, e.g. in subset selection (chapter 6) and in time domain system identification for estimating the impulse response by discrete deconvolution (chapter 7) and for estimating the parameters in a transfer function model (chapter 8). They are compared with the current solution techniques in the particular domain and their performance is experimentally evaluated by simulation and with real life data. This forms **the third main contribution** of the thesis.

And last but not least, the **efficiency of the TLS computations** plays an important role in a lot of applications e.g. in real-time estimation. A careful analysis of the TLS solution and the computation of the SVD allows to improve the computational speed of the classical TLS algorithm in a **direct way**. This results in the development of a new, computationally improved and equally reliable algorithm (chapter 9). Moreover, if a priori information is available (e.g. the TLS solution at a previous time step when slowly varying sets of equations must be solved at each time instant), the computational speed can be improved by computing the TLS solution in an **iterative way** (chapter 10). A thorough analysis of the convergence properties and computational cost of the existing iterative methods allows to evaluate the efficiency and applicability of these methods in the TLS computation and makes a comparison with the direct computation methods possible. This enables the user to choose the most efficient method according to his own needs and problem characteristics, and forms the **fourth contribution** of this thesis.

OUTLINE OF THE THESIS

In the **first 5 chapters** of this thesis, **different aspects of the TLS technique** are treated.
Chapter 1 shows which **kind of problems can be solved with the TLS technique** and outlines **the computational aspects** of the TLS solution in each case. First, the main principle and solution of the one-dimensional, generic TLS problem with unique solution is presented and compared with the classical LS method from a geometric point of view. This means that only one observation vector is considered (one-dimensional), that the TLS solution can be computed with the algorithm of Golub and Van Loan (generic) and is unique. Next, attention is given to the possible extensions of this problem. For each case, it is shown how the TLS solution can be obtained. We discuss consecutively problems having more than one observation vector (multidimensional TLS problems), problems for which the TLS solution is no longer unique, problems which the algorithm of Golub and Van Loan can not handle (nongeneric TLS problems) and finally, problems in which some of the variables or columns in the data matrix are known exactly. At the end, the TLS computation is summarized in one algorithm which takes into account all the extensions given above.

Chapter 2 proves interesting **algebraic relationships** between the TLS technique and the classicial LS method w.r.t. their solution, residuals, approximation efforts and associated subspaces.

In **chapter 3**, the **sensitivity of the TLS technique** is compared w.r.t. the classical LS method in the presence of errors on the data. The perturbation effects on their solution and on the SVD of their associated matrices are analyzed and confirmed by simulations.

Chapter 4 presents the **statistical properties** of the TLS technique based on knowledge of the distribution of the errors on the data. It is shown that the TLS technique computes the best estimate (in a statistical sense) if all errors on the data are rowwise independently and identically distributed with zero mean and common covariance matrix of the form $\sigma^2 I$.

In **chapter 5**, some **algebraic equivalences**, interrelations and differences in approach **between the classical linear regression estimators** (least squares, principal component, ridge regression and latent root regression) **and the TLS technique** in the presence of multicollinearities are outlined, using the SVD and geometric concepts.
In the **next 3 chapters**, the **practical applicability of TLS** in different domains is explored.

Chapter 6 discusses the problem of **subset selection in multicollinearity problems** whose variables can not be observed without error. Using the TLS approach, a new subset selection algorithm is derived, as well as some variants. They are evaluated in a simulation set up w.r.t. their selection abilities and accuracy in parameter estimation and prediction.

In **chapter 7, the applicability of TLS in impulse response estimation by discrete deconvolution** is treated. Its advantages w.r.t. the LS method are illustrated in a simulation. In particular, it is shown how the technique can be applied in the computation of the renal retention function by deconvolving the renogram in an efficient and reliable way.

Chapter 8 considers another problem of **system identification**, namely **the estimation of parameters in a transfer function model.** By outlining the relationships between the TLS technique and the current instrumental variable methods and by evaluating their performance experimentally with simulation data, the applicability of the TLS technique in this problem is demonstrated.

The **last two chapters** show how to improve **the efficiency of the TLS computations.**
In **chapter 9**, the TLS computations **are speeded up in a direct way by modifying the SVD computation** appropriately. These modifications are summarized in a new, computationally improved algorithm. An analysis of its operation counts, as well as computational results, show its relative efficiency w.r.t. the classical computations.

Chapter 10 describes how the TLS computations **can be speeded up in an iterative way** if a priori information about the TLS solution is available. **Four iterative methods**, namely Chebyshev iteration, inverse iter-

ation, Rayleigh quotient iteration and the Lanczos method are discussed. Based on their convergence properties and number of operations in each iteration step, their efficiency in TLS computations is evaluated and compared with the direct computation methods. At the end, the most promising iterative technique, namely inverse iteration, and the direct computation methods are compared w.r.t. their efficiency in a practical real life problem concerning the estimation of frequency response functions.

Finally, in **chapter 11** the summarizing **conclusions of this research** are presented and **suggestions** are made **for further work.**

NOTATIONS AND PRELIMINARIES

Before starting, we introduce our **notations** and list the **assumptions** made about the problems in this thesis.

- A **matrix** is always denoted by a **capital** letter, e.g. A. The corresponding lowercase letter with the subscript i and ij refers to the i-th column and (i,j)-th entry respectively, e.g. a_i, a_{ij}.

- A **vector** is represented by a **lowercase** letter, e.g. b. The individual components are denoted with single subscripts, e.g. b_i.

- The superscript T denotes the **transpose** of a vector or matrix.

- A special notation is convenient for **diagonal matrices**. If A is an m by n matrix and we write

  $$A = \text{diag}(\alpha_1, \ldots, \alpha_p) \qquad p = \min(m,n)$$

  then $a_{ij} = 0$ whenever $i \neq j$ and $a_{ii} = \alpha_i$ for $i = 1, \ldots, p$.

- The m bij m **identity** matrix is denoted by I_m.

- The **set of** m **linear equations** in n by d unknowns X is represented in matrix form by :

  $$AX = B \tag{0.1}$$

  A is the m by n data matrix and B is the m by d matrix of observations.

- X' is the n by d **minimum norm** least squares (LS) **solution** and X is the n by d minimum norm total least squares (TLS) solution of (0.1).

- For the **one-dimensional** problem, i.e. d = 1, the matrices are replaced by their corresponding vector notations, e.g. the vectors b and x are used instead of the matrices B and X in (0.1).

- Denote the **singular value decomposition** (SVD) of A in (0.1) by

$A = U' \Sigma' V'^T$

with $U' = [U'_1; U'_2]$, $U'_1 = [u'_1, \ldots, u'_n]$, $U'_2 = [u'_{n+1}, \ldots, u'_m]$,

$u'_i \in \mathbb{R}^m$, $U'^T U' = I_m$

$V' = [v'_1, \ldots, v'_n]$, $v'_i \in \mathbb{R}^n$, $V'^T V' = I_n$

$\Sigma' = \text{diag}\ (\sigma'_1, \ldots, \sigma'_n)$ and $\sigma'_1 \geq \ldots \geq \sigma'_n \geq 0$ (0.2)

and denote the SVD of $[A;B]$ in (0.1) by $[A;B] = U \Sigma V^T$

with $U = [U_1;U_2]$, $U_1 = [u_1, \ldots, u_n]$, $U_2 = [u_{n+1}, \ldots, u_m]$, $u_i \in \mathbb{R}^m$,

$U^T U = I_m$

$$V = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{matrix} n \\ d \end{matrix} = [v_1, \ldots, v_n, v_{n+1}, \ldots, v_{n+d}],\ v_i \in \mathbb{R}^{n+d},$$

$\quad n \quad d$

$V^T V = I_{n+d}$

$\Sigma = \text{diag}\ (\sigma_1, \ldots, \sigma_{n+d})$ and $\sigma_1 \geq \ldots \geq \sigma_{n+d} \geq 0$

$\Sigma_1 = \text{diag}\ (\sigma_1, \ldots, \sigma_n) \in \mathbb{R}^{n \times n}$ and

$\Sigma_2 = \text{diag}\ (\sigma_{n+1}, \ldots, \sigma_{n+d})$ (0.3)

The columns of U' and U are the left singular vectors, the columns of V' and V are the right singular vectors and the diagonal elements of Σ' and Σ are the singular values of matrix A and $[A;B]$ respectively.

- $v_{j,i}$ is the j-th component of the i-th singular vector v_i. In particular, $v_{n+1,n+1}$ is the (n+1)-th component of the (n+1)-th singular vector v_{n+1}.

- Let $V(\sigma_j)$ (resp., $U(\sigma_j)$) be the right (resp., left) singular subspace of $[A;B]$ associated with the singular value σ_j, and let $V'(\sigma_j)$ (resp., $U'(\sigma_j)$) be the right (resp., left) singular subspace of A associated with the singular value σ_j.

- Denote by R(S) the **range** of matrix S, i.e. the space generated by its columns.

- $[A;B']$ is the LS **approximation** of $[A;B]$ with B' the orthogonal projection of B onto the range R(A) of A. The TLS approximation of $[A;B]$ (defined in section 1.2 and 1.4) is denoted by $[\hat{A};\hat{B}]$.

- The **correction** matrix $\Delta B' = B - B'$ is then the LS approximation effort and the correction matrix $[\Delta\hat{A};\Delta\hat{B}] = [A-\hat{A};B-\hat{B}]$ is the TLS approximation effort in order to obtain a solution of (0.1)

- One easily observes that the LS correction matrix $\Delta B'$ equals the LS **residue** R', defined by $B-AX'$. The TLS residue $\hat{R}$ however, is defined by $B-A\hat{X}$ and differs from the TLS correction matrix $[\Delta\hat{A};\Delta\hat{B}]$.

- Denote by $[A_o;B_o]$ the exact but unobservable data, perturbed by errors $[\Delta A;\Delta B]$ such that $[A;B] = [A_o + \Delta A; B_o + \Delta B]$.

- The **Probenius** norm of an m by n matrix M is defined by ("tr" means trace) :

$$\|M\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{m} m_{ij}^2} = \sqrt{\text{tr}\ (M^T M)} \qquad (0.4)$$

- The **2-norm** or **Euclidean norm** of an n-dimensional **vector** y is defined by :

$$\|y\|_2 = \sqrt{\sum_{i=1}^{n} y_i^2} \qquad (0.5)$$

- Based on (0.5), the **2-norm** of an m by n **matrix M** is defined by :

$$\|M\|_2 = \sup_{y \neq 0} (\|My\|_2 \ / \ \|y\|_2) \qquad (0.6)$$

and equals the largest singular value of M.

- Throughout the thesis, we **assume** that the set of equations $AX = B$ is **overdetermined**, i.e. $m > n$, and that all **preprocessing** measures on the data (such as scaling, whitening, centering, standardizing, ...) have been performed in advance. This avoids the need to burden the equations with left and right multiplications by diagonal matrices as in [47].

- Furthermore we usually deal with **one-dimensional generic** TLS problems having a **unique** solution.

- We call the problem **one-dimensional** if we only consider one observation vector b, i.e. d = 1. If d > 1, the problem is called multidimensional. (0.7)

- We call the problem **generic** if $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$ with $p \geq 0$ and the submatrix

$$\begin{bmatrix} v_{n+1,n-p+1}, \ldots, v_{n+1,n+d} \\ \vdots \qquad\qquad \vdots \\ v_{n+d,n-p+1}, \ldots, v_{n+d,n+d} \end{bmatrix}$$

of V in (0.3) has full rank d. If $\sigma_n > \sigma_{n+1}$, this means that V_{22} is nonsingular (or $v_{n+1,n+1} \neq 0$ if d = 1). The TLS solution of generic problems can be computed with the algorithm of Golub and Van Loan [47]. (0.8)

- We call the solution **unique** if the problem (0.1) has only one solution satisfying the TLS criteria (defined in section 1.2 and 1.4). This is the case for generic problems where $\sigma_n > \sigma_{n+1}$. (0.9)

The assumptions (0.7), (0.8) and (0.9) are mostly satisfied and will be alleviated in the sections 1.4 to 1.8 where the generic one-dimensional problem with unique solution is extended.

CHAPTER 1

THE TOTAL LEAST SQUARES PROBLEM : FORMULATION, EXTENSIONS AND ALGORITHM

1.1. INTRODUCTION

As described in the general introduction, many problems in signal processing, system theory, automatic control and in general engineering, physics and economics give rise to an **overdetermined** set of linear equations Ax = b which are usually solved with the linear **least squares** technique (LS). This technique assumes that al the **errors are confined to the observation vector** b. Unfortunately, this assumption is frequently **unrealistic**; sampling errors, human errors, modelling errors and instrument errors may imply inaccuracies on the data matrix A. For those cases a better **more general** fitting technique, **total least squares** (TLS), has been devised **to compensate for data errors**. Although studies of the univariate problem, i.e. line fitting, are quite old [3], the multivariate problem has only been analyzed recently. Golub and Van Loan [47] introduced the method in the field of numerical analysis by presenting a first **singular value decomposition** (SVD) analysis of the problem.

In this chapter, the **TLS problem** is **fully analyzed** : we first review the generic TLS problem as stated by Golub and Van Loan, and then generalize the results to nongeneric problems. An analytical and geometric comparison with the ordinary LS technique enlightens the conceptual and practical significance of the TLS problem. In section 1.2 the **main principle** of the TLS problem is presented. Its definition is formulated and illustrated with an extreme example and its **solution** is given for the **generic one-dimensional** TLS problem with **unique** solution. This means that only one observation vector is given (one-dimensional), that the TLS solution is unique and can be computed with the algorithm of Golub and Van Loan (generic). In section 1.3, the differences between the TLS problem with respect to the ordinary LS problem are clarified by a **geometric** representation of the problem in its column and row space.

In the following sections, the generic one-dimensional TLS problem with unique solution, described in section 1.2, is **extended** to all cases. First in section 1.4, the TLS solution is given for problems in which **more than one observation vector** is associated with the same data matrix A (multidimensional TLS problem). Second, problems for which the TLS **solution is no longer unique**, are treated in section 1.5. This happens if the smallest singular values of the associated matrix [A;B] coincide. In order to preserve uniqueness, TLS seeks for the minimum norm solution. Third, in section 1.6, we **generalize** the TLS solution **to nongeneric problems**, i.e. problems for which the **algorithm of Golub and Van Loan** [47] fails to produce a solution. We describe the properties of those nongeneric problems and prove that the proposed **generalization is always optimal** with respect to the TLS criteria in the one-dimensional case, as well as in the multidimensional case. Fourth, in section 1.7, the TLS problem is extended to cases in which **some of the columns** in the data matrix have **no error**.
Finally in section 1.8, the TLS computation is **summarized** in one **algorithm** which takes into account all the extensions described above. First, the generalized TLS algorithm is given for erroneous data. A fully documented program of this algorithm is provided in [121]. Second, the extension of the algorithm in case of some error-free columns in the data matrix A, is presented. In this way, one algorithm can handle the ordinary LS problem (A is assumed to be entirely error-free) as well as a variety of TLS problems. We call this algorithm the **classical TLS algorithm** because it is based on the algorithm of Golub and Van Loan. A computationally **more efficient** algorithm, called **partial TLS** (PTLS), is developed in chapter 9. Moreover, if a priori information about the solution of the problem to be solved is available, the TLS computations can be speeded up in an **iterative** way (see chapter 10).

A nice **overview of all TLS problems** discussed in this chapter, is given in the following diagram. For each case, it describes the main properties of the TLS solution.

Definitions :

- SVD of $[A;B] = \sum_{i=1}^{n+d} \sigma_i \cdot u_i \cdot v_i^T$ with $v_i \in \mathbb{R}^{n+d}$, $u_i \in \mathbb{R}^m$ orthonormal vectors and $\sigma_1 \geq \cdots \geq \sigma_{n+d} \geq 0$

- $V_Y = \begin{bmatrix} v_{n+1,n-p+1} & \cdots & v_{n+1,n+d} \\ \vdots & & \vdots \\ v_{n+d,n-p+1} & \cdots & v_{n+d,n+d} \end{bmatrix}$ with $p \geq 0$ and $v_{j,i}$ the j-th component of v_i.

- $V_{22} = V_Y$ with $p = 0$

- a TLS solution of $AX = B$ is any solution X of the set

$$(A-\Delta\hat{A})X = B-\Delta\hat{B}$$

that satisfies the TLS criteria :

$$R(B-\Delta\hat{B}) \subset R(A-\Delta\hat{A})$$

$$\|[\Delta\hat{A};\Delta\hat{B}]\|_F \text{ minimal}$$

The problem of finding such a $[\Delta\hat{A};\Delta\hat{B}]$ is referred to as the TLS problem.

1.2. FORMULATION AND SOLUTION OF THE GENERIC ONE-DIMENSIONAL TLS PROBLEM WITH UNIQUE SOLUTION

In this section we formulate the **main principle** of the TLS problem and give its **solution** for the **generic** case. A good way to introduce and motivate the method is to recast the **ordinary least squares** (LS) problem [48, sec. 6.1] as follows :

LS definition :

Given an overdetermined set of m linear equations $Ax = b$ in n unknowns x, a **least squares (LS) solution** is any solution x of the set :

$$Ax = b' \qquad (1.1)$$

where b' is determined such that

$$b' \in R(A) \tag{1.2}$$

$$\|r'\|_2 = \|b-b'\|_2 \text{ is minimal} \tag{1.3}$$

The **problem of finding b'** such that (1.2 - 1.3) are satisfied, is referred to as the **LS problem.**

Whenever the LS solution is not unique, LS singles out the minimum norm LS solution, denoted by x'. Thus, the LS problem amounts to **perturbing the observation vector** b by a minimum amount r' so that b' = b-r' can be predicted by the columns of A. (1.2) and (1.3) are satisfied if b' is the orthogonal projection of b onto R(A). Using (0.2), the minimum norm LS solution x' is given by $V' \Sigma'^{-1} U'^T b$.
Knowing that the idea behind TLS is to consider **pertubations of both b and A**, it is now easy to formulate the TLS problem :

TLS definition :

Given an overdetermined set of m linear equations Ax = b in n unknowns x, a **total least squares (TLS) solution** is any solution x of the set

$$\hat{A}x = \hat{b} \tag{1.4}$$

where $\hat{A}$ and $\hat{b}$ are determined such that

$$\hat{b} \in R(\hat{A}) \tag{1.5}$$

$$\|[\Delta\hat{A};\Delta\hat{b}]\|_F = \|[A;b] - [\hat{A};\hat{b}]\|_F \text{ is minimal} \tag{1.6}$$

The **problem of finding** $[\Delta\hat{A};\Delta\hat{b}]$ such that (1.5 - 1.6) are satisfied, is referred to as the **TLS problem.**

Whenever the TLS solution is not unique, the TLS technique singles out one TLS solution with minimal norm (see section 1.5). This **unique minimum norm** TLS solution is denoted by $\hat{x}$.

It is easily proven that the TLS **approximation effort** (1.6) is always **less** than the LS approximation effort (1.3) [130]. This is nicely illustrated in the following extreme example.

Example 1-1

For any set of equations Ax = b of the form (1.7) the LS and TLS solution can be computed analytically [105] :

$$\begin{bmatrix} N-1 & -1 & \dots & -1 \\ -1 & N-1 & \dots & -1 \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ -1 & -1 & \dots & N-1 \\ -1 & -1 & \dots & -1 \\ -1 & -1 & \dots & -1 \end{bmatrix}_{N,N-2} \begin{bmatrix} x_1 \\ \cdot \\ \cdot \\ \cdot \\ x_{N-3} \\ x_{N-2} \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ \cdot \\ \cdot \\ \cdot \\ -1 \\ N-1 \\ -1 \end{bmatrix} \tag{1.7}$$

The deviations are

$$\Delta b' = b-b' = 0.5\ [0, \dots, 0, N, -N]^T \tag{1.8}$$

$$[\Delta\hat{A};\Delta\hat{b}] = [A;b] - [\hat{A};\hat{b}] = [-1, -1, \dots, -1, N-1]^T\ [1, 1, \dots, 1]/(N-1) \tag{1.9}$$

Hence the ratio of the length of the deviations

$$\|\Delta b'\|_2^2 \ / \ \|[\Delta\hat{A};\Delta\hat{b}]\|_F^2 = N/2 \tag{1.10}$$

becomes large for large dimension N.

$$\text{As } \lim_{N\to\infty} \|\Delta b'\|_2 \ / \ \|b\|_2 = \lim_{N\to\infty} \sqrt{N/(2(N-1))} = \sqrt{2} \tag{1.11}$$

$$\text{and } \lim_{N\to\infty} \|\Delta\hat{b}\|_2 \ / \ \|b\|_2 = \lim_{N\to\infty} \|\Delta\hat{A}\|_F \ / \ \|A\|_F = \lim_{N\to\infty} 1/(N-1) = 0 \tag{1.12}$$

one observes that, if only the vector b is allowed to vary (the LS approach), b has to be modified over approximately 70 % for large N. If the matrix A is allowed to vary also (TLS approach), a small change of 1/(N-1) is sufficient. The reader can appreciate the difference between LS and TLS for this example with N = 4 from the geometric representation in fig. 1.1.

To **solve the TLS problem in the generic case** a very simple deviation is developed by Golub and Van Loan [47]. Hereto we bring Ax = b into the form :

$$[A;b]\ [x^T;-1]^T = 0 \tag{1.13}$$

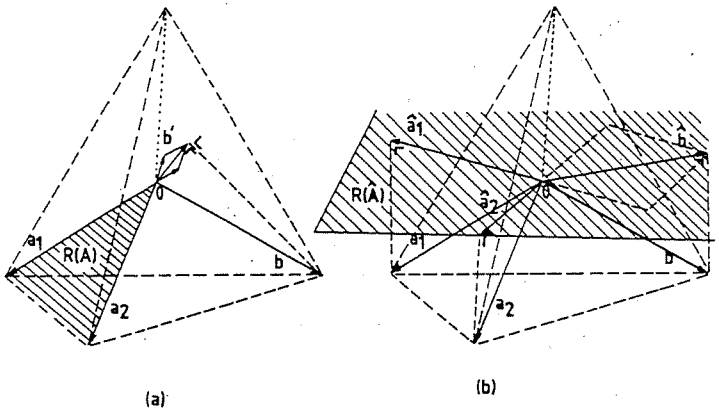


Fig. 1.1 : Geometric illustration of example 1-1 for N = 4.

(a) The LS solution is obtained by solving Ax = b' with b' the orthogonal projection of b onto R(A), generated by the columns a_1 and a_2.

(b) The TLS solution is obtained by approximating the columns a_1 and a_2 of A by $\hat{a}_1$ and $\hat{a}_2$ and b by $\hat{b}$ until $\hat{b}$ is in the space $R(\hat{A})$ generated by the columns $\hat{a}_1$ and $\hat{a}_2$ and solving $\hat{A}x = \hat{b}$.

If $\sigma_{n+1} \neq 0$, [A;b] is of rank n+1 and the space S generated by the rows of [A;b] coincides with $\mathbb{R}^{n+1}$. There is no nonzero vector in the orthogonal complement of S, hence the set of equations (1.13) is incompatible. In order to obtain a solution, the rank of [A;b] must be reduced to n. Based on the Eckart-Young theorem of the SVD (see theorem A.2), the best n-th rank approximation $[\hat{A};\hat{b}]$ of [A;b] which **minimizes the deviations** in variance i.e. (1.6) is satisfied, is obtained by making σ_{n+1} zero. The **lower rank approximation** for (1.13) is then

$$[\hat{A};\hat{b}]\,[\hat{x}^T;-1]^T = 0 \qquad (1.14)$$

with $[\hat{A};\hat{b}] = U\,\hat{\Sigma}\,V^T$ and $\hat{\Sigma} = \mathrm{diag}\,(\sigma_1, \ldots, \sigma_n, 0)$.

Hence, the **minimal correction matrix** is given by the rank-one matrix

$$[\Delta\hat{A};\Delta\hat{b}] = \sigma_{n+1}\,u_{n+1}\,v_{n+1}^T \qquad (1.15)$$

and its Frobenius norm, namely σ_{n+1}, represents the **minimal TLS approximation effort.** It is clear that the solution of (1.14) is given by the only vector v_{n+1}, **normal** to the n remaining right singular vectors of [A;b], i.e. the last column of V. The TLS solution is then obtained by **scaling** v_{n+1} until its last component is -1 or

$$[\hat{x}^T;-1]^T = -v_{n+1} \,/\, v_{n+1,n+1} \qquad (1.16)$$

The conditions (0.8) en (0.9) guarantee that $v_{n+1,n+1} \neq 0$ i.e. (1.5) is satisfied and hence, the TLS solution can be computed with the algorithm of Golub and Van Loan [47]. Observe that, if σ_{n+1} is zero, [A;b] is of rank n in the generic case with v_{n+1} orthogonal to its row space. Hence, the set of equations (1.13) is compatible and no approximation is needed to obtain the exact solution, given by (1.16).

1.3. A GEOMETRIC COMPARISON OF THE TLS PROBLEM WITH RESPECT TO THE LS PROBLEM

1.3.1. Viewed from the column space

The main principle of the total least squares technique (TLS) can be best understood by comparing it with ordinary least squares from a geometric point of view.

The classical least squares (LS) problem of Ax = b aims at finding a best approximation b' to b such that Ax' = b' is compatible. Geometrically the fact that Ax = b has no solution implies that the n-dimensional subspace R(A) of $\mathbb{R}^m$, generated by the columns of A, does not contain the vector b. The best approximation b', satisfying (1.2) and (1.3), is then the orthogonal projection of b into that subspace (fig. 1.2.a).

In many applications, both A and b are inaccurate. It seems then unfair to ask all the approximation effort from b. In order to overcome this deficiency, the total least squares problem was proposed.

By looking at fig. 1.2.b, clearly TLS bends b **and** the set of columns a_i of A towards each other, until the new set of equations is compatible (see (1.5)), and such that the **approximation effort** is **minimized** (see

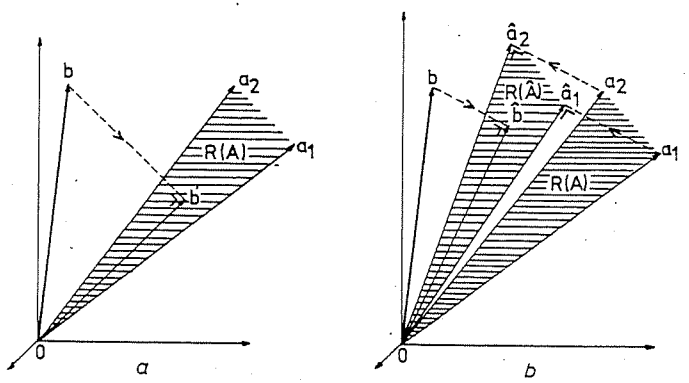


Fig. 1.2 (a) The LS solution is obtained by projecting b orthogonally onto R(A) and solving Ax = b'.

(b) The TLS solution is obtained by approximating the columns a_i of A and b by $\hat{a}_i$ and $\hat{b}$ until $\hat{b}$ is in the space R(Â) generated by the columns $\hat{a}_i$ and solving $\hat{A}x = \hat{b}$.

(1.6)). This effort is given by the square root of the sum of squared lengths of all corrections $\|\Delta\hat{a}_i\|_2$ and $\|\Delta\hat{b}\|_2$ applied to the columns of A and b, and equals σ_{n+1}. Observe that **all correction vectors** $\Delta\hat{a}_i$ and $\Delta\hat{b}$ are **parallel** with each other. This is evident from the expression of $[\Delta\hat{A};\Delta\hat{b}]$ given in (1.15).

1.3.2. VIEWED FROM THE ROW SPACE

Here we present an interesting geometric interpretation of the difference between TLS and LS in the row space $\mathbb{R}^{n+1}$. If no errors are present in the data, the set Ax = b is compatible. Hence the rank of [A;b] equals n and the space $R_R[A;b]$, generated by the rows $r_1, \ldots, r_m$ of [A;b], is n-dimensional. Any n linearly independent rows determine the hyperplane $R_R[A;b]$ whose equation provides the coefficients of the exact solution x. If **errors** occur, then the observed row vectors $r_1, \ldots, r_m$ of [A;b] are **scattered** around the hyperplane. The space spanned by $r_1, \ldots, r_m$ coincides with the ambient space $\mathbb{R}^{n+1}$. As [A;b] is of rank n + 1 now, the set Ax = b is no longer compatible and a method of fitting a lower-dimensional hyperplane to these points is required in order to estimate the solution x.

The LS approach (for the case n = 2, see fig. 1.3.a) then looks for a best approximation b' to b, satisfying (1.2) and (1.3), such that the space $R_R[A;b']$ generated by the rows of [A;b'] is a lower dimensional subspace (two-dimensional in the case of fig. 1.3.a). **Only the last components** of $r_1, \ldots, r_m$ **can vary**. The solution $[x'^T;-1]^T$ is then found at the intersection of the orthogonal complement of $R_R[A;b']$ with the hyperplane $x_{n+1} = -1$. The coefficients of the solution determine the equation of the hyperplane $R_R[A;b']$. Since the observed row vectors have added random disturbances in all coordinate directions, the ordinary LS method, which assumes random errors along **one** coordinate axis only, is not appropriate.

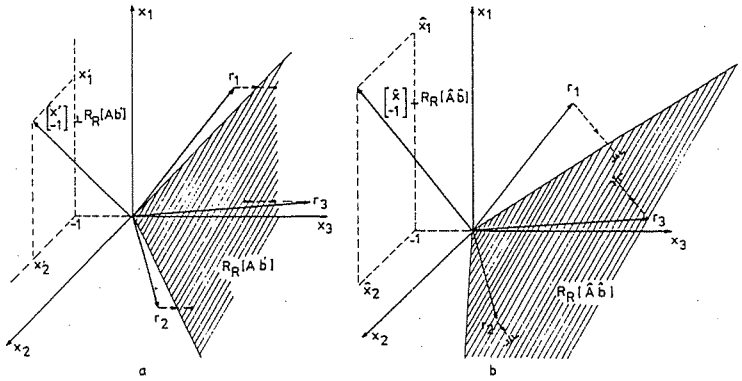


Fig. 1.3 : The difference between (a) the LS solution x' and (b) the TLS solution $\hat{x}$ in the n + 1 = 3-dimensional row space $\mathbb{R}^{n+1}$. r_1, r_2 and r_3 are the rows of [A;b]. $R_R[A;b']$ is the two-dimensional hyperplane best fitting according to LS. $R_R[\hat{A};\hat{b}]$ is the two-dimensional hyperplane best fitting according to TLS.

The TLS approach (fig. 1.3.b) looks for a lower dimensional row space $R_R[\hat{A};\hat{B}]$, closest to the row vectors $r_1, \ldots, r_m$ of $[A;b]$, such that (1.5) and (1.6) are satisfied. Again the solution $[\hat{x}^T;-1]^T$ is then found at the intersection of the orthogonal complement of $R_R[\hat{A};\hat{B}]$ with the hyperplane $x_{n+1} = 1$. Clearly TLS can fit **more easily** a subspace than LS since the **variations are not restricted** to be along **one** coordinate axis x_{n+1}. Again the equation of the hyperplane $R_R[\hat{A};\hat{B}]$ is given by the coefficients of the solution. Observe that all **corrections** applied to the rows of $[A;b]$ and given by the rows of $[\Delta\hat{A};\Delta\hat{B}]$ are **parallel** with the solution vector. This is evident from (1.15) and (1.16). The total approximation effort is given by the square root of the sum of squared **perpendicular** distances from each row vector r_i to the hyperplane $R_R[\hat{A};\hat{B}]$. The simplest case n = 1, i.e. the fitting of a straight line, has been described in the general introduction and is illustrated in fig. I.1.

1.4. THE GENERIC MULTIDIMENSIONAL TLS PROBLEM WITH UNIQUE SOLUTION

The TLS solution of a set of equations AX = B where B is an m by d observation matrix follows from a straightforward generalization of the one-dimensional TLS problem, given in section 1.2. If we allow **more than one** observation vector, the TLS definition can be generalized as follows :

TLS definition :

Given an overdetermined set of m linear equations AX = B, $B \in \mathbb{R}^{m\times d}$ in n×d unknowns X, a **TLS solution** is any solution X of the set

$$\hat{A}X = \hat{B} \tag{1.17}$$

where $\hat{A}$ and $\hat{B}$ are determined such that

$$R(\hat{B}) \subset R(\hat{A}) \tag{1.18}$$

$$\left\|[\Delta\hat{A};\Delta\hat{B}]\right\|_F = \left\|[A;B] - [\hat{A},\hat{B}]\right\|_F \text{ is minimal} \tag{1.19}$$

The **problem of finding** $[\Delta\hat{A};\Delta\hat{B}]$ such that (1.18 - 1.19) are satisfied, is referred to as the **TLS problem.**

Whenever the TLS solution is not unique, TLS singles out the unique minimum norm TLS solution denoted by $\hat{X}$ (see section 1.5).

To solve the **generic multidimensional** problem [48, sec. 12.3] we bring AX = B into the form :

$$[A;B]\,[X^T;-I_d]^T = 0 \tag{1.20}$$

In order to obtain a solution, the rank of $[A;B]$ must be reduced to n. Based on the Eckart-Young theorem of the SVD (see theorem A.2) the best n-th rank approximation $[\hat{A};\hat{B}]$ of $[A;B]$ which minimizes (1.19) is obtained by making the smallest d singular values of $[A;B]$ zero. If the SVD of $[A;B]$ is given by (0.3), then the **lower** rank approximation of (1.20) is :

$$[\hat{A};\hat{B}]\,[\hat{X}^T,-I_d]^T = 0 \tag{1.21}$$

with $[\hat{A};\hat{B}] = U_1\ \Sigma_1\ [V_{11}^T;\ V_{21}^T]$ the TLS approximation of $[A;B]$.
Hence, the minimal correction matrix is given by the rank-d matrix

$$[\Delta\hat{A};\Delta\hat{B}] = U_2\ \Sigma_2\ [V_{12}^T;V_{22}^T] \tag{1.22}$$

and its Frobenius norm

$$\left\|[\Delta\hat{A};\Delta\hat{B}]\right\|_F = \left\|\Sigma_2\right\|_F = \sqrt{\sum_{i=1}^{d}\sigma_{n+i}^2} \tag{1.23}$$

represents the **minimal TLS approximation effort.**
It is clear that the solution of (1.21) is given by the d right singular vectors **normal** to the rows of $[\hat{A};\hat{B}]$ or :

$$\begin{bmatrix}\hat{X}\\ -I\end{bmatrix} /\!/ \begin{bmatrix}V_{12}\\ V_{22}\end{bmatrix} \tag{1.24}$$

If V_{22} is **nonsingular**, the TLS solution follows immediately :

$$\hat{X} = -V_{12}V_{22}^{-1} \tag{1.25}$$

The TLS solution can also be computed directly from (1.17). Indeed, with (1.21), (1.17) becomes :

$$U_1 \Sigma_1 V_{11}^T \hat{X} = U_1 \Sigma_1 V_{21}^T \tag{1.26}$$

Hence if V_{11} is nonsingular, the TLS solution is obtained from :

$$\hat{X} = (V_{11}^T)^{-1} V_{21}^T = (V_{21} V_{11}^{-1})^T \tag{1.27}$$

From [48, theorem 2.4-1] and the orthonormality of V, we know that $\|V_{11}^{-1}\|_2 = \|V_{22}^{-1}\|_2$. This implies that the minimal singular values of V_{11} and V_{22} are equal. Hence the singularity of V_{22} implies that of V_{11}. If $n < d$, this procedure would be more economical than the TLS computation given by (1.25). In practice, the inverses in (1.25) and (1.27) are never computed explicitly but the solution $\hat{X}$ is obtained by solving a set of equations (see section 1.8).

The assumption (0.8) guarantees that V_{22} (and V_{11}) is **nonsingular**, i.e. (1.18) is satisfied and hence the TLS solution is generic. This is proven in the following theorem (which is a correction of the proof given in [48, theorem 12.3-1]).

Theorem 1-1 : Existence of the generic TLS solution

Let the SVD of $[A;B]$ be given by (0.3) and the SVD of A by (0.2), then

$$\sigma_n' > \sigma_{n+1} \Longrightarrow V_{22} \text{ is nonsingular} \tag{1.28}$$

Proof :

Assume that there exists a vector $z \neq 0$ such that $V_{22} z = 0$. Then it follows from the singular vector equations

$$[A;B] [V_{12}^T;V_{22}^T]^T = U_2 \Sigma_2 \tag{1.29}$$

that $A(V_{12} z) = U_2 \Sigma_2 z$. Hence,

$$\|A(V_{12} z)\|_2 = \|U_2 \Sigma_2 z\|_2 = \|\Sigma_2 z\|_2 \leq \sigma_{n+1} \|z\|_2 \tag{1.30}$$

From $V_{22} z = 0$ and using the properties of the SVD and the orthonormality of V, we have :

$$\sigma_n' \|V_{12} z\|_2 = \sigma_n' \left\| \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} z \right\|_2 = \sigma_n' \|z\|_2 \leq \|A (V_{12} z)\|_2 \tag{1.31}$$

From the assumption $\sigma_n' > \sigma_{n+1}$, (1.30) and (1.31) are contradictory unless $z = 0$. Thus, V_{22} must be nonsingular. ■

This same condition (1.28) also guarantees **uniqueness** of the generic TLS solution. This follows directly from the following theorem :

Theorem 1-2 : Uniqueness of the generic TLS solution

Let the SVD of $[A;B]$ be given by (0.3) then :

$$\left.\begin{array}{l} \sigma_n > \sigma_{n+1} \text{ and} \\ V_{22} \text{ nonsingular} \end{array}\right\} \Longrightarrow \text{TLS solution } \hat{X} = -V_{12} V_{22}^{-1} \text{ is unique} \tag{1.32}$$

Proof

$R(\hat{B}) \subset R(\hat{A})$ implies that there exists an $X \in \mathbb{R}^{nxd}$ such that

$$\hat{A}X = \hat{B}, \text{ i.e. } [\hat{A};\hat{B}] [X^T;-I]^T = 0.$$

Thus, $[\hat{A};\hat{B}]$ has at most rank n. From the Eckart-Young theorem of the SVD (appendix A), it follows that :

$$\|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 \geq \sum_{i=1}^{d} \sigma_{n+i}^2 \tag{1.33}$$

and that equality results if $[\Delta\hat{A};\Delta\hat{B}]$ is given by (1.22). The assumption $\sigma_n > \sigma_{n+1}$ ensures that $[\Delta\hat{A};\Delta\hat{B}]$ given by (1.22) is the unique minimizer. Since V_{22} is nonsingular its inverse exists and hence, $\hat{X} = -V_{12} V_{22}^{-1}$ solves the TLS problem. ■

Using the interlacing property for singular values (see theorem A.3), $\sigma_n' > \sigma_{n+1}$ implies that $\sigma_n > \sigma_{n+1}$. Combining theorem 1-1 and 1-2, we have :

$$\sigma_n' > \sigma_{n+1} \Longrightarrow \hat{X} = -V_{12} V_{22}^{-1} \text{ is the unique and generic TLS solution of (0.1).} \tag{1.34}$$

If $\sigma_n' \leq \sigma_{n+1}$, the TLS solution may be **nongeneric** or **not unique.**

If σ_{n+1} coincides with σ_n or more generally if $\sigma_{n-p} = \ldots \sigma_{n+1}$ with $p \geq 0$, the TLS solution can still be generic but no longer unique (see section 1.5).

If V_{22} is **singular**, then the TLS solution does not exist in the generic sense but can be obtained by generalizing the TLS solution to nongeneric problems (see section 1.6).

1.5. NON-UNIQUENESS OF THE GENERIC TLS SOLUTION

As mentioned in the previous section, the TLS solution is **no longer unique** if $\sigma_n = \sigma_{n+1}$. Indeed, if $d = 1$ and $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$ with $p \geq 1$, then any linear combination of $v_{n-p+1}, \ldots, v_{n+1}$ resulting in a vector of the form $[\hat{x}^T;-1]^T$, gives a TLS solution $\hat{x}$, satisfying (1.5) and (1.6). Hence it may be desirable to **single out** a **minimum norm** solution. If X is given by (1.25), then, as proven in chapter 2, its **two-norm** is given by :

$$\|\hat{x}\|_2^2 = \|V_{12}\, V_{22}^{-1}\|_2^2 = (1-\sigma_d^2\,(V_{22}))\,/\,\sigma_d^2\,(V_{22}) \tag{1.35}$$

This suggests choosing V_{22} so that its minimal singular value is maximized.

For the **one-dimensional** case $d = 1$ [48, section 12.3], this can be accomplished as follows. Suppose $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$ with $p \geq 0$. If Q is an orthonormal matrix (e.g. Householder matrix) such that

$$[v_{n-p+1}, \ldots, v_{n+1}]Q = \begin{array}{c} \left[\begin{array}{c|c} \diagdown\!\!\diagup & z \\ \hline 0\ldots0 & \gamma \end{array}\right] \begin{array}{l} n \\ 1 \end{array} \\ \begin{array}{cc} p & 1 \end{array} \end{array} \tag{1.36}$$

then $\begin{bmatrix} z \\ \gamma \end{bmatrix}$ has the largest (n+1)-th component of all the vectors in span $\{v_{n-p+1}, \ldots, v_{n+1}\}$. If $\gamma \neq 0$, the minimum norm generic TLS solution is obtained from :

$$\hat{x} = -z/\gamma \tag{1.37}$$

The TLS approximation and corresponding correction matrix of rank 1 satisfying (1.5) and (1.6) are given by :

$$[\hat{A};\hat{b}] = [A-\Delta\hat{A};\ b-\Delta\hat{b}] \quad \text{with } [\Delta\hat{A};\Delta\hat{b}] = [A;b]\begin{bmatrix} z \\ \gamma \end{bmatrix}[z^T;\gamma] \tag{1.38}$$

We also obtain the minimum norm solution $\hat{x}$ **directly** from the vectors $v_{n-p+1}, \ldots, v_{n+1}$ as follows. Let

$$[\hat{x}^T;-1]^T = \sum_{i=n-p+1}^{n+1} \lambda_i\, v_i \tag{1.39}$$

Then λ_i must satisfy the conditions

$$\sum_{i=n-p+1}^{n+1} \lambda_i\, v_{n+1,i} = -1 \quad \text{and} \quad \|[\hat{x}^T;-1]\|_2 \quad \text{minimal}$$

If not all $v_{n+1,i}$ with $i=n-p+1, \ldots, n+1$ are zero, the minimum norm generic TLS solution is :

$$\hat{x} = -\Big(\sum_{i=n-p+1}^{n+1} v_{n+1,i}\,[v_{1,i}, \ldots, v_{n,i}]^T\Big) \Big/ \sum_{i=n-p+1}^{n+1} v_{n+1,i}^2 \tag{1.40}$$

If rank $[A;b] = n-p < n+1$ then $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1} = 0$ or $p+1$ singular values coincide at 0. In this case, $[A;b] = [\hat{A};\hat{b}]$ and the exact minimum norm TLS solution can be found in the same way as described above.

In practice, singular values of $[A;b]$ will **rarely** coincide. Due to **errors** in the data, coinciding singular values of the corresponding unobservable exact relation $A_o x_o = b_o$ become **distinct**. In this case it is realistic to define an error bound ε such that all singular values σ_i, satisfying $|\sigma_i - \sigma_{n+1}| < \varepsilon$, are considered as coinciding with σ_{n+1} and the TLS solution is obtained as above (see also section 1.8.1).

For the **multidimensional** case, i.e. $d > 1$, the minimum norm generic TLS solution can be computed **analogously**. If $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$, we have to select d base vectors $\begin{array}{c} \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} \begin{array}{l} n \\ d \end{array} \\ d \end{array}$ within the span $\{v_{n-p+1}, \ldots, v_{n+d}\}$ such that the minimal singular value of V_{22} is maximized (see (1.35)). This is done by computing an orthonormal matrix Q (e.g. by using Householder transformations) such that

$$[v_{n-p+1}, \ldots, v_{n+d}]\,Q = \begin{bmatrix} V_1\,Q \\ \hline V_2\,Q \end{bmatrix} \begin{array}{l} n \\ d \end{array} = \begin{array}{c} \left[\begin{array}{c|c} \diagdown\!\!\diagup & Z \\ \hline 0\ldots0 & \Gamma \end{array}\right] \begin{array}{l} n \\ d \end{array} \\ \begin{array}{cc} p & d \end{array} \end{array} \tag{1.41}$$

V_2 and V_2Q have the same singular values. From the interlacing property for singular values (see theorem A.3), we know that the singular values of a submatrix of V_2 or V_2Q (obtained by deleting p columns) are never

larger than the corresponding ones of V_2 or V_2Q. In other words, they are maximized if they coincide with those of V_2. Since the d by d submatrix Γ has the same singular values as V_2, we conclude that $\begin{bmatrix} Z \\ \Gamma \end{bmatrix}$ in (1.41) possesses the matrix Γ with largest minimal singular value over all possible sets of d base vectors within span $\{v_{n-p+1}, \ldots, v_{n+d}\}$. Hence if Γ is nonsingular, the minimum norm generic TLS solution $\hat{X}$ follows immediately :

$$\hat{X} = -Z\Gamma^{-1} \tag{1.42}$$

The corresponding correction matrix of rank d

$$[\Delta\hat{A};\Delta\hat{B}] = [A;B]\begin{bmatrix} Z \\ \Gamma \end{bmatrix}[Z^T;\Gamma^T] \tag{1.43}$$

only satisfies (1.19) if $\sigma_{n+1} = \ldots \sigma_{n+d}$. Only in this case, any orthonormal d-dimensional basis $\begin{bmatrix} Z \\ \Gamma \end{bmatrix} \subset \text{span}\{v_{n-p+1}, \ldots, v_{n+d}\}$ is a right singular subspace of $[A;B]$, associated with the same singular value $\sigma_{n-p+1} = \ldots = \sigma_n = \sigma_{n+1} = \ldots = \sigma_{n+d}$. Hence, the Frobenius norm of (1.43) equals $d.\sigma_{n+1}^2$ and satisfies (1.19).

But if **not all** σ_{n+i}, $i = 1, \ldots, d$ **coincide**, then the minimum norm solution computed by (1.42) may have an associated correction matrix (1.43) with Frobenius norm **larger** than the minimal value $\sqrt{\sum_{i=1}^{d} \sigma_{n+i}^2}$. In this case, **two options** are possible :

- The minimization of the norm of $\hat{X}$ is more relevant than that of $[\Delta\hat{A};\Delta\hat{B}]$. Then, the TLS procedure described above is appropriate.
- The minimization of $\|[\Delta\hat{A};\Delta\hat{B}]\|_F$ is more important.

The **first** option is **most realistic** if the set of equations $AX = B$ observe an exact but unknown relation $A_oX_o = B_o$. The p + d smallest singular values of $[A;B]$ are in fact "observations" of the exact minimal singular value of $[A_o;B_o]$ with multiplicity p + d but, due to errors, they no longer coincide. In this case the p + d smallest singular values of $[A;B]$ can indeed be considered as equal and a strict minimization of $\|[\Delta\hat{A};\Delta\hat{B}]\|_F$ makes no sense.

The **second** option makes only sense if the singular values $\sigma_{n+1}, \ldots, \sigma_{n+d}$ differ **significantly**. However, this situation means that the equations $AX = B$ are highly incompatible. If the user would be confronted with such a situation, he might simply conclude that the data are not appropriate for linear modelling and reject the problem as **irrelevant** from a linear modelling point of view. Nevertheless, the situation may occur. Let (0.3) be the SVD of $[A;B]$ and assume $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1} = \ldots = \sigma_{n+r} > \sigma_{n+r+1} \geq \ldots \geq \sigma_{n+d}$.

In order to minimize the approximation effort $\|[\Delta\hat{A};\Delta\hat{B}]\|_F$, the n by d TLS solution $\hat{X}$ must be obtained from the d-r right singular vectors, associated with the d-r smallest singular values of $[A;B]$, and an r-dimensional subset of the (r+p)-dimensional right singular subspace associated with σ_{n+1}, such that $\|\hat{x}\|_2$ given in (1.35), is minimized. Procedures for selecting this subset have not yet been fully analyzed.

1.6. THE NONGENERIC TLS PROBLEM AND SOLUTION

1.6.1. THE ONE-DIMENSIONAL CASE

If $v_{n+1,n+1} = 0$ then the approximation $[\hat{A};\hat{b}]$ proposed in (1.14), does **not** satisfy the TLS condition (1.5). Geometrically, this implies that the vector orthogonal onto the row space of $[\hat{A};\hat{b}]$ is **parallel** with the hyperplane $x_{n+1} = -1$ (see fig. 1.3.b). In the simple case when n = 1, i.e. the fitting of a straight line, this implies that the TLS fitting line in fig. I.1 would be vertical. For those cases, Golub and Van Loan [47] argue that the TLS problem has no solution because (1.16) becomes infinite. In this section however, we claim that a TLS approximation $[\hat{A};\hat{b}]$, satisfying both (1.5) and (1.6), still exists and can be determined by making the next larger singular value σ_n of $[A;b]$ in (1.14) zero. This is equivalent with the case that TLS searches for a solution in a lower dimensional subspace $[\hat{\hat{A}};\hat{\hat{b}}]$, obtained by making one more σ_i in (1.14) zero, i.e. $\sigma_n = \sigma_{n+1} = 0$. Using theorem 1.8 (see further) we can indeed prove that this solution is still optimal with respect to (1.6) and so that (1.5) is satisfied. Moreover it is proven that LS searches its solution in a subspace of the same dimensionality i.e. there doesn't exist a LS solution in a subspace of higher dimensionality than that of the TLS approximation. The need for the extension can be best motivated with an example (see fig. 1.4).

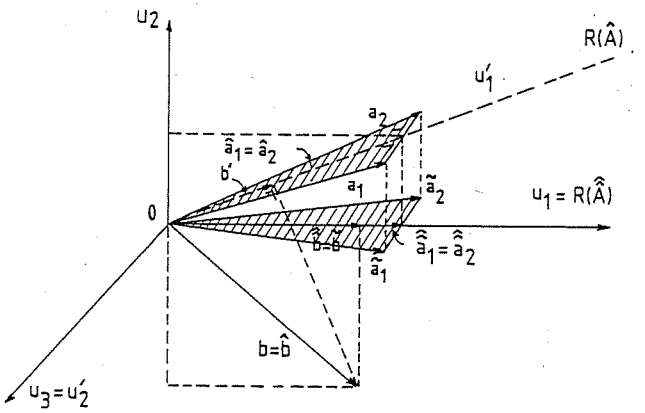


Fig. 1.4 : Geometric representation of example 1-2 with 3 equations in 2 unknowns, characterized by $v_{3,3} = 0$. This implies that $b \perp u_3$.
$\hat{a}_1$, $\hat{a}_2$ and $\hat{b}$ are rank-2 approximations of A and b with $\hat{b} \notin R(\hat{A})$, $\tilde{a}_1$, $\tilde{a}_2$ and $\tilde{b}$ are rank-1 approximations of A and b with $\tilde{b} \in R(\tilde{A})$ and $\tilde{a}_1$, $\tilde{a}_2$ and $\tilde{b}$ are the TLS approximations of A and b with minimal approximation effort $\|[A-\tilde{A};b-\tilde{b}]\|_F$ such that $\tilde{b} \in R(\tilde{A})$.

Example 1-2 :

Consider the set of 3 equations in 2 unknowns :

$$\begin{bmatrix} 2\sqrt{3/2} & 2\sqrt{3/2} \\ \frac{1}{2\sqrt{2}} & -\frac{1}{2\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ -\sqrt{3} \end{bmatrix} \qquad (1.44)$$

Taking the SVD of [A;b] we obtain :

$$U \Sigma V^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 4 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & .5 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{2\sqrt{2}} & \frac{\sqrt{3}}{2\sqrt{2}} & \frac{1}{2} \\ \frac{1}{2\sqrt{2}} & \frac{1}{2\sqrt{2}} & \frac{-\sqrt{3}}{2} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix} \qquad (1.45)$$

Note that $v_{n+1,n+1} = v_{3,3} = 0$. Since $b = \sigma_1 v_{3,1} u_1 + \sigma_2 v_{3,2} u_2 + 0 \cdot u_3$ this implies that $b \perp u_3$. By taking the rank-n approximation (1.14), one obtains the approximating vectors

$$\hat{a}_1 = \hat{a}_2 = 2\sqrt{\tfrac{3}{2}}\, u_1 + \frac{1}{\sqrt{2}} u_2 \quad ; \quad \hat{b} = b \qquad (1.46)$$

Since $\hat{b} \notin R(\hat{A})$ (1.6) is not satisfied. Hence, $[\hat{A};\hat{b}]$ as defined in (1.46) is not a valid TLS approximation and can not produce a TLS solution in the generic sense. Therefore, Golub and Van Loan [47] argue that no TLS solution exists. However, conceptually there is no reason why there is no TLS solution. Moreover the LS solution exists. Indeed from

$$A = U' \Sigma' V'^T = \begin{bmatrix} \frac{2\sqrt{3}}{\sqrt{13}} & 0 \\ 0 & 1 \\ \frac{1}{\sqrt{13}} & 0 \end{bmatrix} \begin{bmatrix} \sqrt{13} & 0 \\ 0 & .5 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \qquad (1.47)$$

one obtains the LS solution x' as

$x' = V' \Sigma'^{-1} U'^T b =$

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{13}} & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2\frac{\sqrt{3}}{\sqrt{13}} & 0 & \frac{1}{\sqrt{13}} \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 0 \\ -\sqrt{3} \end{bmatrix} = \begin{bmatrix} \frac{3}{13} & \frac{\sqrt{3}}{\sqrt{2}} \\ \frac{3}{13} & \frac{\sqrt{3}}{\sqrt{2}} \end{bmatrix} \qquad (1.48)$$

A generalized TLS solution can be obtained by dropping one more singular triplet in (1.45) i.e. by replacing σ_2 and σ_3 by zero. Then one obtains the TLS approximation $[\hat{A};\hat{b}]$ such that $\hat{b} \in R(\hat{A})$. $\hat{a}_1 = \hat{a}_2 = 2\sqrt{3/2}\, u_1$ and $\hat{b} = 2\, u_1$. Note that $\hat{b}$ remains orthogonal onto u_3. Now $\hat{b} \in R(\hat{A})$ and the generalized TLS solution is obtained from (1.40) :

$$\hat{x}^T = [-1/\sqrt{6},\ -1/\sqrt{6}]\ . \tag{1.49}$$

As $v_{n+1,n+1} = 0$, $[\hat{x}^T;-1]^T$ is orthogonal onto v_3 and hence, $[\hat{x}^T;-1]^T$ is parallel with v_2. We only have to approximate $[A;b]$ by making only σ_2 in (1.45) zero, in order to satisfy (1.5). As the total approximation effort must be minimal, i.e. (1.6) must be satisfied, σ_3 need not be zeroed. Hence, the TLS approximation which satisfies (1.5) and (1.6), is :

$$[\tilde{A};\tilde{b}] = U \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix} V^T \tag{1.50}$$

and the corresponding correction matrix of rank 1 is given by (1.38) :

$$[\Delta\tilde{A};\Delta\tilde{b}] = \sigma_2\, u_2\, v_2^T = [A;b] \begin{bmatrix} \hat{x} \\ -1 \end{bmatrix} [\hat{x}^T;-1] \tag{1.51}$$

Observe that $\sigma_2' = \sigma_3 = .5$ and $u_2' = u_3$. Hence $b \perp u_2'$ and b' remains orthogonal on u_2'. Al those facts are generalized and proven in theorem 1-3. Observe from fig. 1.4 that the case where the generic TLS solution does not exist, only happens when the length $\|b-b'\|_2$ of the orthogonal projection of b onto R(A) is **larger** than the length σ_n' of the projection of all columns a_i of A onto its lowest singular vector. This is the case when A is (nearly) **rank-deficient** or when the set of equations is **highly incompatible** like in this example.

Theorem 1-3 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., $[A;b]$). Let b' be the orthogonal projection of b onto R(A) and $[\hat{A};\hat{b}]$ the rank-n approximation of $[A;b]$, as given by (1.14) or (1.38). If $V'(\sigma_j)$ (resp., $U'(\sigma_j)$) is the right (resp., left) singular subspace of A associated with σ_j, then the following relations can be proven :

(a) $v_{n+1,j} = 0 \iff v_j = \begin{bmatrix} v' \\ 0 \end{bmatrix}$ with $v' \in V'(\sigma_j)$

(b) $\Longrightarrow \sigma_j = \sigma_k'$ with $k = j-1$ or $k = j$ and $1 \leq k \leq n$

(c) $\Longrightarrow b \perp u'$ with $u' \in U'(\sigma_j)$

(d) $\Longleftrightarrow u_j = u'$ with $u' \in U'(\sigma_j)$

(e) $\Longrightarrow b' \perp u'$ with $u' \in U'(\sigma_j)$

(f) $\Longrightarrow \hat{b} \perp u'$ with $u' \in U'(\sigma_j)$ (1.52)

If σ_j is an isolated singular value, the converse of relation (c) and (e) also holds.

Proof :

(a) $\Longleftarrow$ obvious

$\Longrightarrow$ Since v_j is a right singular vector of $[A;b]$, it is an eigenvector of $[A;b]^T[A;b]$ and satisfies :

$$\begin{bmatrix} A^TA & A^Tb \\ b^TA & b^Tb \end{bmatrix} v_j = \sigma_j^2\, v_j \tag{1.53}$$

Since $v_{n+1,j} = 0$, v_j has to be of the form $v_j = \begin{bmatrix} v_j'' \\ 0 \end{bmatrix}$ with v_j'' an n-dimensional vector. Hence, (1.53) reduces to :

$$\begin{cases} A^TA\, v_j'' = \sigma_j^2\, v_j'' & (1.54) \\ b^TAv_j'' = 0 & (1.55) \end{cases}$$

From (1.54), it follows that v_j'' is an eigenvector of A^TA with corresponding eigenvalue σ_j^2 and thus, v_j'' must be a right singular vector v' of A with corresponding singular value σ_j.

(b) The proof follows immediately from (1.54). Since $v_j'' = v'$ is a singular vector of A, the corresponding singular value σ_k' of A must be equal to σ_j. Based on the interlacing property for singular values (see theorem A.3), it is clear that σ_j can either be equal to the previous or the corresponding singular value of A.

(c) The proof follows immediately from (1.55). Indeed, since $v_j'' = v'$ as proven in (a), (1.55) becomes :

$$b^T (Av') = 0 \iff b \perp Av' \tag{1.56}$$

Using the definition of the SVD there exists a left singular vector $u' \in U'(\sigma_j)$ such that :

$$Av' = \sigma_j u' \tag{1.57}$$

Together with (1.56) this proves relation (c).

(d) $\Leftarrow$ Since u' (resp., u_j) is a left singular vector of A (resp., [A;b]), associated with σ_j, we have from the definition of the SVD :

$$A^T u' = \sigma_j v' \tag{1.58}$$

$$[A;b]^T u_j = \sigma_j v_j \tag{1.59}$$

with v' (resp., v_j) the corresponding right singular vector associated with σ_j. Partitioning v_j as $\begin{bmatrix} v_j'' \\ v_{n+1,j} \end{bmatrix}$ with v_j'' an n-dimensional vector, (1.59) becomes :

$$A^T u_j = \sigma_j v_j'' \tag{1.60}$$

$$b^T u_j = \sigma_j v_{n+1,j} \tag{1.61}$$

Since $u_j = u'$, (1.60) becomes :

$$A^T u_j = A^T u' = \sigma_j v_j'' \tag{1.62}$$

From (1.58) and (1.62), one obtains $v_j'' = v'$. Hence,

$$[A;b]\, v_j = A\, v_j'' + b\, v_{n+1,j} = A\, v' + b\, v_{n+1,j} = \sigma_j u' + b\, v_{n+1,j} \tag{1.63}$$

Since $[A;b]\, v_j = \sigma_j u'$, we conclude from (1.63) that

$$v_{n+1,j} = 0 \qquad \text{q.e.d.}$$

$\Rightarrow$ Using relation (c), we know that b must be orthogonal onto u' with $u' \in U'(\sigma_j)$. Since u' is a left singular vector of A with corresponding singular value σ_j and thus an eigenvector of AA^T with corresponding eigenvalue σ_j^2 by definition, we have :

$$AA^T u' = \sigma_j^2 u' \tag{1.64}$$

Using (c) we can also write :

$$(AA^T + bb^T)\, u' = \sigma_j^2 u' \tag{1.65}$$

Since $[A;b]\,[A;b]^T = AA^T + bb^T$, u' must be an eigenvector of $[A;b]\,[A;b]^T$ with corresponding eigenvalue σ_j^2. Hence, [A;b] has a left singular vector $u_j = u'$ corresponding to the singular value σ_j. q.e.d.

(e) Since b' is the orthogonal projection of b onto R(A), it can be written as :

$$b' = U_1' (U_1'^T b) \tag{1.66}$$

Using (c), we obtain from (1.66) :

$$b'^T u' = b^T U_1' U_1'^T u' = b^T u' = 0 \iff b' \perp u' \quad \text{q.e.d.} \tag{1.67}$$

(f) Since $\hat{b}$ is the orthogonal projection of b onto $\{u_1, \ldots, u_n\}$, it is given by :

$$\hat{b} = \sum_{i=1}^{n} u_i u_i^T b = \sum_{i=1}^{n} \sigma_i v_{n+1,i} u_i \tag{1.68}$$

If $j = n+1$, the proof follows immediately using relation (d).
If $j \leq n$, we obtain from (1.68), using relations (c) and (d) :

$$\hat{b}^T u' = \hat{b}^T u_j = b^T u_j = \sigma_j v_{n+1,j} = 0 \iff \hat{b} \perp u' \quad \text{q.e.d.} \tag{1.69}$$

If σ_j is an isolated singular value, than the singular vectors u_j and v_j associated with σ_j, are unique. In this case, the converse of relation (c) and (e) can be proven.

(c) $\Leftarrow$ Since u' is a left singular vector of A associated with σ_j :

$$AA^T u' = \sigma_j^2 u' \tag{1.70}$$

Since $b \perp u'$, we also have :

$$(AA^T + bb^T)\, u' = AA^T u' = \sigma_j^2 u' \tag{1.71}$$

Hence, u' is a singular vector of $[A;b]$ associated with σ_j. Since σ_j is distinct, its corresponding right singular vector u_j is unique and thus, $u' = u_j$.
Using the converse relation (d), the proof follows immediately.

(e) $\Leftarrow$ The converse proof follows immediately from the definition of b'. As b' is the orthogonal projection of b onto R(A), $b' \perp u'$ implies that $b \perp u'$. Hence, using the converse relation (c), the proof follows immediately. ■

As said before, the generic TLS solution does not exist if $v_{n+1,n+1} = 0$. Theorem 1-3 describes the properties of this situation, namely : if $v_{n+1,n+1} = 0$, then $\sigma_{n+1} = \sigma'_n$. This result can also be derived from theorem 1-1 for d = 1 by contraposition. If additionnally $\sigma_n > \sigma_{n+1}$ then $u_{n+1} = u'_n$, $v_{n+1} = \begin{bmatrix} +v'_n \\ 0 \end{bmatrix}$ and b, b' as well as $\tilde{b}$ are orthogonal onto u'_n. **For those cases we want to prove now that a TLS solution $\hat{x}$ satisfying (1.5) and (1.6) still exists and how it is deduced.** This is done in the next theorem which includes all cases in which the generic TLS solution fails to exist.

Theorem 1-4 : Let (0.3) be the SVD of $[A;b]$ and $p \leqslant n$.

Assume $v_{n+1,j} = 0$ for $j = p+1, \ldots, n+1$ and $v_{n+1,p} \neq 0$. Then, if $\sigma_{p-1} > \sigma_p$:

$$[\hat{x}^T;-1]^T = -v_p \,/\, v_{n+1,p} \tag{1.72}$$

is the unique nongeneric TLS solution satisfying (1.5) and (1.6) with

$$[\Delta\hat{A};\Delta\hat{b}] = \sigma_p\, u_p\, v_p^T \tag{1.73}$$

$$[\hat{A};\hat{b}] = U\, \hat{\Sigma}\, V^T \quad \text{with} \quad \hat{\Sigma} = \text{diag}\,(\sigma_1, \ldots, \sigma_{p-1}, 0, \sigma_{p+1}, \ldots, \sigma_{n+1}) \tag{1.74}$$

Proof :

If $v_{n+1,j} = 0$ for $j = p+1, \ldots, n+1$, then using theorem 1-3 this implies that b (and $\tilde{b}$) $\perp u_j$ and $v_j = \begin{bmatrix} v''_j \\ 0 \end{bmatrix}$ for $j = p+1, \ldots, n+1$.
Hence b can only be approximated in the space generated by the remaining u_j, $j = 1, \ldots, p$. This means that there is no TLS solution along v_j, $j = p+1, \ldots, n+1$. Hence, $[\hat{x}^T;-1]^T$ can only be written as linear combination of the remaining v_j, $j = 1, \ldots, p$.

The nongeneric TLS then searches in the p-dimensional space $R[A;b]_p$ with $[A;b]_p = U\, \hat{\Sigma}_p\, V^T$ and $\hat{\Sigma}_p = \text{diag}\,(\sigma_1, \ldots, \sigma_p, 0, \ldots, 0)$ for an approximation $[A;b]_{p-1}$ of rank p-1 with minimal deviation such that (1.5) is satisfied. From the Eckart-Young theorem (see theorem A.2) this approximation of $[A;b]$, minimizing (1.6), is given by

$$[A;b]_{p-1} = \sum_{i=1}^{p-1} \sigma_i . u_i . v_i^T \tag{1.75}$$

obtained by making σ_p in $[\hat{A};\hat{b}]_p$ zero.

Since $\sigma_p = \ldots = \sigma_{n+1}$ of $[A;b]_{p-1}$ are zero, $[A;b]_{p-1}$ has a singular value σ_{n+1} with multiplicity n-p+2. We can compute the minimum norm solution (1.40). As not all $v_{n+1,j}$ with $j = p, \ldots, n+1$ are zero, (1.40) is a TLS solution of $[A;b]_p\, [\hat{x}^T;-1]^T = 0$. It is moreover unique as only one $v_{n+1,p} \neq 0$. From (1.38), the correction matrix (1.73) follows immediately. Subtracting (1.73) from $[A;b]$, the TLS approximation (1.74) is obtained and satisfies (1.5) and (1.6). Observe that $[A;b]_{p-1}$ satisfies (1.5) but not (1.6). Hence it is not a TLS approximation, although it produces the same TLS solution $\hat{x}$. ■

Comments :

- The **condition $\sigma_{p-1} > \sigma_p$ is not a restriction**, and is only used here for reasons of simplicity. Indeed, if $\sigma_{p-1} = \sigma_p$, then the nongeneric TLS solution still exists. One must take the **minimum norm** solution $[\hat{x}^T;-1]^T$ in the r-dimensional right singular subspace $V(\sigma_p)$ of $[A;b]$, associated with σ_p of multiplicity r. The correction matrix is given by (1.38). However, if several $v_{n+1,j} \neq 0$ for $j = p-r+1, \ldots, p$, the **nongeneric TLS solution is no longer unique.**

- If $v_{n+1,n+1} = 0$, $v_{n+1,n} \neq 0$ and $\sigma_{n-1} > \sigma_n$, the nongeneric TLS solution

follows directly from (1.72) in theorem 1-4 :

$$[\hat{x}^T;-1]^T = -v_n / v_{n+1,n} \tag{1.76}$$

with corresponding correction matrix $[\Delta\hat{A};\Delta\hat{b}] = \sigma_n u_n v_n^T$ having minimal Frobenius norm σ_n such that $(b-\Delta\hat{b}) \in R(A-\Delta\hat{A})$.

- Let us also **compare LS and TLS in the nongeneric case.** LS looks for a solution x' such that $\sum_{i=1}^{n} \sigma'_i u'_i v'^T_i x' = b'$. If $v_j = \begin{bmatrix} v''_j \\ v_{n+1,j} \end{bmatrix}$ with $v_{n+1,j} = 0$ for j = p+1, ..., n+1, then using theorem 1-3, this implies that $\sigma'_{j-1} = \sigma_j$ and b as well as b' are orthogonal onto $u_j = u'$ with $u' \in U'(\sigma_j)$ for j = p+1, ..., n+1. Or, $[A;b]_p = [A_{p-1};b]$ with $[A;b]_p$ (resp., A_{p-1}) the rank-p (resp., rank-(p-1)) approximation of [A;b] (resp., A), as derived from the Eckart-Young theorem (see theorem (A.2)). Also, $x' \perp v''_j$, j = p+1, ..., n+1. Hence, LS must also look for a solution x' in the (p-1)-dimensional row space $R_R(A_{p-1})$ of A_{p-1}, orthogonal onto span $\{v''_j \mid j = p+1, \ldots, n+1\}$, which equals span $\{v'_j \mid j = p, \ldots, n\}$.
Thus b, as well as $\hat{b}$ and b', are orthogonal onto the same n-p+1 singular vectors u_j, j = p+1, ..., n+1. TLS as well as LS search for an optimal solution in a subspace of same dimension p-1 with p the largest index such that $v_{n+1,p} \neq 0$ i.e. dim $(R[A_{p-1};b']) = \dim (R[A_{p-1};b]_{p-1}) = p-1$. Hence, it follows that :

$$\|b-b'\|_2 = \|[A_{p-1};b] - [A_{p-1};b']\|_F \geq \min_{\text{rank}(C)=p-1} \|[A_{p-1};b] - C\|_F =$$

$$\sigma_p = \|[\Delta\hat{A};\Delta\hat{b}]\|_F \tag{1.77}$$

with $[\Delta\hat{A};\Delta\hat{b}]$ given by (1.73). The **LS approximation effort** remains **larger** than the TLS approximation effort.
Observe that the **generic** TLS solution does **not exist**, if the set of equations Ax = b is **highly incompatible** (so that $\|b-b'\|_2$ is large and exceeds σ'_p of A), or if A is **nearly rank-deficient** (i.e. $\sigma'_p \approx \ldots \approx \sigma'_n \approx 0$).

- In the extreme case where b is orthogonal on R(A), TLS as well as LS give the zero solution : as $b \perp R(A)$, b' = 0. Hence, x' is zero. The zero TLS solution follows from direct application of theorem 1-4. In-

deed, the set of left singular vectors of [A;b] is precisely $\{u'_1, u'_2, \ldots, b / \|b\|_2, u'_2, \ldots, u'_n\}$. This is the set of left singular vectors of A where b is introduced at the i-th location corresponding to its size $\sigma'_{i-1} \geq \|b\|_2 \geq \sigma'_i$. Correspondingly, the i-th column v_i of V is [0, ..., 0, 1] and the last row of V is [0, ..., 0, 1, ..., 0] with its i-th element equal to one. Since $v_{n+1,j} = 0$ for j = i+1, ..., n+1 and $v_{n+1,i} \neq 0$, the TLS solution $[\hat{x}^T;-1]^T$ is given by $-v_i / v_{n+1,i}$. Hence, $\hat{x}$ is zero.

- In practice $v_{n+1,j}$ will seldom be zero. Indeed, due to errors in the observations [A;b], a zero-valued $v_{n+1,j}$ in the SVD of the unobservable exact data matrix $[A_o;b_o]$, will differ from zero in the SVD of [A;b]. Hence it is advisable to define an error bound ε such that all $|v_{n+1,j}| < \varepsilon$ are considered to be zero (see section 1.8.1).

1.6.2. THE MULTIDIMENSIONAL CASE

When the set of equations AX = B has **more than one** observation vector the generic TLS solution $\hat{X}$ does not exist if V_{22} is **singular**. This happens if $\sigma'_n \leq \sigma_{n+1}$. Indeed :

Theorem 1-5 :

Let (0.3) be the SVD of [A;B] and the SVD of A be given by (0.2) :

$$V_{22} \text{ singular} \Longrightarrow \sigma_{n+d} \leq \sigma'_n \leq \sigma_{n+1} \tag{1.78}$$

Proof

From theorem 1-1, one obtains :

$$V_{22} \text{ singular} \Longrightarrow \sigma'_n \leq \sigma_{n+1} \tag{1.79}$$

Using the interlacing property for singular values (see theorem A.3), σ'_n is bounded as follows :

$$\sigma_{n+d} \leq \sigma'_n \leq \sigma_n \tag{1.80}$$

From (1.79) and (1.80), one obtains (1.78). ■

In order to **generalize** theorem 1-3 for d > 1, it is convenient to describe the singularity of V_{22} in function of its **kernel**, denoted by Ker (V_{22}).

Theorem 1-6 :

Let (0.3) be the SVD of [A;B]. If Ker (V_{22}) represents the kernel of V_{22}, then :

$$V_{22} \text{ singular} \Leftrightarrow \exists x \neq 0,\ x \in \text{Ker } (V_{22}) : \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} x = \begin{bmatrix} y \\ 0 \end{bmatrix} \begin{matrix} n \\ d \end{matrix}$$

$$\text{with } \|x\|_2 = \|y\|_2 \qquad (1.81)$$

Proof :

The proof is a straightforward application of the definition of singularity :

$$V_{22} \text{ singular} \Leftrightarrow \exists x \neq 0 : V_{22}\, x = 0 \qquad (1.82)$$

and the definition of a kernel :

$$x \in \text{Ker } (V_{22}) \Leftrightarrow V_{22}\, x = 0 \qquad (1.83)$$

Hence,

$$\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} x = \begin{bmatrix} V_{12}\, x \\ V_{22}\, x \end{bmatrix} = \begin{bmatrix} V_{12}\, x \\ 0 \end{bmatrix} \qquad (1.84)$$

Taking $y = V_{12}x$, one obtains (1.81). Since V is an orthonormal matrix, we have :

$$\left\| \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} x \right\|_2 = \|x\|_2 = \|y\|_2 \qquad (1.85)$$

■

Observe that the case V_{22} **singular** only happens when $\sigma'_n \leq \sigma_{n+1}$, i.e. the length of the projection σ'_n of all columns a_i of A onto its lowest singular vector is smaller than the length of the projection of all columns of [A;B] onto its (n+1)-th singular vector, associated with σ_{n+1}. This is the case if A is **(nearly) rank-deficient**, i.e. $\sigma'_n \approx 0$, or when the

set of equations (0.1) (or at least one subset $Ax_i = b_i$) is **highly incompatible.**

Before describing the properties of this situation, we first **prove that a TLS solution $\hat{X}$** satisfying (1.18) and (1.19) **still exists** and how it is deduced. This is done in the next theorem which generalizes theorem 1-4 and **includes all cases** in which the **generic** multidimensional **TLS solution fails to exist.**

Theorem 1-7 :

Let (0.3) be the SVD of [A;B]. Assume $\sigma_p > \sigma_{p+1} = \ldots = \sigma_{n+1}$. Let Q_1 be an orthonormal matrix such that $[v_{p+1}, \ldots, v_{n+d}]\, Q_1 = \left[\begin{array}{c|c} Y_1 & Z_1 \\ \hline 0 & \Gamma_1 \end{array}\right] \begin{matrix} n \\ d \end{matrix}$ (1.86)

$$\begin{matrix} \\ n-p \end{matrix} \quad \begin{matrix} \\ d \end{matrix}$$

with Γ_1 singular of corank κ and $p \leq n$.

Let Q be an orthonormal matrix such that $[v_{q+1}, \ldots, v_{n+d}]Q = \left[\begin{array}{c|c} Y & Z \\ \hline 0 & \Gamma \end{array}\right] \begin{matrix} n \\ d \end{matrix}$ (1.87)

$$\begin{matrix} n-q & d \end{matrix}$$

with minimal n-q such that Γ is nonsingular ($p-q \geq \kappa$). Assume $\sigma_q > \sigma_{q+1}$. Then

$$\hat{X} = -\, Z\, \Gamma^{-1} \qquad (1.88)$$

is the unique nongeneric TLS solution satisfying (1.18) and (1.19) with

$$[\Delta\hat{A};\Delta\hat{B}] = [A;B] \begin{bmatrix} Z \\ \Gamma \end{bmatrix} [Z^T;\Gamma^T] \qquad (1.89)$$

Proof :

As $\sigma_p > \sigma_{p+1} = \ldots = \sigma_{n+1}$, the solution space, i.e. the orthogonal row complement of the approximation $[A;B]_p = \sum_{i=1}^{p} \sigma_i\, u_i\, v_i^T$, is span $\{v_{p+1}, \ldots, v_{n+d}\}$ within which the d-dimensional generic minimum norm TLS solution $[\hat{X}^T;-I_d]^T$ must be singled out (section 1.5). In order to satisfy (1.18), this span must contain a d-dimensional subspace $\begin{bmatrix} Z \\ \Gamma \end{bmatrix}$ such that Γ is nonsingular.

However, the assumption Γ_1 singular means that no d-dimensional subspace $\begin{bmatrix} Z \\ \Gamma \end{bmatrix}$ of span $\{v_{p+1}, \ldots, v_{n+d}\}$ exists with nonsingular Γ. (Since Γ^{-1} does

not exist, no solution $\begin{bmatrix}\hat{X}\\ -I\end{bmatrix} // \begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ with $\hat{X} = -Z\Gamma^{-1}$ can be deduced. Hence, (1.18) can not be satisfied and thus, the TLS solution can not be computed in the generic sense (section 1.5). Hence we must extend the solution space until it contains a d-dimensional subspace $\begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ with Γ nonsingular. Hereto, the nongeneric TLS searches for an approximation $[A;b]_q$ of $[A;b]$ of rank $q < p$ with minimal deviation such that (1.18) and (1.19) are satisfied. From the Eckart-Young theorem (see theorem A.2), this rank q approximation of $[A;b]$ minimizing (1.19), is given by

$$[A;b]_q = \sum_{i=1}^{q} \sigma_i u_i v_i^T \qquad (1.90)$$

obtained by making the next singular values $\sigma_{q+1}, \ldots, \sigma_p$ of $[A;b]_p$ zero. The orthogonal complement of the approximation $[A;B]_q$, i.e. the solution space given by span $\{v_{q+1}, \ldots, v_{n+d}\}$, is hereby extended and must contain a subspace $\begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ with Γ nonsingular such that (1.18) is satisfied. Since $\sigma_{q+1} = \ldots = \sigma_{n+d}$ of $[A;b]_q$ are zero, $[A;b]_q$ has a singular value σ_{n+1} of multiplicity n-q+d. We can compute the minimum norm solution (1.49) of $[A;b]_q$ $[\hat{X}^T;-I_d]^T = 0$ and obtain (1.88). As Γ in (1.87) is nonsingular, (1.18) is satisfied. From (1.43) the corresponding correction matrix (1.89) follows immediately. The approximation $[A;b]_q$ satisfies (1.18) but not (1.19). In order to obtain the TLS approximation with minimal approximation effort corresponding to this solution (1.88), i.e. (1.17) and (1.18) are satisfied, we subtract (1.89) from $[A;B]$. Moreover this solution (1.88) is unique since we assumed n-q minimal. This means that span $\{v_{q+1}, \ldots, v_{n+d}\}$ contains only one subspace $\begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ with nonsingular Γ. This implies also that no subspace $\begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ of span $\{v_{q+2}, \ldots, v_{n+d}\}$ exists with nonsingular Γ. At least one base vector v_{q+1} must be added to satisfy (1.18), i.e. to find a d-dimensional basis $\begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ such that Γ is nonsingular and (1.88) exists.
The computed basis $\begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ is orthogonal onto all base vectors of span $\{v_{q+1}, \ldots, v_{n+d}\}$ which make Γ singular. Indeed those base vectors can be written as $\begin{bmatrix}w\\ o\end{bmatrix}\begin{matrix}n\\ d\end{matrix}$ according to theorem 1-6. Since Γ is nonsingular, each vector of the form $\begin{bmatrix}w\\ o\end{bmatrix}$ within the span $\{v_{q+1}, \ldots, v_{n+d}\}$ can only be generated by base vectors of the span $\begin{bmatrix}Y\\ o\end{bmatrix}\begin{matrix}n\\ d\end{matrix}$ in (1.87). Those base vectors are orthogonal onto the base vectors of $\begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ and hence also orthogonal onto the solution space $\begin{bmatrix}X\\ -I\end{bmatrix} // \begin{bmatrix}Z\\ \Gamma\end{bmatrix}$. Since corank $(\Gamma_1) = \kappa$, Ker (Γ_1) has κ base vectors. Therefore, the dimension n-q of $\begin{bmatrix}Y\\ o\end{bmatrix}$ in (1.87) must be at least $n-p+\kappa$, i.e. $p-q \geq \kappa$. ■

Comments :

- If $\sigma_n > \sigma_{n+1}$ and V_{22} singular (case $p = 0$) or $\sigma_p > \sigma_{p+1} = \ldots = \sigma_{n+1}$ and Γ_1 singular (case of non-uniqueness), the **generic** TLS solution, as computed by the algorithm of Golub and Van Loan, **fails** to exist. For those cases, theorem 1-7 **proves** that **a TLS solution**, called **nongeneric**, satisfying the TLS criteria (1.18) and (1.19), still **exists** and how it is computed : if the corank κ of V_{22} or Γ_1 is ≥ 1, we must add the minimum number (at least κ) of right singular vectors associated with the smallest singular values of $[A;B]$, until a nonsingular Γ in (1.87) is obtained. The basis $\begin{bmatrix}Z\\ \Gamma\end{bmatrix}$ is orthogonal onto the remaining independent base vectors $\begin{bmatrix}Y\\ o\end{bmatrix}$ of span $\{v_{q+1}, \ldots, v_{n+d}\}$ which make Γ singular. The latter vectors belong to the kernel of the submatrix A_q of the q-th rank approximation $[A;B]_q$ of $[A;B]$.

- The **condition** $\sigma_q > \sigma_{q+1}$ **is not a restriction** and is only used here for reasons of simplicity. Indeed, if $\sigma_q = \sigma_{q+1}$ then the nongeneric TLS solution still exists and one must take the **minimum norm** solution $[\hat{X}^T; -I_d]^T$ in the space generated by $[V(\sigma_{q+1}), v_{q+2}, \ldots, v_{n+d}]$ where $V(\sigma_{q+1})$ is the r-dimensional singular subspace, associated with the singular value σ_{q+1} of multiplicity r. In this case, the nongeneric TLS solution is however no longer unique but one singles out the minimum norm nongeneric TLS solution to preserve uniqueness.

In the following theorems, we describe the **properties in case of singularity** of V_{22} (or Γ_1) by **generalizing** the properties given in theorem 1-3 to the multidimensional case. Observe that **additional conditions** concerning the multiplicity of σ_{n+1} are imposed.
Let us first assume that the **unique generic multidimensional TLS solution does not exist**, i.e. $\sigma_n > \sigma_{n+1}$ and V_{22} is singular.

Theorem 1-8 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., $[A;B]$). Let $\sigma_n > \sigma_{n+1}$. Call $V'(\sigma_j)$ (resp., $U'(\sigma_j)$) the right (resp., left) singular subspace of A associated with σ_j, and $V(\sigma_j)$ (resp., $U(\sigma_j)$) the right (resp., left) singular subspace of $[A;B]$, associated with σ_j. Consider any unit vector $x \in$ Ker (V_{22}) and let b (resp., r) be the index of the first (resp., last) nonzero component of x.

Then the following relations can be proven.

If $\sigma_{n+b} = \ldots = \sigma_j = \ldots = \sigma_{n+r}$ then

(a) $\exists v \in V(\sigma_j)$, $\exists v' \in V'(\sigma_j)$: $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} x = v = \begin{bmatrix} v' \\ 0 \end{bmatrix}$

(b) $\exists i : \sigma_j = \sigma'_i$

(c) $\exists u' \in U'(\sigma_j) : B \perp u'$

(d) $\exists u \in U(\sigma_j)$, $\exists u' \in U'(\sigma_j) : u = u'$

(e) $\exists u' \in U'(\sigma_j) : B' \perp u'$

(f) $\exists u' \in U'(\sigma_j) : \hat{B} \perp u'$ (1.91)

Proof :

(a) If V_{22} is singular then, using theorem 1-6, there exists an $x \in \text{Ker}(V_{22})$, $\|x\|_2 = 1$, such that $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} x = \begin{bmatrix} y \\ 0 \end{bmatrix}$, $\|y\|_2 = 1$. We have to prove now that y is a right singular vector of A if the additional conditions are imposed. Since the columns $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix}$ are eigenvectors of $[A;B]^T[A;B]$ associated with the eigenvalues Σ_2^2, we obtain :

$$\begin{bmatrix} A^TA & A^TB \\ B^TA & B^TB \end{bmatrix} \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} x = \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} \Sigma_2^2 x \qquad (1.92)$$

The condition $\sigma_{n+b} = \ldots = \sigma_j = \ldots = \sigma_{n+r}$ with b and r defined above, implies that $\Sigma_2^2 x = \sigma_j^2 x$. Hence,

$$A^TA (V_{12} x) = \sigma_j^2 (V_{12} x) \qquad (1.93)$$

$$B^TA (V_{12} x) = \sigma_j^2 V_{22} x = [0, \ldots, 0]^T \qquad (1.94)$$

Since (1.93) are the eigenvector equations of A^TA, associated with the eigenvalue σ_j^2, we conclude that $V_{12} x = y = v'$ with $v' \in V'(\sigma_j)$.

Hence $V_{12} x$ is a right singular vector of A associated with the singular value σ_j. As the b-th to r-th column of $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix}$ are a basis of the singular subspace of [A;B], associated with the singular value $\sigma_{n+b} = \ldots = \sigma_j = \ldots = \sigma_{n+r}$, every linear combination of those columns is a right singular vector of [A;B], associated with the singular value σ_j or $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} x = v \in V(\sigma_j)$.

(b) The proof follows directly from (a). If v' (resp., v) is a singular vector of A (resp., [A;B]) associated with σ_j, A must have a singular value $\sigma'_i = \sigma_j$. The bounds are derived from theorem 1-5 and the interlacing property for singular values. Since v must be a linear combination of the columns of $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix}$, the bounds $n+1 \leq j \leq n+d$ follow immediately.

From theorem 1-5, V_{22} can only be singular if $\sigma'_n \leq \sigma_{n+1}$. Not only the smallest singular value of A can be $\leq \sigma_{n+1}$ but, from the interlacing property for singular values, we have : $\sigma_{n+d-k} \leq \sigma'_{n-k} \leq \sigma_{n+1-k}$ (1.95)

V_{22} may be singular as long as $\sigma'_{n-k} \leq \sigma_{n+1}$. From (1.95), this implies that $k \leq d-1$ from which the bounds for σ'_i are deduced.

(c) The proof follows from relation (a). Let u' be the left singular vector of A, associated with σ_j, then we obtain from (1.94) :

$$B^T A(V_{12} x) = B^T (Av') = \sigma_j B^T u' = [0, \ldots, 0]^T \qquad (1.96)$$

Hence $B \perp u'$ with $u' \in U'(\sigma_j)$.

(d) Since u' is a left singular vector of A associated with σ_j, we obtain from the corresponding eigenvector equations :

$$AA^T u' = \sigma_j^2 u' \qquad (1.97)$$

Using relation (c), $B \perp u' \in U'(\sigma_j)$. Hence, we can also write :

$$(AA^T + BB^T) u' = AA^T u' = \sigma_j^2 u' \qquad (1.98)$$

(1.98) are the eigenvector equations of [A;B] associated with σ_j.

Hence, u' is also a singular vector of $[A;B]$ associated with σ_j. And thus, $\exists u \in U(\sigma_j)$ such that $u = u'$. q.e.d.

(e) Since B' is the orthogonal projection of B onto $R(A)$, it can be written as :

$$B' = U_1' U_1'^T B \tag{1.99}$$

Using relation (c), this implies :

$$B'^T u' = B^T U_1' U_1'^T u' = B^T u' = [0, \ldots, 0]^T \tag{1.100}$$

Hence $B' \perp u' \in U'(\sigma_j)$. q.e.d.

(f) Since $\hat{B}$ is the orthogonal projection of B onto U_1, it is given by :

$$\hat{B} = U_1 \Sigma_1 V_{12}^T \tag{1.101}$$

Using relation (c) and (d), $B \perp u' \in U'(\sigma_j)$ and $u = u'$ with $u \in U(\sigma_j)$ with $n+1 \leq j \leq n+d$. This implies that $u \in U_2$. From (1.101), $\hat{B}$ is orthogonal onto U_2 and hence, $\hat{B} \perp u' \in U'(\sigma_j)$. q.e.d. ■

Comments :

- If all components of $x \in \text{Ker}\,(V_{22})$ differ from zero, the additional conditions are $\sigma_{n+1} = \ldots = \sigma_{n+d}$. In this case, every linear combination of the columns of $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix}$ is a right singular vector of $[A;B]$, associated with σ_{n+1}.

- For the one-dimensional case ($d = 1$) with $\sigma_n > \sigma_{n+1}$ and $v_{n+1,n+1} = 0$, no additional conditions are required and theorem 1-8 reduces to a special case of theorem 1-3.

- **Assuming that the conditions of theorem 1-8** are satisfied, let us consider the **properties of the nongeneric TLS solution** (1.88), as deduced in theorem 1-7. Assume that a nonsingular Γ can be obtained in (1.87) of theorem 1-7 for $q = n-1$ and $\sigma_{n-1} > \sigma_n$. In this case, **corank** $(V_{22}) = 1$. Consider $\sigma'_{n-1} > \sigma'_n$, then theorem 1-8 yields :

$$\exists v \in V(\sigma_j), \quad \exists u \in U(\sigma_j) : \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} x = v = \begin{bmatrix} v'_n \\ 0 \end{bmatrix}, \quad \sigma_j = \sigma'_n, \quad u = u'_n$$

$$B \text{ (and } \hat{B}) \perp u = u'_n \in U(\sigma_j) \text{ and } n+1 \leq j \leq n+d \tag{1.102}$$

Hence, B can only be approximated in the space generated by the left singular vectors $\{u_1, \ldots, u_{n+d}\}$ of $[A;B]$ which are orthogonal onto u'_n. This means that there is no TLS solution along v (see (1.102)). Hence, $[\hat{X}^T; -I_d]^T$ can only be written as a linear combination of the right singular vectors v_i of $[A;B]$ which are orthogonal onto v. The correction matrix, given by (1.89), has minimal norm given by :

$$\left\| [\Delta\hat{A}; \Delta\hat{B}] \right\|_F = \sqrt{\sum_{\substack{k=n \\ k \neq j}}^{n+d} \sigma_k^2} \tag{1.103}$$

- For this nongeneric multidimensional case, we can **compare the TLS and LS solution** : LS looks for a solution X' such that $\sum_{i=1}^{n} \sigma'_i u'_i v'^T_i X' = B'$. Under the conditions given in theorem 1-8 and 1-7 for $q = n-1$, B as well as B' are orthogonal onto u'_n, $\sigma'_n = \sigma_j$, $n+1 \leq j \leq n+d$. This means that LS and TLS look for a lower rank approximation of $[A_{n-1};B]$ with A_{n-1} the approximation of rank $n-1$ of A obtained by making σ'_n zero. Also, X' must be orthogonal onto v'_n and LS must look for a solution X' in the $(n-1)$-dimensional subspace $\{v'_1, \ldots, v'_{n-1}\}$ orthogonal onto v'_n. This solution is given by $X' = \sum_{i=1}^{n-1} \sigma'^{-1}_i v'_i u'^T_i B$. Thus B, as well as $\hat{B}$ and B', are orthogonal onto the same singular vector u'_n. TLS as well as LS search for an optimal solution in a subspace of the same dimension $n-1$, approximating $[A_{n-1};B]$. Hence :

$$\left\| B-B' \right\|_2 = \left\| [A_{n-1};B] - [A_{n-1};B'] \right\|_2 \geq \min_{\text{rank}(C)=n-1} \left\| [A_{n-1};B] - C \right\|_2 = \sigma_n = \left\| [\Delta\hat{A}; \Delta\hat{B}] \right\|_2 \tag{1.104}$$

and

$$\left\| B-B' \right\|_F = \left\| [A_{n-1};B] - [A_{n-1};B'] \right\|_F \geq \min_{\text{rank}(C)=n-1} \left\| [A_{n-1};B] - C \right\|_F = \left\| [\Delta\hat{A}; \Delta\hat{B}] \right\|_F \tag{1.105}$$

where $[\Delta\hat{A};\Delta\hat{B}]$ is given by (1.89) and its Frobenius norm by (1.103). As in the one-dimensional case, we observe that the generic TLS solution does not exist if (0.1) is highly incompatible (so that $\|B-B'\|_2 > \sigma'_n$), or if A is (nearly) rank-deficient (i.e. $\sigma'_n \approx 0$). Those situations seldom occur and we may conclude that **nearly always the TLS solution can be computed in the generic sense.**

- Consider now the **more general** case : **corank** $(V_{22}) = \kappa \geq 1$. We can always compute another basis of the right singular subspace of $[A;B]$ associated with Σ_2, such that :

$$\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} Q = \begin{bmatrix} X & Y \\ 0 & \Lambda \end{bmatrix} \begin{matrix} n \\ d \end{matrix} \qquad (1.106)$$

$\kappa \quad d-\kappa$

with Q an orthonormal matrix. From (1.106) it follows that the first κ columns $Q_1 = [q_1, \ldots, q_\kappa]$ of Q are a κ-dimensional orthonormal basis of the kernel of V_{22}. Let b (resp., r) be the minimal (resp., maximal) index of the first (resp., last) nonzero component over all base vectors $q_1, \ldots, q_\kappa$. Then if $\sigma_{n+b} = \ldots = \sigma_j = \ldots = \sigma_{n+r}$, theorem 1-8 can be applied for each vector $\in$ Ker (V_{22}). The involved singular vectors of $[A;B]$ and A, corresponding to each base vector q_i, i=1, ..., κ are the base vectors of a κ-dimensional subspace, associated with the same singular value σ_j. This is clear from (1.106). Indeed

$$\left(\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix}[q_1, \ldots, q_\kappa]\right)^T \left(\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix}[q_1, \ldots, q_\kappa]\right) = I_\kappa .$$

By a straightforward **extension** of theorem 1-8 to **κ-dimensional singular subspaces**, we obtain :

Theorem 1-8 bis :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., $[A;B]$). Let $\sigma_n > \sigma_{n+1}$. $V'(\sigma_j)$ (resp., $U'(\sigma_j)$) is the right (resp., left) singular subspace of A, associated with σ_j and $V(\sigma_j)$ (resp., $U(\sigma_j)$) the right (resp., left) singular subspace of $[A;B]$, associated with σ_j. Consider any orthonormal basis $K_{d\times\kappa}$ of Ker (V_{22}) with corank κ and let b (resp., r) be the minimal (resp., maximal) row index over all nonzero elements of K. Then the following relations can be proven :

If $\sigma_{n+b} = \ldots = \sigma_j = \ldots \sigma_{n+r}$ then

(a) $\exists V_\kappa \subset V(\sigma_j)$, $\exists V'_\kappa \subset V'(\sigma_j)$: $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} K = V_\kappa = \begin{bmatrix} V'_\kappa \\ 0 \end{bmatrix}$

(b) $\forall i \in \{n-\kappa+1, \ldots, n\} : \sigma_j = \sigma'_i$

(c) $\exists U'_\kappa \subset U'(\sigma_j) : B \perp U'_\kappa$

(d) $\exists U_\kappa \subset U(\sigma_j)$, $\exists U'_\kappa \subset U'(\sigma_j) : U_\kappa = U'_\kappa$

(e) $\exists U'_\kappa \subset U'(\sigma_j) : B' \perp U'_\kappa$

(f) $\exists U'_\kappa \subset U'(\sigma_j) : \hat{B} \perp U'_\kappa$ (1.107)

(V_κ and V'_κ are n+d by κ matrices. U_κ and U'_κ are m by κ matrices.)

Comments :

- Observe that the **corank** of V_{22} is given by the **multiplicity** κ of σ'_n. Its corresponding singular subspaces are given by U'_κ and V'_κ.
- The proof of theorem 1-8 bis follows from a straightforward extension of the proofs given in theorem 1-8 and are left to the reader. Instead of singular vectors, one has to deal with κ-dimensional singular subspaces associated with σ'_n of multiplicity κ.

- In these cases, the nongeneric TLS **solution** satisfying (1.18) and (1.19) is also deduced from theorem 1-7 with p = n. B, $\hat{B}$ as well as B' are orthogonal onto a κ-dimensional subspace U'_κ associated with σ'_n of multiplicity κ. TLS as well as LS search for an optimal solution in a subspace of same dimensionality $n-\kappa$, orthogonal onto U'_κ. The TLS solution $\hat{X}$ as well as the LS solution X' are orthogonal onto the corresponding right singular subspace V'_κ, associated with σ'_n.

- Let us now describe **the properties in case the minimum norm generic TLS solution**, as described in section 1.5, **does not exist**, i.e. $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$, p > 1 and Γ in (1.42) singular. The properties of this case follow from a straightforward extension of theorem 1-8.

Theorem 1-8 tris :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., [A;B]). Assume $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$ with $p \geq 1$. Let Q, Γ and Z be defined by (1.41) and assume that Γ is singular. Let $V'(\sigma_j)$ (resp., $U'(\sigma_j)$) be the right (resp., left) singular subspace of A, associated with σ_j and $V(\sigma_j)$ (resp., $U(\sigma_j)$) the right (resp., left) singular subspace of [A;B], associated with σ_j. Consider any unit vector $x \in \text{Ker}(\Gamma)$. Then the following relations can be proven.

If $\sigma_{n-p+1} = \ldots = \sigma_j = \ldots = \sigma_{n+d}$ then

(a) $\exists v \in V(\sigma_j)$; $\exists v' \in V'(\sigma_j)$: $\begin{bmatrix} Z \\ \Gamma \end{bmatrix} x = v = \begin{bmatrix} v' \\ 0 \end{bmatrix}$

(b) $\exists i : \sigma_j = \sigma'_i \qquad n-p-d+1 \leq i \leq n$

(c) $\exists u' \in U'(\sigma_j) : B \perp u'$

(d) $\exists u \in U(\sigma_j)$, $\exists u' \in U'(\sigma_j) : u = u'$

(e) $\exists u' \in U'(\sigma_j) : B' \perp u'$

(f) $\exists u' \in U'(\sigma_j) : \hat{B} \perp u'$ (1.108)

This theorem can be proven in the same way as theorem 1-8 and is hence left to the reader.

- If corank $(\Gamma) = \kappa > 1$, theorem 1-8 tris can be generalized in the same way as theorem 1-8 was generalized to theorem 1-8 bis.

General comments :

- The **additional conditions** may appear somehow **restrictive.** However, **solving** the set of equations **AX = B with TLS** makes **only sense when the data A and B are observations of an exact but unobservable relation** $A_o X = B_o$. Due to errors, the singular values $\sigma_{n+1}, \ldots, \sigma_{n+d}$ of [A;B] will differ from the exact singular values $\sigma^o_{n+1} = \ldots = \sigma^o_{n+d} = 0$ of $[A_o;B_o]$. Hence, it is advisable to consider $\sigma_{n+1}, \ldots, \sigma_{n+d}$ of [A;B] as **coinciding** and thus assume that the additional conditions of theorem 1-8 are indeed satisfied.

- If V_{22} (or Γ) is singular and the **additional conditions are not satisfied,** the nongeneric TLS solution still exists as proven in theorem 1-7. However, for those cases the nice properties described in theorem 1-8 are no longer valid and no adequate comparison with the LS solution can be made. This situation will occur if at least one subset $Ax_i = b_i$ in (0.1) is highly incompatible. It affects the accuracy of the other solutions when solving the d-dimensional TLS problem. In this case, it is expected to obtain more accurate TLS solutions $\hat{x}_i$, $i = 1, \ldots, d$, by solving d single one-dimensional TLS problems $Ax_i = b_i$, $i = 1, \ldots, d$.

1.7. THE TLS SOLUTION IN CASE OF EXACTLY KNOWN COLUMNS

In certain linear modelling problems AX = B, **some of the columns of A** may be **known exactly.** Hence, it is advisable to take this a priori information in account in order to maximize the accuracy of the solution. It is quite easy to determine the solution when there is no error in some of the columns of the data matrix [43]. The technique involves computing a **QR factorization of the "known" columns** and then, **solving a TLS problem of reduced dimension.**

Let a matrix $A = [A_1;A_2]$ be given whose first p columns A_1 have no error. Then, the algorithm is as follows. Perform p Householder transformations Q on the matrix [A;B] so that one has a new matrix

$$Q^T [A_1;A_2;B] = \begin{bmatrix} R_{11} & R_{12} & R_{1b} \\ 0 & R_{22} & R_{2b} \end{bmatrix} \begin{matrix} p \\ n+d-p \end{matrix} \qquad (1.109)$$

$$\qquad\quad p \quad n-p \quad d$$

where R_{11} is a p by p upper triangular matrix. We then compute the TLS solution $\hat{X}_2$, associated with $[R_{22};R_{2b}]$. $\hat{X}_2$ yields the last n-p components of each solution vector $\hat{x}_i$. To find the first p rows $\hat{X}_1$ of the solution matrix $\hat{X}$ we solve

$$R_{11} \hat{X}_1 = R_{1b} - R_{12} \hat{X}_2 \qquad (1.110)$$

This is simply the **LS solution** obtained by **projecting** the reduced observation vector $B - A_2 \hat{X}_2$ into the space $R(A_1)$ generated by the "known" columns of A.

Using a **generalization of the Eckart-Young** theorem, Golub and Stewart [46] proved that the procedure above, computing the generalized TLS estimate $[\hat{X}_1^T;\hat{X}_2^T]^T$, finds the best rank r approximation ($r \leq n$) $[A_1;\hat{A}_2;\hat{B}]$ to $[A;B]$ that leaves A_1 fixed such that

$$\left\|[A_1;A_2;B] - [A_1;\hat{A}_2;\hat{B}]\right\|_F = \min_{\text{rank}[A_1;\tilde{A}_2;\tilde{B}] \leq r} \left\|[A_1;A_2;B] - [A_1;\tilde{A}_2;\tilde{B}]\right\|_F \tag{1.111}$$

Geometrically, the technique implies that the range of the TLS approximation $[\hat{A}_2;\hat{B}]$ is made orthogonal onto the range $R(A_1)$ of the LS approximation. By **varying** p from zero to n, this technique could handle the ordinary least squares situation as well as every TLS problem.

1.8. OUTLINE OF THE EXTENDED TLS ALGORITHM

1.8.1. NO EXACTLY KNOWN COLUMNS

In this section, we want to summarize the practical TLS computation into a **generalized** algorithm which handles the generic TLS problem of any dimension d, as well as the case of non-uniqueness and the nongeneric case.

Algorithm 1-1 : Classical TLS computation of the TLS solution $\hat{X}$ of AX=B

Given : an m by n data matrix A and an m by d observation matrix B.

Step 1 :

1.a. : If $m > \frac{5}{3}(n+d)$, transform $[A;B]$ into upper triangular form R by Householder transformations.

1.b. : Compute the Singular Value Decomposition (0.3) of $[A;B]$ (or R).

Step 2 : If not user determined, compute the rank r ($\leq n$) of $[A;B]$ by :

$$\sigma_1^2 \geq \ldots \geq \sigma_r^2 > 2 \max(m,n+d)\,\sigma_v^2 \geq \sigma_{r+1}^2 \geq \ldots \geq \sigma_{n+d}^2 \tag{1.112}$$

with σ_v the standard deviation of the errors on the entries in $[A;B]$.

Step 3 : Compute a Householder matrix Q such that

$$[v_{r+1}, \ldots, v_{n+d}]\,Q = \begin{bmatrix} \times & Z \\ 0 & \Gamma \end{bmatrix} \begin{matrix} n \\ d \end{matrix} \qquad \begin{matrix} \\ n-r \end{matrix}\ \begin{matrix} \\ d \end{matrix} \tag{1.113}$$

with Γ a d by d upper triangular matrix.

Step 4 : If Γ singular then begin
lower the rank r with the multiplicity of σ_r
go back to step 3
end
else solve by forward elimination : $\hat{X}\,\Gamma = -Z$

(1.114)

END

Comments :

- A fully documented Fortran program of this classical TLS algorithm is given in [121].
- If $m < n$, the set of equations AX = B are always **underdetermined.** This implies that the solution, generically, is never unique. In this case, the algorithm 1-1 can still be applied to compute the minimum norm solution.
- If d = 1, the rank r = n, $|\Gamma| > \varepsilon > 0$ and $\sigma_n > \sigma_{n+1}$, (1.114) reduces to (1.16). The TLS solution is obtained from a simple scaling of the right singular vector v_{n+1} of $[A;b]$, associated with its minimal singular value σ_{n+1}.
- If $m > \frac{5}{3}(n+d)$, it is more efficient to first **triangularize** $[A;B]$, using a QR factorization [19] :

$$Q^T[A;B] = R = \begin{bmatrix} R_{11} & R_{21} \\ 0 & R_{22} \end{bmatrix} \begin{matrix} n \\ d \end{matrix} \qquad \begin{matrix} n & d \end{matrix} \tag{1.115}$$

with R, R_{11} and R_{22} upper triangular and Q orthogonal and then proceed with R. Indeed, $[A;B]$ and R have the same singular values and right singular vectors. Hence, the TLS solution $\hat{X}$ of AX = B can also be obtained from the SVD of R. This option has been incorporated into algorithm 1-1 (see [121]). More details are given in chapter 9.

- Observe from (1.113) that we **only need to compute a few singular vectors**, associated with the smallest singular values of [A;B] in order to obtain the TLS solution $\hat{X}$. Moreover, we **only need to compute a basis of the singular subspace corresponding to the smallest n+d-r singular values with r the rank of [A;B]**.
  Indeed, the TLS solution $\hat{X}$ is **invariant** with respect to orthogonal base transformations P in its solution space $\begin{bmatrix} \hat{X} \\ -I_d \end{bmatrix}$. With $\begin{bmatrix} Z \\ \Gamma \end{bmatrix}$ in (1.113) an orthonormal basis of the TLS solution space, we have :

$$\begin{bmatrix} Z \\ \Gamma \end{bmatrix} P = \begin{bmatrix} Z\ P \\ \Gamma\ P \end{bmatrix} \tag{1.116}$$

  Hence, the TLS solution $\hat{X} = -(ZP)(\Gamma P)^{-1} = -Z\,PP^T\Gamma^{-1} = -Z\,\Gamma^{-1}$ remains invariant.
  Based on these properties we were able to **improve the efficiency** of the TLS computations by **computing the SVD** of [A;B] in step 1 only **"partially"**. This results in the development of a **new algorithm** "partial total least squares (PTLS)". PTLS is about two times faster than the classical TLS computation given in Algorithm 1-1, while the same accuracy can be maintained. For more details, we refer the reader to chapter 9.

- With respect to **rank determination**, the SVD is the only fully reliable way to treat **rank-deficient** problems. As said before, the rank r is less than n in order to compute the TLS solution. The rank determination method [105-23] used in step 2, consists in approximating the given data [A;B] by an approximation $[\hat{A};\hat{B}]$ of **maximal minimal signal-to-noise ratio** (MMSNR) but living in a subspace of dimension r [105]. This dimension is chosen directly by the user (leading to a maximal MSNR for this dimension) or indirectly by specifying the standard deviation σ_v of the error on the entries of [A;B] and a required signal-to-noise ratio R_o. If

$$\sigma_1^2 \geq \ldots \geq \sigma_r^2 \geq \max(m,n+d)\,\sigma_v^2\,(R_o+1) > \sigma_{r+1}^2 \geq \ldots \geq \sigma_{n+d}^2 \tag{1.117}$$

  then the r-dimensional space $[\hat{A};\hat{B}]$, associated with $\sigma_1, \ldots, \sigma_r$, has an expected MSNR $R \geq R_o$ given by

$$R = (\sigma_r^2)_o \,/\, \max(m,n+d)\,\sigma_v^2 \tag{1.118}$$

  with $(\sigma_r)_o$ the r-th singular value of the exact but unobservable data $[A_o;B_o]$, measured by [A;B]. If the entries in [A;B] are corrupted by independently distributed errors with zero mean and variance σ_v^2, the expected value $E(\sigma_r^2)$ of the squared r-th singular value σ_r^2 of [A;B] is given by [23] :

$$E(\sigma_r^2) = (\sigma_r^2)_o + \max(m,n+d)\,\sigma_v^2 \tag{1.119}$$

  Substituting $(\sigma_r^2)_o$ in (1.118) by (1.119), we obtain from $R \geq R_o$:

$$\sigma_r^2 - \max(m,n+d)\,\sigma_v^2 \geq \max(m,n+d)\,\sigma_v^2\,R_o \tag{1.120}$$

  or

$$\sigma_r^2 \geq \max(m,n+d)\,(R_o+1)\,\sigma_v^2 \tag{1.121}$$

  By choosing $R_o = 1$, we obtain the rank determinator in (1.112).

- In step 3 of algorithm 1-1, the **multiplicity** of σ_r must be determined. One approach is to regard all computed singular values in the interval $[\sigma_r^2, \sigma_r^2 + \delta]$ as being identical. $\delta > 0$ may be specified by the user or automatically computed from (1.119). In the latter case δ is given by $\max(m,n+d)\,\sigma_v^2$ with σ_v the standard deviation of the error on each entry in the data [A;B].

- The upper triangular matrix Γ must be **tested for nonsingularity.** This is done by comparing the absolute value of each diagonal element $|\Gamma_{ii}|$, i = 1, ...d with a given small positive number ε. The choice of ε can be based on the numerical accuracy of the SVD computation [45] or, preferably, depend upon the perturbations of the data [133]. A generally applicable, reliable error bound ε in function of the standard deviation σ_v of the perturbations, can not be deduced. However the user can get more insight in the choice of ε from the sensitivity analysis given in chapter 3.

- As mentioned previously, we assume that the data [A;B] have been **properly scaled.** This is done by multiplying the rows and/or columns of the matrix [A;B] by nonzero scaling factors. This amounts to replacing [A;B] by $D_r\,[A;B]\,D_c$, where D_r and D_c are diagonal matrices consisting respectively of the row and column scaling factors. The mathematically equivalent scaled version of the linear system AX = B is

given by :

$$D_r\,[A;B]\,D_c\left(D_c^{-1}\begin{bmatrix} X \\ -I_d \end{bmatrix}\right) = 0 \qquad (1.122)$$

The **basic** recommendation is that **scaling** of equations and unknowns must proceed on a **problem-by-problem** basis. General scaling strategies are unreliable. It is best to scale on the basis of what the source problem proclaims about the significance of each entry in $[A;B]$. Measurement units and data error may have to be considered.
In order to attain the optimal TLS solution properties (see chapter 4), we must assume a simple but revealing error model in which the entries of the error matrix $[\Delta A;\Delta B]$ must be **independent** random variables with zero mean and **equal** standard deviation σ_v. This implies that the error covariance matrix is given by $\sigma_v^2 I_{n+d}$. The assumption of a common standard deviation has important implications for the scaling of $[A;B]$. From (1.122) it is evident that any scaling $D_r\,[A;B]\,D_c$ of $[A;B]$ automatically induces the same scaling $D_r\,[\Delta A;\Delta B]\,D_c$ of the error matrix $[\Delta A;\Delta B]$. Specifically, if the entries of $[\Delta A;\Delta B]$ are assumed to be independent, then the rows and columns of $[A;B]$ must be scaled so that all the elements of $[\Delta A;\Delta B]$ are roughly equal. The failure to observe this dictum can result in spurious indications of degeneracy, a phenomenon sometimes referred to as artificial ill-conditioning [110]. Consequently, the following **scaling strategy is recommended** : estimate the absolute size of the errors in the matrix $[A;B]$ and then scale so that as nearly as possible all the estimates are equal. It is important to realize that the errors to be estimated include all initial errors (measurement, computational, and rounding errors). In many applications rounding errors will be the least of these. If the error in each entry is proportional to the size of the element, the strategy reduces to scaling the original matrix $[A;B]$ so that all its elements are roughly equal. This is a frequently recommended strategy, but it is applicable only when the error matrix $[\Delta A;\Delta B]$ represents a uniform relative error in the entries of $[A;B]$. To obtain the proper scaling factors, **"equilibrium"** routines intended to scale a matrix so that its elements are roughly equal in size [12 -117], can be applied instead of the matrix of error estimates. It should be noted, however, that it is **not always possible to equilibrate** the matrix of error estimates by row and column scaling, in which case the optimal statistical properties of the TLS solution are not obtained (see chapter 4).

- If the **error covariance matrix is known** but not of the form $\sigma_v^2 I_{n+d}$, we can **transform** the data $[A;B]$ such that the covariance matrix of the transformed set is given by $\sigma_v^2 I_{n+d}$. This scaling method is described in chapter 4.

1.8.2. WITH EXACTLY KNOWN COLUMNS

If K columns of the m by n data matrix A are **known exactly**, the solution of (0.1) is obtained by computing the QR factorization of the "known" columns and solving a TLS problem of reduced dimension n-K using algorithm 1-1. The technique is summarized in the following algorithm.

Algorithm 1-2 : Classical TLS computation of the solution of AX = B, in case a subset A_1 of A is known exactly

Given an m by n data matrix with K exactly known columns, $0 \leq K \leq n$, an m by d observation matrix B and an n-dimensional vector t indicating whether the i-th column of A is known exactly ($t_i = 0$) or not ($t_i = 1$), i = 1, ..., n.

Step 1 : Permute the columns of A such that

$$AP = [A_1;A_2] \qquad (1.123)$$

where P is an n by n permutation matrix and A_1 an m by K matrix containing the K exactly known columns of A.

Step 2 : Compute the QR factorization of $[A_1;A_2;B]$ such that

$$Q^T\,[A_1;A_2;B] = \begin{bmatrix} R_{11} & R_{12} & R_{1b} \\ 0 & R_{22} & R_{2b} \end{bmatrix} \qquad (1.124)$$

where Q is orthogonal and R_{11} is an upper triangular K by K submatrix.

Step 3 : Compute the n-K by d solution $\hat{X}_2$ of $R_{22}\,\hat{X}_2 = R_{2b}$ with algorithm 1-1.

Step 4 : Compute the K by d solution $\hat{X}_1$ by back substitution, from :

$$R_{11}\,\hat{X}_1 = R_{1b} - R_{12}\,\hat{X}_2 \qquad (1.125)$$

Step 5 : Undo the permutations P to obtain the n by d solution $\hat{X}$ of AX = B :

$$\hat{X} = P\begin{bmatrix}\hat{X}_1\\ \hat{X}_2\end{bmatrix} \qquad (1.126)$$

END

Comments :

- If $m > \frac{5}{3}(n-K+d)$, it is more efficient to compute the complete QR factorization of $[A;B]$ such that R_{22} in (1.124) is also upper triangular [19].
- In computing the QR factorization of $[A_1;A_2;B]$, **pivoting** [48, section 6.4] may be done within three groups of columns : the first K columns A_1, the next n-K columns A_2 and the last d columns B. Columns may not be pivoted with columns from another group. Even for full rank problems, column pivoting seems to produce more accurate solutions [59]. If pivoting is done, the inverse permutations must be performed in the last step of algorithm 1-2.

1.9. CONCLUDING REMARKS

In this chapter the total least squares (TLS) problem and the algorithm to solve it are fully analyzed. As shown in the following chapters, TLS is useful in all linear problems AX = B where both the data matrix A and observation matrix B are inaccurate.

We first present the **main principle** and solution of the one-dimensional generic TLS problem with unique solution (section 1.2) and compare it with the ordinary least squares method from a **geometric** point of view (section 1.3) and with an extreme example. Furthermore, the generic one-dimensional TLS problem with unique solution is **extended** to all cases. It is shown how the **multidimensional** generic TLS solution is computed (section 1.4) and how TLS singles out the minimum norm solution if the solution of the problem is **no longer unique** (section 1.5). Moreover, we generalize the TLS computation to all **nongeneric** cases, i.e.

problems in which **the algorithm of Golub and Van Loan fails to produce a solution.** We describe **the properties** of those problems and prove that the proposed generalization is always **optimal** with respect to the TLS criteria in the one-dimensional (section 1.6.1) as well as in the multidimensional case (section 1.6.2). Furthermore, the TLS solution is presented for cases in which some of the columns of the data matrix A have no error (section 1.7). Finally, the TLS computation is **summarized** in one algorithm which includes all the extensions described above (section 1.8).

CHAPTER 2

PROPERTIES OF THE TLS PROBLEM AND THEIR RELATION TO THOSE OF THE LS PROBLEM

2.1. INTRODUCTION

In the previous chapter, the TLS method is introduced as a better more global fitting technique than the ordinary least squares (LS) technique for solving overdetermined sets of linear equations AX = B if errors occur in all data. While the ordinary LS confines all errors to the observation matrix B, TLS considers perturbations of both B and A and tries to compensate these. It is shown in chapter 1 how every possible problem can be solved with the TLS technique and the computational aspects are discussed. Finally, the TLS computation is summarized into one algorithm.

The main contribution of this chapter is a **direct analytical comparison** and evaluation of the TLS technique with respect to the ordinary LS method from different points of view. Hereto, new interesting **relationships** are now presented and many remarkable **properties** proven concerning **different aspects** of the problem. They clearly enlighten new equivalences and differences between both techniques.
First of all, **solution related** properties are described in section 2.2. Various relationships between both solutions are proven and different TLS related matrices are expressed in function of the solution, the original data and its SVD. In section 2.3, **residual** properties are discussed. While LS minimizes a sum of squared residuals $\|B - AX\|_F$ of the problem, it is shown how TLS in fact minimizes a sum of weighted squared residuals. Different relations describing the differences between both residuals are given and show the influencing parameters. Furthermore, the **approximation efforts** of TLS and LS to make the set of equations compatible, are compared analytically in section 2.4. Expressions are derived, relating the solution to the relative lengths of the corrections applied to each column of A and B. From the correction matrix of each technique, assumptions about the underlying perturbation model are deduced. Finally, the **approximate subspaces** into which the data A and B are projected to make the set of equations compatible are compared in section 2.5. The closeness of both subspaces is described by an appropriate distance function and useful expressions of this distance are derived for the one-dimensional problem.

All the properties proven in this chapter, analyze new aspects of the relationship between the TLS approach and the LS approach. They are **useful tools** in the study of the **sensitivity** of both techniques with respect to perturbations on the data (see chapter 3) and increase the reader's insight into the **practical significance** of the TLS technique for his own applications.

Most of the properties presented in this chapter are proven for the generic one-dimensional or multidimensional problem with unique solution. As these conditions are usually satisfied, these assumptions are not very restrictive. Moreover, many properties can be easily extended.

2.2. SOLUTION PROPERTIES

In this section, various new characterizations of the TLS solution and its relationship with the LS solution enlighten interesting properties of the TLS problem. A first **useful expression of the TLS solution** is given in the following theorem.

Theorem 2-1 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., [A;B]). If $\sigma'_n > \sigma_{n+1} = \ldots = \sigma_{n+d}$, the TLS solution $\hat{X}$ is given by

$$\hat{X} = (A^T A - \sigma^2_{n+1} I_d)^{-1} A^T B \qquad (2.1)$$

Proof :

From (1.17), we obtain

$$(A - \Delta\hat{A})\ \hat{X} = B - \Delta\hat{B} \qquad (2.2)$$

Multiply (2.2) with A^T from the left :

$$A^T A\hat{X} - A^T [\Delta\hat{A};\Delta\hat{B}] [\hat{X}^T;-I_d]^T = A^T B \tag{2.3}$$

Comparing (1.21) with (1.22), we observe that $A^T [\Delta\hat{A};\Delta\hat{B}] = \Delta\hat{A}^T [\Delta\hat{A};\Delta\hat{B}]$. Hence, using (1.22) and (1.25), the second term in (2.3) can be written as :

$$\Delta\hat{A}^T [\Delta\hat{A};\Delta\hat{B}] [\hat{X}^T;-I_d]^T = - V_{12} \Sigma_2^2 V_{22}^{-1} \tag{2.4}$$

The condition $\sigma_{n+1} = \ldots = \sigma_{n+d}$, implies that $\Sigma_2^2 = \sigma_{n+1}^2 I_d$. Hence, using (1.25), (2.4) equals $\sigma_{n+1}^2 \hat{X}$. Substituting this in (2.3) gives :

$$(A^T A - \sigma_{n+1}^2 I_d) \hat{X} = A^T B \tag{2.5}$$

Since $\sigma_n' > \sigma_{n+1}$, $A^T A - \sigma_{n+1}^2 I$ is nonsingular, hence its inverse exists and (2.5) implies (2.1). ■

Theorem 2-1 is a generalization of [47, theorem 4.1] and proven in a different way. Observe from (2.1) that $\hat{X}$ approaches X' as σ_{n+1} goes to zero. This means that **the TLS and LS solutions approach each other when the set of equations (0.1) becomes more compatible.** Since $\sigma_n' > \sigma_{n+1}$, we know from (1.34) that $\hat{X}$ in (2.1) is generic and is the unique solution of the TLS problem.

For nongeneric one-dimensional TLS problems, (2.1) can also be applied : just replace σ_{n+1} by the singular value σ_j corresponding to the $v_{n+1,j} \neq 0$ with largest index j, if $\sigma_{j-1}' > \sigma_j$.

The characterization (2.1) points out an **interesting connection between TLS and ridge regression.** Ridge regression is a way of regularizing the solution to an ill-conditioned LS problem (see chapter 5). It is easy to show that

$$x_{RR}(k) = (A^T A + kI)^{-1} A^T b \tag{2.6}$$

minimizes the function

$$\Phi(x;k) = \|b-Ax\|_2^2 + k\|x\|_2^2 \qquad \text{with } k \geq 0 \tag{2.7}$$

and that $\|x_{RR}(k)\|_2$ becomes small as k becomes large. This is the key to ridge regression : by controlling k one can control the 2-norm of $x_{RR}(k)$. For k = 0 in (2.6), one obtains the ordinary LS solution. What is particularly interesting, however, is that the TLS solution $\hat{x} = x_{RR}(-\sigma_{n+1}^2)$. That is, total least squares is a deregularizing procedure, a kind of "**reverse**" regression. As we shall see, this implies that the condition of the TLS problem is always worse than the condition of the corresponding LS problem. For this reason, it is interesting to compare the LS and TLS fits with one another.

From (2.1) and the dyadic decomposition (see (A.2)), we can easily derive an **exact expression for the difference between the LS solution X' and the TLS solution $\hat{X}$** which turns out to be quite useful in the evaluation of both techniques with respect to perturbations on the data (see chapter 3).

Corollary 2-2 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., [A;B]).

If $\sigma_n' > \sigma_{n+1} = \ldots = \sigma_{n+d}$, then

$$\hat{X} = \sum_{i=1}^{n} v_i' \sigma_i' (\sigma_i'^2 - \sigma_{n+1}^2)^{-1} u_i'^T B \tag{2.8}$$

$$\hat{X} - X' = \sum_{i=1}^{n} v_i' \sigma_{n+1}^2 \sigma_i'^{-1} (\sigma_i'^2 - \sigma_{n+1}^2)^{-1} u_i'^T B \tag{2.9}$$

$$\|\hat{x}\|_F > \|x'\|_F \tag{2.10}$$

Proof :

Using the SVD (0.2) of A, we obtain from (2.1) :

$$\hat{X} = V' (\Sigma'^2 - \sigma_{n+1}^2 I)^{-1} \Sigma'^T U'^T B \tag{2.11}$$

Using the dyadic decomposition, (2.11) yields (2.8).

On the other hand, we have :

$$X' = (A^T A)^{-1} A^T B = V' \Sigma'^{-1} U'^T B = \sum_{i=1}^{n} v_i' \sigma_i'^{-1} u_i'^T B \tag{2.12}$$

By subtracting (2.12) from (2.8), one obtains (2.9). Moreover

$$\|\hat{x}\|_F^2 = \sum_{i=1}^{n} \|u_i'^T B\|_2^2 \sigma_i'^2 (\sigma_i'^2 - \sigma_{n+1}^2)^{-2} \tag{2.13}$$

$$\|\hat{x}'\|_F^2 = \sum_{i=1}^{n} \|u_i'^T B\|_2^2 \sigma_i'^{-2} \tag{2.14}$$

Since each of the terms in the summation (2.13) is larger than the corresponsing term in the summation (2.14), we obtain (2.10). ■

Note in (2.8) that the TLS solution can be considered **as the LS solution of a slightly modified set of equations** $\tilde{A}X = B$. Matrix $\tilde{A}$ (whose singular values are given by $(\sigma_i'^2 - \sigma_{n+1}^2) / \sigma_i'$) only differs from A in its singular values. As σ_{n+1} goes to zero, $\tilde{A}$ approaches A and both solutions approach each other. Observe also from (2.9) that the difference between LS and TLS depends primarily on $\sigma_n'^2 - \sigma_{n+1}^2$. Interesting **upper bounds for the maximal difference between both solutions** are directly deduced from (2.9).

Corollary 2-3 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., [A;B]).
If $\sigma_n' > \sigma_{n+1} = \ldots = \sigma_{n+d}$, then

$$\|\hat{X} - X'\|_F \leq \sigma_{n+1}^2 \|X'\|_F (\sigma_n'^2 - \sigma_{n+1}^2)^{-1} \tag{2.15}$$

$$\leq \sigma_{n+1}^2 \|B\|_F \sigma_n'^{-1} (\sigma_n'^2 - \sigma_{n+1}^2)^{-1} \tag{2.16}$$

Proof :

(2.15) and (2.16) follow directly from (2.9) by taking norms and assuming $X' \,/\!/\, v_n'$. ■

Observe that the **upper bound is reached for** $B \,/\!/\, u_n'$ **or equivalently** $X' \,/\!/\, v_n'$ i.e. B is parallel with the singular vector of A associated with its smallest singular value. In this case, the solution is **most sensitive** to noise and the accuracy is most of all affected. This worst case is studied in chapter 3.
The bounds in Corollary 2-3 are large whenever σ_{n+1} is close to σ_n'. This occurs for example, whenever σ_{n+1} is a nearly repeated singular value. In this case the generic TLS solution is unstable, as suggested by (2.8). In other words, $\sigma_n'^2 - \sigma_{n+1}^2$ **is a measure of how close (0.1) is to the class of nongeneric TLS problems.**

Another interesting relation given in theorem 2-4, characterizes the **relationship between the TLS and LS solution.**

Theorem 2-4 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., [A;B]).

If $\sigma_n' > \sigma_{n+1} = \ldots = \sigma_{n+d}$, then

$$\hat{X} = [I - \sigma_{n+1}^2 (A^T A)^{-1}]^{-1} X' \tag{2.17}$$

$$= [I + \sigma_{n+1}^2 (A^T A - \sigma_{n+1}^2 I)^{-1}] X' \tag{2.18}$$

Proof :

Since $X' = (A^T A)^{-1} A^T B$ we have from (2.1) : (2.19)

$$\hat{X} = (A^T A - \sigma_{n+1}^2 I)^{-1} A^T A X' \tag{2.20}$$

Using [95, lemma 30], (2.20) yields (2.17) and (2.18). ■

Observe again that both solutions approach each other when σ_{n+1} goes to zero.

Since V is orthogonal, we can use the interesting **C-S decomposition** [48, theorem 2.4-1], i.e.

C-S decomposition :

If $V = \begin{matrix} \\ n \\ d \end{matrix} \overset{\begin{matrix} n & d \end{matrix}}{\begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix}}$ is orthogonal and $n \geq d$,

there exist orthogonal matrices W_1, $T_1 \in \mathbb{R}^{n \times n}$ and orthogonal matrices W_2, $T_2 \in \mathbb{R}^{d \times d}$ such that

$$\begin{bmatrix} W_1 & 0 \\ 0 & W_2 \end{bmatrix}^T \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix} = \begin{matrix} \\ n-d \\ d \\ d \end{matrix} \overset{\begin{matrix} n-d & d & d \end{matrix}}{\begin{bmatrix} I & 0 & 0 \\ 0 & C & S \\ 0 & -S & C \end{bmatrix}} \tag{2.21}$$

where $C = \text{diag}(c_1, \ldots, c_d)$ and $S = \text{diag}(s_1, \ldots, s_d)$ and $c_i = \cos\theta_i$, $s_i = \sin\theta_i$, $c_i^2 + s_i^2 = 1 \quad \forall i = 1, \ldots, d$ and $0 \leq \theta_1 \leq \ldots \leq \theta_d \leq \pi/2$.
The assumption $n \geq d$ is made for notational convenience only.
The C-S decomposition allows to derive exact expressions for the **2-norm and Frobenius norm of the TLS solution** $\hat{X}$ and to deduce interesting **properties of the TLS related submatrices of V.**

Theorem 2-5 :

Let (0.3) be the SVD of [A;B].

If $\sigma_n > \sigma_{n+1}$ and if V_{22} is nonsingular with singular values γ_i, i = 1, ..., d, then

$$\|\hat{x}\|_2^2 = (1 - \gamma_d^2)\ \gamma_d^{-2} \tag{2.22}$$

$$\|\hat{x}\|_F^2 = \sum_{i=1}^{d} (1 - \gamma_i^2)\ \gamma_i^{-2} = \|V_{22}^{-1}\|_F^2 - d \tag{2.23}$$

Proof :

With $\hat{X}$ given by (1.25) and since the 2-norm is invariant with respect to orthogonal transformations [48, section 2.2], we obtain using the C-S decomposition (2.21) of V (partitioned as in (0.3)) :

$$\|\hat{x}\|_2^2 = \|V_{12}\ V_{22}^{-1}\|_2^2 = \|W_1\ V_{12}\ T_2\ T_2^T\ V_{22}^{-1}\ W_2^T\|_2^2 = \|SC^{-1}\|_2^2 \tag{2.24}$$

Since $c_1 \geq \ldots \geq c_d$ and $s_1 \leq \ldots \leq s_d$, (2.24) yields :

$$\|SC^{-1}\|_2^2 = s_d^2 / c_d^2 = (1 - c_d^2) / c_d^2 \tag{2.25}$$

Since V_{22} and C have the same singular values, (2.22) follows directly from (2.25). The Frobenius norm is also invariant with respect to orthogonal transformations. Hence, we derive analogously as in (2.24) :

$$\|\hat{x}\|_F^2 = \|SC^{-1}\|_F^2 = \sum_{i=1}^{d} (1 - c_i^2) / c_i^2 = \sum_{i=1}^{d} (1/c_i^2) - d \tag{2.26}$$

Since C and V_{22} have the same singular values and c_i^{-1}, i = 1, ..., d are the singular values of V_{22}^{-1}, (2.23) follows directly from (2.26). ■

Theorem 2-5 can be straightforwardly extended to problems for which the TLS solution is no longer unique. If $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$ with $p \geq 1$, we just replace V_{22} by Γ defined in (1.41), to obtain the 2-norm and Frobenius norm of the TLS minimum norm solution from (2.22) and (2.23).

From the C-S decomposition (2.21) we also deduce directly the following interesting **properties of the submatrices** V_{11}, V_{22}, V_{12}, V_{22} **of V.**

- The singular values of V_{11} are $\{\underbrace{1, \ldots, 1}_{n-d}, \cos\theta_1, \ldots, \cos\theta_d\}$ (2.27)

- The singular values of V_{22} are $\{\cos\theta_1, \ldots, \cos\theta_d\}$ (2.28)

- The singular values of V_{12} and V_{21} are $\{\sin\theta_d, \ldots, \sin\theta_1\}$ (2.29)

It is evident from (2.27) and (2.28) that V_{11} and V_{22} have the same minimal singular value, or

$$\|V_{11}^{-1}\|_2 = \|V_{22}^{-1}\|_2 \tag{2.30}$$

Hence, the (non) singularity of V_{11} implies that of V_{22}.
Using (2.22) with γ_d the minimal singular value of V_{22}, we also have :

$$\gamma_d = (1 + \|\hat{x}\|_2^2)^{-\frac{1}{2}} = \|V_{12}\|_2 \ / \ \|\hat{x}\|_2 \tag{2.31}$$

From (2.27) and (2.31) one can easily deduce the **condition of the TLS problem** which **measures the worst-case sensitivity** of the TLS problem. Since the TLS solution is also obtained from (I.27) i.e. by solving

$$V_{11}^T\ \hat{x} = V_{21}^T \tag{2.32}$$

it follows from (2.27) that the condition number (defined in section 3.2.1) of the TLS problem is given by $1 / \cos\theta_d$. From (2.27) and (2.31) with γ_d the minimal singular value of V_{22}, we obtain :

$$\text{cond (TLS)} = 1 / \gamma_d = \sqrt{1 + \|\hat{x}\|_2^2} = \|\hat{x}\|_2 \ / \ \|V_{12}\|_2 \tag{2.33}$$

From (2.31) and (2.32) one observes that for V_{22} nearly singular, $\|\hat{x}\|_2$ must go to infinity and the generic TLS problem is badly conditioned. We refer the reader to chapter 3 for a more detailed comparison of the sensitivity of the TLS problem versus that of the LS problem.
Using [48, theorem 12.2-1] and (2.31), the following **bounds** for the **smallest singular value** σ_n' **of A** are obtained :

$$\sigma_n \ / \ \|V_{11}^{-1}\|_2 = \sigma_n / \sqrt{1 + \|\hat{x}\|_2^2} \leq \sigma_n' \leq \sigma_n \tag{2.34}$$

Observe that $\sigma_n' \to \sigma_n$ as $\|\hat{x}\|_2 \to 0$. This happens if $\|B\|_F \ll \|A\|_F$. Geometrically, this implies that TLS forces B to move towards A. In this case, the TLS solution $\hat{X}$ comes close to the LS solution X' where B is completely bent towards A (see section 2.4).
For the one-dimensional case (d=1), V_{22} is a scalar $v_{n+1,n+1}$ and we obtain from (2.28) and (2.31) :

$$v_{n+1,n+1} = \cos\theta = (1 + \hat{x}^T\ \hat{x})^{-\frac{1}{2}} \tag{2.35}$$

$$\|V_{12}\|_2 = \|V_{21}\|_2 = \sin\theta \tag{2.36}$$

and thus, $\|\hat{x}\|_2 = \tan\theta$. (2.37)

Furthermore some other **relationships between the TLS related matrices and the TLS solution** are deduced, which are quite useful in the derivation of the asymptotic properties of the TLS problem (see chapter 4).

Theorem 2-6 :

Let (0.3) be the SVD of $[A;B]$. If $\sigma_n > \sigma_{n+1}$ and V_{22} nonsingular, then

$$V_{11}\,V_{11}^T = (I + \hat{X}\,\hat{X}^T)^{-1} \tag{2.38}$$

$$V_{22}\,V_{22}^T = (I + \hat{X}^T\,\hat{X})^{-1} \tag{2.39}$$

$$V_{12}\,V_{12}^T = \hat{X}\,(I + \hat{X}^T\,\hat{X})^{-1}\,\hat{X}^T \tag{2.40}$$

Proof :

From (1.27) and the orthonormality of V, we obtain :

$$I + \hat{X}\,\hat{X}^T = [I;\hat{X}]\begin{bmatrix} I \\ \hat{X}^T \end{bmatrix} = (V_{11}^{-1})^T\,[V_{11}^T;V_{21}^T]\begin{bmatrix} V_{11} \\ V_{21} \end{bmatrix} V_{11}^{-1} = (V_{11}\,V_{11}^{-1})^T \tag{2.41}$$

which implies (2.38). From (1.25) and the orthonormality of V, we obtain :

$$I + \hat{X}^T\,\hat{X} = [\hat{X}^T;I]\begin{bmatrix} \hat{X} \\ I \end{bmatrix} = (V_{22}^{-1})^T\,[V_{12}^T;V_{22}^T]\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} V_{22}^{-1} = (V_{22}\,V_{22}^{-1})^T \tag{2.42}$$

which proves (2.39). From (1.25) and (2.39), (2.40) follows immediately. ■

If $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$ with $p \geq 1$, we can analogously prove that

$$\Gamma\,\Gamma^T = (I + \hat{X}^T\,\hat{X})^{-1} \tag{2.43}$$

with Γ nonsingular and given by (1.41).

Using (2.38), the **relationship between the data matrix A and its TLS approximation $\hat{A}$** is easily deduced.

Corollary 2-7 :

Let (0.3) be the SVD of $[A;B]$.

If $\sigma_n > \sigma_{n+1}$ and V_{22} nonsingular, then

$$\hat{A} = (A + B\,\hat{X}^T)\,(I + \hat{X}\,\hat{X}^T)^{-1} \tag{2.44}$$

Proof :

From (1.21), we obtain $\hat{A} = U_1\,\Sigma_1\,V_{11}^T$ (2.45)

One can easily prove that (2.45) also equals :

$$\hat{A} = [A;B]\,[V_{11}^T;V_{21}^T]^T\,V_{11}^T \tag{2.46}$$

and thus :

$$\hat{A} = (A + B\,V_{21}\,V_{11}^{-1})\,V_{11}\,V_{11}^T \tag{2.47}$$

Substituting (1.27) and (2.38) into (2.47) gives (2.44). ■

(2.44) suggests that the **TLS approximation $\hat{A}$ may not be considered as an estimate of the exact but unobservable data matrix A_o**, observed by A. This is proven in chapter 4 (see also section 2.4 for more details).

2.3. RESIDUAL PROPERTIES

In this section, the **minimization techniques** of TLS and LS are compared. While LS minimizes the sum of squared residuals of $Ax = b$, it is shown here that **TLS in fact minimizes a sum of weighted squared residuals** of $Ax = b$. The results derived from [41], are generalized for the multidimensional problem (0.1). Furthermore, various new relationships between LS and TLS residuals are presented and upper bounds are given.

Consider the model

$$A = A_o + \Delta A$$

$$B = B_o + \Delta B = A_o X + \Delta B \tag{2.48}$$

with A_o and B_o the exact but unobservable data matrix and observation matrix respectively and X the exact solution. $[\Delta A;\Delta B]$ is a random error

matrix whose rows are independently and identically distributed with common zero mean and common covariance matrix $\Sigma = \sigma^2 I_{n+d}$.

The estimation of the parameters X, A_o and σ^2 of (2.48) has received much attention in the literature (see chapter 4) and various approaches have been suggested. Let us consider the **generalized least squares estimation (GLSE)** approach. The GLSE approach makes no assumption about the distribution of $[\Delta A;\Delta B]$ beyond those made above. Since under the model (2.48) the residual matrix B - AX has zero mean and common covariance matrix $\sigma^2 (I_d + X^T X)$ [41], the GLS estimates of X are chosen to minimize some scalar function f of the **normalized** "error" or "**residual**" **matrix**, defined as :

$$N(X) = (B - AX)(I + X^T X)^{-\frac{1}{2}} \tag{2.49}$$

where $(I + X^T X)^{\frac{1}{2}}$ is any square root of $I + X^T X$.
In the case d = 1, (2.49) is an m-dimensional vector and minimization of the length of (2.49) yields the generalized LS estimator of Sprent [104]. This estimator is identical to the TLS solution of Ax = b. Indeed, the scalar function f is defined as :

$$f(N(x)) = \left\| [A;b]\,[x^T;-1]^T \right\|_2 / \left\| [x^T;-1]^T \right\|_2 \tag{2.50}$$

If the SVD of [A;b] is given by (0.3), it is easy to verify that $f(N(x)) \geq \sigma_{n+1}$. Equality holds if and only if $[x^T;-1]^T$ is in the right singular subspace associated with σ_{n+1}. Combining this fact with (1.16) or (1.37), we see that the generic TLS problem amounts to finding an $\hat{x} \in \mathbb{R}^n$ such that (2.50) is minimized and thus equals σ_{n+1}.
When d > 1, many scalar functions f of (2.49) could be used. Gleser and Watson [42] minimized the Frobenius norm of (2.49). It is easy to show that their GLS estimator, i.e. the value of X minimizing the Frobenius norm of (2.49), is identical with our TLS solution (1.25) or (1.42) of the problem (0.1). We can thus conclude that the generic **TLS solution $\hat{X}$ amounts to minimize the Frobenius norm** of the normalized residual matrix (2.49). This result is generalized in the next theorem.

Theorem 2-8 :

Let (0.3) be the SVD of [A;B]. If V_{22} nonsingular then :
The TLS solution $\hat{X}$ of (0.1) minimizes any orthogonally invariant norm of the normalized "error" or "residual" matrix :

$$N(X) = (B - AX)(I + X^T X)^{-\frac{1}{2}}$$

where $(I + X^T X)^{\frac{1}{2}}$ is any square root of $I + X^T X$.

The proof is given in [41, theorem 2.1].

In addition to the GLSE approach, an **approach** analogous to ordinary LS could be used, based on **minimizing some scalar function of a "total error" matrix**, defined as :

$$F(X, A_o) = [A-A_o;B-A_oX] \tag{2.51}$$

over X, A_o. When based on minimizing the **Frobenius norm** of (2.51), we obtain **the TLS minimization function** (1.19). This result is generalized in the next theorem.

Theorem 2-9 :

Assume the model (2.48). Let (0.3) be the SVD of [A;B]. If V_{22} nonsingular, then :
The TLS solution $\hat{X}$ of (0.1) and the TLS approximation $\hat{A}$, given by (1.25) and (1.21) respectively, minimize any orthogonally invariant norm of the "total error" matrix

$$F(X, A_o) = [A-A_o;B-A_oX]$$

over X and A_o.

The proof is given in [41, theorem 2.1].

Another approach to estimating X comes from noting that $B - A\hat{X}$ and $B\hat{X}^T + A$ are **uncorrelated.** This suggests choosing X so that

$$(B - AX)^T (BX^T + A) = 0 \tag{2.52}$$

Thus the **TLS solution $\hat{X}$ makes its residual $B - A\hat{X}$ and $B\hat{X}^T + A$ uncorrelated.**

We now prove some new expressions of the TLS residual $\hat{R}$ defined by $B - A\hat{X}$, and describe its relationship with respect to the LS residual $R' = B - AX'$.

Theorem 2-10 :

Let $\hat{X}$ be the TLS solution of (0.1) and $[\Delta\hat{A};\Delta\hat{B}]$ its corresponding TLS correction matrix satisfying (1.18) and (1.19), then :

$$\hat{R} = \Delta\hat{B}\ (\hat{X}^T\ \hat{X} + I) \tag{2.53}$$

$$= \Delta\hat{B} - \Delta\hat{A}\ \hat{X} \tag{2.54}$$

Proof :

From the relation $(A - \Delta\hat{A})\ \hat{X} = B - \Delta\hat{B}$ given by (1.17) and (1.19), (2.54) follows immediately. From (1.43), we obtain :

$$\Delta\hat{B} = [A;B] \begin{bmatrix} Z \\ \Gamma \end{bmatrix} \Gamma^T \tag{2.55}$$

Multiplying (2.55) with $(\hat{X}^T\ \hat{X} + I)$ from the right and substituting (2.43) and (1.42) into it, gives $\hat{R}$ in (2.53). ■

Observe from (2.53) that the TLS **residual is orthogonal onto the TLS approximate subspace** $\hat{R}([\hat{A};\hat{B}])$ and thus $\hat{R} \subset R\ ([\Delta\hat{A};\Delta\hat{B}])$.
Using (2.39), we immediately deduce the next result :

Corollary 2-11 :

Let (0.3) be the SVD of $[A;B]$. If $\sigma_n > \sigma_{n+1} = \ldots = \sigma_{n+d}$ and V_{22} nonsingular, then :

$$\hat{R}^T\ \hat{R} = (\hat{X}^T\ \hat{X} + I)\ \sigma_{n+1}^2 \tag{2.56}$$

Proof :

From (2.53) we obtain that

$$\hat{R}^T\ \hat{R} = (\hat{X}^T\ \hat{X} + I)\ \ \Delta\hat{B}^T\ \Delta\hat{B}\ (\hat{X}^T\ \hat{X} + I) \tag{2.57}$$

Using the assumptions above and (1.22), (2.57) equals
$(\hat{X}^T\ \hat{X} + I)\ \sigma_{n+1}^2\ V_{22}\ V_{22}^T\ (\hat{X}^T\ \hat{X} + I)$.

Substituting (2.39) into it, gives (2.56). ■

By taking the square root of the trace of (2.56), the **length of the TLS residual** follows immediately. For the one-dimensional case, we have :

$$\|\hat{r}\|_2 = \|b - A\hat{x}\|_2 = \sigma_{n+1}\sqrt{\|\hat{x}\|_2^2 + 1} \tag{2.58}$$

Other expressions for $\|\hat{r}\|_2$ have been proven in [122] :

$$\|\hat{r}\|_2 = \|\Delta b\|_2\ (\|\hat{x}\|_2^2 + 1) = \|\hat{x}\|_2\ \|\Delta\hat{A}\|_F + \|\Delta b\|_2 \tag{2.59}$$

(2.59) shows that $\|\hat{r}\|_2$ is precisely composed of the minimized TLS ingredients $\|\hat{x}\|_2$, $\|\Delta\hat{A}\|_F$ and $\|\Delta b\|_2$.

Several **relationships** between the TLS and LS **residual** and **upper bounds** have also been given by Golub and Van Loan [47]. These results can be straightforwardly extended to the multidimensional case when Σ_2 in (0.3) equals $\sigma_{n+1}\ I_d$. However, in most cases these upper bounds are too crude, e.g. when $\|B\|_2 >> \sigma_{n+1}$, and are never attained (unless $B \perp R(A)$). Therefore we present in the next theorem more realistic bounds, derived from expressions which describe the **exact difference between the LS and TLS residual.**

Theorem 2-12 :

Let (0.2) (resp., 0.3)) be the SVD of A (resp., $[A;B]$).
If $\sigma'_n > \sigma_{n+1} = \ldots = \sigma_{n+d}$, then

$$\hat{R} - R' = \sigma_{n+1}^2\ A\ (A^T\ A - \sigma_{n+1}^2\ I)^{-1}\ X' \tag{2.60}$$

$$= \sum_{i=1}^{n} u'_i\ \sigma_{n+1}^2\ (\sigma_i'^2 - \sigma_{n+1}^2)^{-1}\ u_i'^T\ B \tag{2.61}$$

$$\|\hat{R} - R'\|_F \leq \sigma_{n+1}^2\ \|B\|_F\ (\sigma_n'^2 - \sigma_{n+1}^2)^{-1} \tag{2.62}$$

$$\leq \sigma_{n+1}^2\ \|X'\|_F\ \sigma_n'^{-1}\ (\sigma_n'^2 - \sigma_{n+1}^2)^{-1} \tag{2.63}$$

$$\|\hat{R}\|_F \geq \|R'\|_F \tag{2.64}$$

Proof :

Since $\hat{R} - R' = A\ (\hat{X} - X')$ (2.65)

(2.60) follows directly by multiplying (2.18) with A from the left.
Substituting (2.9) into (2.65) gives :

$$\hat{R} - R' = \sum_{i=1}^{n} A\ v'_i\ \sigma_{n+1}^2\ \sigma_i'^{-1}\ (\sigma_i'^2 - \sigma_{n+1}^2)^{-1}\ u_i'^T\ B \tag{2.66}$$

Since $A\ v'_i = \sigma'_i\ u'_i$ from (0.2), (2.66) yields (2.61).

Taking the norm of (2.61) :

$$\|\hat{R} - R'\|_F^2 = \sum_{i=1}^{n} \|u_i'^T B\|_2^2 \; \sigma_{n+1}^4 \; (\sigma_i'^2 - \sigma_{n+1}^2)^{-2} \qquad (2.67)$$

we observe that (2.67) is maximized if $B /\!/ u_n'$. In this case, $\|u_i'^T B\|_2 = 0$ $\forall i \neq n$ and $\|u_n'^T B\|_2 = \|B\|_F$. Substituting this in (2.67) gives (2.62). $B /\!/ u_n'$ is equivalent with $X' /\!/ v_n'$ and thus, $\|B\|_F = \|X'\|_F / \sigma_n'$. Substituting this in (2.62) gives (2.63).
Since LS always minimizes any orthogonally invariant norm of its residual, (2.64) follows directly. ■

Observe from (2.62) and (2.63) that **the TLS and LS residuals approach each other**, if :

- σ_{n+1} is small, i.e. the set of equations (0.1) is only slightly incompatible.
- $\|B\|_F$ is small, i.e. the TLS solution becomes close to the LS solution.
- $\sigma_n' >> \sigma_{n+1}$, i.e. A may not be (nearly) rank-deficient.

2.4. APPROXIMATION EFFORT PROPERTIES

While LS minimizes the Frobenius norm of the residual matrix of (0.1) TLS minimizes the approximation effort given by (1.19). In the previous section, we presented the residual relationships. In this section, new theorems are given which analyze the approximation efforts, i.e. the corrections made by TLS and LS to the problem (0.1), in order to make the set compatible. We also deduce the assumptions about the underlying perturbation model. Furthermore, the relative TLS approximation efforts on A with respect to B and with respect to LS, are evaluated.

Our first theorem shows **the relation between the solution $\hat{X}$ and the applied corrections $\Delta\hat{A}$ and $\Delta\hat{B}$ on the data.** From this, the underlying TLS perturbation model can be deduced.

Theorem 2-13 :

Let $\hat{X}$ be a TLS solution of (0.1) with corresponding correction matrix $[\Delta\hat{A};\Delta\hat{B}]$ satisfying (1.18) and (1.19), then :

$$\Delta\hat{B}\,\hat{X}^T = -\Delta\hat{A} \qquad (2.68)$$

Proof :

Let (0.3) be the SVD of $[A;B]$. From (1.43), we obtain :

$$-\Delta\hat{A} = -[A;B]\begin{bmatrix} Z \\ \Gamma \end{bmatrix} Z^T \qquad (2.69)$$

$$\Delta\hat{B} = [A;B]\begin{bmatrix} Z \\ \Gamma \end{bmatrix} \Gamma^T \qquad (2.70)$$

Multiplying (2.70) with $\hat{X}^T$ and substituting (1.42) into it, gives (2.69), which proves (2.68). ■

Observe that for the **one-dimensional** case, (2.68) reduces to :

$$\hat{x}_i \, \Delta\hat{b} = -\Delta\hat{a}_i \qquad i = 1, \ldots, n \qquad (2.71)$$

where $\Delta\hat{a}_i$ is the i-th column of $\Delta\hat{A}$. (2.71) implies that **the i-th component $\hat{x}_i$ of the solution $\hat{x}$ is given by the ratio $\|\Delta\hat{a}_i\|_2 / \|\Delta\hat{b}\|_2$ of the length of the correction vectors $\Delta\hat{a}_i$ and $\Delta\hat{b}$.** Looking at the **geometric** representation of $Ax = b$ in the (n+1)-dimensional column space $R[\hat{A};\hat{b}]$ (fig. 1.2), this means that each component $\hat{x}_i$ of the solution $\hat{x}$ can be deduced by dividing the length of each $\Delta\hat{a}_i$ by $\|\Delta\hat{b}\|_2$.

Strictly spoken, we conclude from (2.68) that TLS assumes a rank-d **perturbation model** $[\Delta\hat{A};\Delta\hat{B}]$ (a rank-1 model in the one-dimensional case), given by (1.43). This seems to be very unrealistic [25]. Indeed, mostly the noise vectors added to the data, are independently and identically distributed, and thus correspond to a full rank perturbation model. However, it is easily proven that every matrix $[A;B]$, whose SVD is given by (0.3) and perturbed by noise $[\Delta A;\Delta B]$ satisfying (assume $\sigma_n > \sigma_{n+1}$ and V_{22} nonsingular for simplicity) :

$$[\Delta A;\Delta B]^T [\Delta A;\Delta B] = \sum_{i=1}^{d} \sigma_{n+i}^2 \, v_{n+i} \, v_{n+i}^T + \sum_{i=1}^{n} \alpha_i \, h_i \, h_i^T$$

$$\text{with } h_i \in \text{span}\{v_1, \ldots, v_n\} \qquad (2.72)$$

produces the exact solution when solved with TLS. In this case, noise is completely rejected. TLS **makes only assumptions about the noise distribution in the right singular subspace associated with the d smallest singular values** $\sigma_{n+1}, \ldots, \sigma_{n+d}$ of $[A;B] = [A_o;B_o] + [\Delta A;\Delta B]$.

In other words, noise can be completely rejected (and thus the TLS solution equals the exact solution of the exact but unobservable set $A_o X = B_o$), if the kernel of $[A-\Delta A;B-\Delta B] = [A_o;B_o]$ equals the kernel of the TLS approximation $[\hat{A};\hat{B}]$. If $\sigma_n > \sigma_{n+1}$ and V_{22} nonsingular, a basis of this kernel is given by $\begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix}$.

Those assumptions are more and more satisfied when the set of equations becomes more overdetermined (i.e. $m \to \infty$), and the rows of the data are perturbed by independently and identically distributed errors with zero mean and common covariance matrix $\sigma^2 I_{n+d}$ (see chapter 4).
We thus warn the user that even if the TLS solution $\hat{X}$ of (0.1) equals the solution X of the exact but unobservable set $A_o X = B_o$, its corresponding **correction matrix $[\Delta\hat{A};\Delta\hat{B}]$ can differ from the real perturbation matrix** $[\Delta A;\Delta B] = [A-A_o;B-B_o]$. Hence, $[\Delta\hat{A};\Delta\hat{B}]$ may not be considered as the expected real perturbation model of the data $[A;B]$. In fact, any perturbation model satisfying (2.72) yields the exact solution of $A_o X = B_o$ when (0.1) is solved with TLS. For example, the white noise approach [7] assumes in the one-dimensional case the underlying perturbation model $[\Delta\tilde{A};\Delta\tilde{b}]$, satisfying :

$$[\Delta\tilde{A};\Delta\tilde{b}]^T [\Delta\tilde{A};\Delta\tilde{b}] = \sigma_{n+1}^2 \sum_{i=1}^{n+1} v_i v_i^T \qquad (2.73)$$

Subtracting $[\Delta\tilde{A};\Delta\tilde{b}]$ from the data $[A;b]$ produces an approximate matrix $[A-\Delta\tilde{A};b-\Delta\tilde{b}]$ with zero minimal singular value. Hence, the solution equals the TLS solution (1.16). The only difference is that the condition (1.19) is no longer satisfied. For the white noise approach, $\|[\Delta\tilde{A};\Delta\tilde{b}]\|_F = \sqrt{n}\,\sigma_{n+1} > \sigma_{n+1} = \|[\Delta\hat{A};\Delta\hat{b}]\|_F$. Only the TLS correction matrix minimizes (1.19). We can thus conclude that **many perturbation models** (every model satisfying (2.72)) **correspond with the TLS solution**. If we **single out the model with minimizing norm**, then we obtain **the TLS correction matrix.**
Analogously, it can be seen that **LS assumes a perturbation model** of the form $[\Delta A;\Delta B] = [0;\Delta B']$. For LS, however, only perturbations $[0;\Delta B]$ of $[A_o;B_o]$ satisfying $\Delta B \perp R(A)$, correspond with the LS solution X' of (0.1). X' equals the exact solution of $A_o X = B_o$ if and only if $A = A_o$ and the projection $A_o A_o^T B$ of B onto $R(A_o)$ equals B_o.

From chapter 1, it is clear that the **approximation effort** to make the set of equations (0.1) compatible **is smaller for TLS** than for LS. It is

possible now **to express exactly the superiority** of the TLS approximation $[\hat{A};\hat{B}]$ with respect to the LS approximation $[A;B']$ by the following theorem.

Theorem 2-14 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., $[A;B]$).
If $\sigma_n > \sigma_{n+1}$ and V_{22} nonsingular, then :

$$\prod_{i=1}^{d} \sigma''^2_i \,([\Delta\hat{A};\Delta\hat{B}]) = \prod_{i=1}^{n} (\sigma'_i / \sigma_i)^2 \;\det(\Delta B'^T \Delta B') \qquad (2.74)$$

Proof :

If σ_i (resp., σ'_i) are the singular values of $[A;B]$ (resp., A), then σ_i^2 (resp., σ'^2_i) are the eigenvalues of $[A;B]^T [A;B]$ (resp., $A^T A$).

Hence, $$\det(A^T A) = \prod_{i=1}^{n} \sigma'^2_i \qquad (2.75)$$

and $$\det([A;B]^T [A;B]) = \prod_{i=1}^{n+d} \sigma_i^2 \qquad (2.76)$$

We also have [38]

$$\det([A;B]^T [A;B]) = \det \begin{bmatrix} A^T A & A^T B \\ B^T A & B^T B \end{bmatrix}$$

$$= \det(A^T A)\, \det(B^T B - B^T A\, (A^T A)^{-1} A^T B)$$

$$= \det(A^T A)\, \det(B^T B - B^T A\, X')$$

$$= \det(A^T A)\, \det(B^T B - B'^T B')$$

$$= \det(A^T A)\, \det(\Delta B'^T \Delta B') \qquad (2.77)$$

Substituting (2.75) and (2.76) into (2.77) produces :

$$\prod_{i=1}^{n+d} \sigma_i^2 = \prod_{i=1}^{n} \sigma'^2_i \;\det(\Delta B'^T \Delta B)$$

$$\prod_{i=1}^{d} \sigma_{n+i}^2 = \prod_{i=1}^{n} (\sigma'_i / \sigma_i)^2 \;\det(\Delta B'^T \Delta B')$$

Since $\sigma_n > \sigma_{n+1}$ and V_{22} nonsingular $[\Delta\hat{A};\Delta\hat{B}]$ is given by (1.22). Hence,

$$\prod_{i=1}^{d} \sigma''^{2}_i([\Delta\hat{A};\Delta\hat{B}]) = \prod_{i=1}^{d} \sigma^2_{n+i} \text{ , which proves (2.74).} \quad \blacksquare$$

For the one-dimensional case, (2.74) reduces to (see [122]) :

$$\|[\Delta\hat{A};\Delta\hat{B}]\|_F = \prod_{i=1}^{n} (\sigma'_i / \sigma_i) \, \|\Delta b'\|_2 \qquad (2.78)$$

Theorem 2-14 implies that the **orientation of B** with respect to the singular vectors u'_i of A **characterizes the ratio of the TLS to the LS approximation effort.** If $B \in R(A)$, (0.1) has an exact solution $X = \hat{X} = X'$ and hence both approximation efforts are zero. If $B \notin R(A)$ the TLS approximation effort becomes smaller and smaller with respect to the LS approximation effort as B lies more along the singular vectors associated with the smallest singular values σ'_n, σ'_{n-1} ... of A. It is minimal if B lies along u'_n. Especially in such cases the TLS solution is more accurate than the LS solution. This is illustrated in chapter 3. We can now derive the following **bounds** for the TLS approximation effort.

Theorem 2-15 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., [A;B]).
If $\sigma'_n > \sigma_{n+1} = \ldots = \sigma_{n+d}$, then :

$$\max_{\sigma_1,\ldots,\sigma_n} \|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 / \|\Delta B'\|_F^2 = 1 \qquad (2.79)$$

$$\min_{\sigma_1,\ldots,\sigma_n} \|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 / \|\Delta B'\|_F^2 = \sigma'^{2}_n / (\sigma'^{2}_n - \sigma^2_{n+1} + \|B\|_F^2 / d) \qquad (2.80)$$

Proof :

The interlacing property for singular values (see theorem A.3) implies that $\sigma_i \geq \sigma'_i$ for i = 1, ..., n. Equality is obtained by choosing [A;B] such that $\sigma'_i = \sigma_i$. This implies that $B \perp R(A)$. Hence, $\|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 = \|B\|_F^2 = \|\Delta B'\|_F^2$, which proves (2.79).
To prove (2.80), we start from the following equality, proven in [47] and extended to multidimensional problems satisfying the assumptions above :

$$R'^{T}R' = \sigma^2_{n+1} \left[I_d + B^T U'_1 \left(\Sigma'^{2} - \sigma^2_{n+1} I_n \right)^{-1} U'^{T}_1 B \right] \qquad (2.81)$$

Since R' equals $\Delta B'$ and $\|\Delta B'\|_F^2 = \mathrm{tr}\,(\Delta B'^{T} \Delta B')$, we obtain by taking the trace of both sides of (2.81) :

$$\|\Delta B'\|_F^2 / \sigma^2_{n+1} = \mathrm{tr}\left[I + B^T U'_1 \left(\Sigma'^{2} - \sigma^2_{n+1} I \right)^{-1} U'^{T}_1 B \right] \qquad (2.82)$$

Since $\mathrm{tr}(A + B) = \mathrm{tr}(A) + \mathrm{tr}(B)$ and $\|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 = d\,\sigma^2_{n+1}$, this yields :

$$\|\Delta B'\|_F^2 / \|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 = \left[d + \sum_{i=1}^{n} \|u'^{T}_i B\|_2^2 / (\sigma'^{2}_i - \sigma^2_{n+1}) \right] / d \qquad (2.83)$$

To prove (2.80), the expression in (2.83) must be maximized. This is the case if the orthogonal projection B' of B into R(A) is parallel with u'_n. In this case, (2.83) equals :

$$\max_{\sigma_1,\ldots,\sigma_n} \|\Delta B'\|_F^2 / \|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 = 1 + \|B'\|_F^2 \, d^{-1} \, (\sigma'^{2}_n - \sigma^2_{n+1})^{-1} \qquad (2.84)$$

Since $\|B'\|_F^2 = \|B\|_F^2 - \|\Delta B'\|_F^2$, (2.84) yields :

$$\max_{\sigma_1,\ldots,\sigma_n} \|\Delta B'\|_F^2 / \|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 = \frac{\sigma'^{2}_n - \sigma^2_{n+1} + \|B\|_F^2 / d - \|\Delta B'\|_F^2 / d}{\sigma'^{2}_n - \sigma^2_{n+1}} \qquad (2.85)$$

By moving the last term on the right of (2.85) to the left and since $\|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 = d\,\sigma^2_{n+1}$, we obtain :

$$\max_{\sigma_1,\ldots,\sigma_n} \frac{\|\Delta B'\|_F^2}{\|[\Delta\hat{A};\Delta\hat{B}]\|_F^2} \cdot \frac{\sigma'^{2}_n}{\sigma'^{2}_n - \sigma^2_{n+1}} =$$

$$(\sigma'^{2}_n - \sigma^2_{n+1} + \|B\|_F^2 / d) \; / \; (\sigma'^{2}_n - \sigma^2_{n+1}) \qquad (2.86)$$

By multiplying both sides of (2.86) with $(\sigma'^{2}_n - \sigma^2_{n+1})\,\sigma'^{-2}_n$, we obtain (2.80). $\blacksquare$

Observe that the maximum (2.79) is obtained for B and U_2 parallel. Since $B \perp R(A)$, TLS as well as LS give the zero solution. The minimum (2.80) is obtained for B belonging to the space generated by u'_n and u_{n+1} i.e. B

is mainly directed along the singular direction of A which is most sensitive to noise. The superiority of TLS in those cases becomes apparent in the experiments described in chapter 3. TLS in fact projects A and B onto the n-dimensional subspace $R([\hat{A};\hat{B}])$ with minimal correction $[\Delta\hat{A};\Delta\hat{B}]$. Depending on the location of $R([\hat{A};\hat{B}])$ the bending effort of A or B will be larger. The following theorem shows how $\hat{X}$ determines **the relative orientation** of the space $R([\hat{A};\hat{B}])$ with respect to the column of A and B.

Theorem 2-16 :

Let (0.3) be the SVD of $[A;B]$.

If $\sigma_n > \sigma_{n+1} = \ldots = \sigma_{n+d}$ and V_{22} nonsingular, then :

$$\Delta\hat{B}^T \Delta\hat{B} = \sigma_{n+1}^2 \, (\hat{X}^T\hat{X} + I)^{-1} \tag{2.87}$$

$$\Delta\hat{A}^T \Delta\hat{A} = \sigma_{n+1}^2 \, \hat{X} \, (\hat{X}^T\hat{X} + I)^{-1} \, \hat{X}^T \tag{2.88}$$

Proof :

From (1.22) and the condition $\Sigma_2 = \sigma_{n+1} I_d$, we have :

$$\Delta\hat{B}^T \Delta\hat{B} = \sigma_{n+1}^2 \, V_{22} \, V_{22}^T \tag{2.89}$$

Substituting (2.39) into (2.89), give (2.87).

From (1.22) and the condition $\Sigma_2 = \sigma_{n+1} I_d$, we have :

$$\Delta\hat{A}^T \Delta\hat{A} = \sigma_{n+1}^2 \, V_{12} \, V_{12}^T \tag{2.90}$$

Substituting (2.40) into (2.90), gives (2.88). ■

It is easily proven that (2.87) and (2.88) are still valid if $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1} = \ldots = \sigma_{n+d}$. In the one-dimensional case and using $\sigma_{n+1}^2 = \|[\Delta\hat{A};\Delta\hat{B}]\|_F^2$, (2.87) and (2.88) reduce to :

$$\|\Delta\hat{B}\|_2^2 = \|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 \, / \, (\|\hat{x}\|_2^2 + 1) \tag{2.91}$$

$$\|\Delta\hat{A}\|_F^2 = \|[\Delta\hat{A};\Delta\hat{B}]\|_F^2 \, \|\hat{x}\|_2^2 \, / \, (\|\hat{x}\|_2^2 + 1) \tag{2.92}$$

Equations (2.91) and (2.92) have nice **geometric** properties. Depending on the norm of the solution $\hat{x}$, TLS forces A to move more towards b or conversely to move b more towards A. If $\|\hat{x}\|_2$ is large $\|\Delta\hat{B}\|_2$ is small : A is

bent towards b. If $\|\hat{x}\|_2 = 1$ then $\|\Delta\hat{B}\|_2 = \|\Delta\hat{A}\|_F$: A and b move equally towards each other. If $\|\hat{x}\|_2$ is small, $\|\Delta\hat{A}\|_F$ is small and b is bent towards A. Observe that in the last case TLS comes close to LS where b' is completely bent towards A.

Theorem 2-16 has interesting consequences in a **noisy** environment. In particular, if A and b have the same level of inaccuracy, we expect a more accurate solution when $\|\Delta\hat{B}\|_2 \approx \|\Delta\hat{a}_i\|_2$ for i = 1, .., n. From (2.71), this happens if each component $\hat{x}_i$ of the solution $\hat{x}$ equals one, or when $\|\hat{x}\|_2 \approx \sqrt{n}$. Hence, the solution $\hat{x}$ tends to be more accurate if the relative location of the space $R([\hat{A};\hat{B}])$ with respect to R(A) and b is according to their relative inaccuracy levels.

2.5. SUBSPACE PROPERTIES

In this section we want to study **how close** the **approximate subspaces** $R([A;B']) = R(A)$ of LS and $R([\hat{A};\hat{B}]) = R(\hat{A})$ of TLS are, into which the data A and B are projected to make the set equations (0.1) compatible. A natural way of describing the closeness of R(A) and $R(\hat{A})$ is to define a **distance function** between any two subspaces. This may be done as follows [108] :

Distance definition :

Let S_1, $S_2 \subset \mathbb{R}^n$ be subspaces. The distance between S_1 and S_2 is the number :

$$\text{dist}(S_1, S_2) = \max \left\{ \sup_{\substack{\|x\|=1 \\ x \in S_1}} \inf_{y \in S_2} \|x-y\|, \; \sup_{\substack{\|y\|=1 \\ y \in S_2}} \inf_{x \in S_1} \|y-x\| \right\} \tag{2.93}$$

Throughout this thesis we always **assume the 2-norm** (0.5) in the distance definition. In this important case, the distance function is a metric. More important for our purposes is the following theorem that relates the distance between S_1 and S_2 to their **projectors** onto S_1 and S_2. The latter are defined as follows. If the columns of $V = [v_1, \ldots, v_k]$ are an orthonormal basis for a subspace S, then

$$P = V \, V^T \tag{2.94}$$

is the unique orthogonal projector onto S. It should be stressed that although P is unique, V is not.

Theorem 2-16 :

Let P_1 and P_2 be the orthogonal projectors onto the subspaces S_1 and $S_2 \subset \mathbb{R}^n$. If the 2-norm is used in (2.93), then :

$$\text{dist } (S_1, S_2) = \|P_1 - P_2\|_2 \tag{2.95}$$

The proof is given in [61]. In case of one-dimensional subspaces, dist (S_1, S_2) is geometrically given by the sine of the angle between the two subspaces and equals the largest singular value of $P_1 - P_2$ [48, section 2.4]. This nice geometrical interpretation of the distance function can be extended to higher dimensions by invoking the C-S decomposition.

We now show how an appropriately chosen C-S decomposition reveals much about the "closeness" of the equidimensional subspaces R(A) of LS and $R(\hat{A})$ of TLS.
Consider the orthogonal matrix $Q = U'^T U$ where $U' = [U_1'; U_2']$ as given in (0.2), and $U = [U_1; U_2]$ as given in (0.3), are orthogonal and U_1', $U_1 \in \mathbb{R}^{m \times n}$ and U_2', $U_2 \in \mathbb{R}^{m \times (m-n)}$. The columns of U_1' (resp., U_1) form an orthogonal set of base vectors for $R(A) = R(U_1')$ (resp., $R(\hat{A}) = R(U_1)$). For simplicity, we assume $m \leq 2n$. Take now (2.21) with $d = m-n$ as the C-S decomposition of Q, partitioned as follows :

$$Q = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} = \left[\begin{array}{c|c} U_1'^T U_1 & U_1'^T U_2 \\ \hline U_2'^T U_1 & U_2'^T U_2 \end{array}\right] \begin{matrix} n \\ m-n \end{matrix} \tag{2.96}$$

The numbers θ_i, $i = 1, \ldots, m-n$ in (2.21) define the **canonical angles** between R(A) and $R(\hat{A})$ i.e. the angles between the i-th column of $U_1'^T W_1$ and the i-th column of $U_1^T T_1$. The cosines $c_i = \theta_i$ of the canonical angles are the singular values of $U_1'^T U_1$ [108]. The importance of the canonical angles is that **the distance between any two subspaces is just the sine of the largest canonical angle.** Taking (2.21) with $d = m-n$ as C-S decomposition of Q in (2.96), this means that the distance between the subspaces R(A) and $R(\hat{A})$ is given by :

$$\text{dist } (R(A), R(\hat{A})) = \|S\|_2 = \|\text{diag } (\sin\theta_1, \ldots, \sin\theta_{m-n})\|_2 = \sin\theta_{m-n} \tag{2.97}$$

Using the various relations between the singular values of the submatrices Q_{ij} in the C-S decomposition (see (2.27), (2.28) and (2.29)), **the distance between the subspaces R(A) and $R(\hat{A})$** can be computed in different ways, as shown in theorem 2-18.

Theorem 2-18 :

Let (0.3) be the SVD of [A;B]. Assume $m \leq 2n$, $\sigma_n > \sigma_{n+1}$ and V_{22} nonsingular. Consider the orthogonal matrices $U' = [U_1'; U_2']$, as given in (0.2), and $U = [U_1; U_2]$ as given in (0.3) with
U_1', $U_1 \in \mathbb{R}^{m \times n}$ and U_2', $U_2 \in \mathbb{R}^{m \times (m-n)}$.
Then $R(A) = R(U_1')$ and $R(\hat{A}) = R(U_1)$ and the distance between both spaces is given by

$$\text{dist } (R(A), R(\hat{A})) = \sqrt{1 - \gamma_{min}^2}$$

with γ_{min} the minimal singular value of $U_1'^T U_1$ (2.98)

$$= \|U_1'^T U_2\|_2 \tag{2.99}$$

$$= \|U_2'^T U_1\|_2 \tag{2.100}$$

$$= \|U_1' U_1'^T - U_1 U_1^T\|_2 \tag{2.101}$$

Proof :

The proof follows from a straightforward application of [48, corollary 2.4-2] and the properties of the C-S decomposition. ■

Comments :

- We assume $m \leq 2n$ for simplicity. If $m > 2n$, we may work with the orthogonal complements of R(A) and $R(\hat{A})$.
- Theorem 2-18 can be easily extended if rank (A) = $n-p < n$. Then also rank $(\hat{A}) = n-p$. Assume $\sigma_{n-p} > \sigma_{n-p+1} = \ldots = \sigma_{n+1}$. Then the same expressions of theorem 2-18 can now be used if U_1' (resp., U_1) contains the first n-p left singular vectors of A (resp., [A;B]), and U_2' (resp. U_2) the remaining ones.
- With respect to the **residuals** R' of LS and $\hat{R}$ of TLS, the distance can be characterized as follows. Since $R(\hat{R}) = R(U_2)$ (see theorem 2-10) and $R(R') = R(U_2')$, it follows from (2.99) that the distance between R(A)

and $R(\hat{A})$ is also the 2-norm of the projection of $R(\hat{R})$ onto $R(A)$, or equivalently from (2.100), the 2-norm of the projection of $R(R')$ onto $R(\hat{A})$.

- Since $R(A) = R(U_1')$ is orthogonal onto $R([A;B])^{\perp} = \text{span}\{u_{n+d+1}', \ldots, u_m\}$, it follows that only d canonical angles of $U'^T U$ given by (2.96), differ from zero. In the **one-dimensional case**, i.e. $d = 1$, only θ_{m-n} differs from zero. In this case, the **distance** between $R(A)$ and $R(\hat{A})$ can be more nicely characterized, as proven below :

Corollary 2-19 :

Let (0.2) (resp., (0.3)) be the SVD of A (resp., [A;b]). Assume $m < 2n$, $\sigma_n > \sigma_{n+1}$, $d = 1$ and $v_{n+1,n+1} \neq 0$. Then the distance between $R(A)$ and $R(\hat{A})$ is given by :

$$\text{dist}\,(R(\hat{A}), R(A)) = \sigma_{n+1} \sqrt{\sum_{i=1}^{n} (v_i'^T \hat{x} / \sigma_i')^2} \Big/ \sqrt{\|\hat{x}\|_2^2 + 1} \qquad (2.102)$$

$$= \|\hat{b} - \hat{A}x'\|_2 / \|b - Ax'\|_2 \qquad (2.103)$$

$$= \sqrt{1 - \cos^2\theta} \text{ with } \cos\theta = \frac{\sigma_{n+1}\,(1 + \hat{x}^T x')}{\|b - Ax'\|_2 \sqrt{\|\hat{x}\|_2^2 + 1}} \qquad (2.104)$$

Proof :

Since $d = 1$, it follows that $R(U_1') \perp \text{span}\{u_{n+2}, \ldots, u_m\}$. Hence, (2.99) reduces to $\|U_1'^T u_{n+1}\|_2$. Using the SVD (0.2) and (0.3) of A and [A;b] respectively, we have :

$$U_1'^T u_{n+1} = \Sigma'^{-1} V'^T A\, u_{n+1} = \sigma_{n+1}\, \Sigma'^{-1} V'^T [v_{1,n+1}, \ldots, v_{n,n+1}]^T \qquad (2.105)$$

or using (1.16) and (2.35), this yields :

$$U_1'^T u_{n+1} = -\sigma_{n+1}\, \Sigma'^{-1} V'^T \hat{x}\, v_{n+1,n+1} = -\sigma_{n+1}\, \Sigma'^{-1} V'^T \hat{x} / \sqrt{\|\hat{x}\|_2^2 + 1} \qquad (2.106)$$

Taking the 2-norm of (2.106) and using the dyadic decomposition, gives (2.102).

(2.103) can be proven using (2.100). Since r' is the only vector of [A;b] which belongs to $R(U_2')$, we may, without restriction, assume that $u_{n+1}' = r' / \|r'\|_2$. Hence, it follows that $R(U_1) \perp \text{span}\{u_{n+2}', \ldots, u_m'\}$ and we obtain from (2.100) :

$$\|U_2'^T U_1\|_2 = \|U_1^T u_{n+1}'\|_2 = \|U_1^T r'\|_2 / \|r'\|_2 = \|U_1^T (b - Ax')\|_2 / \|b - Ax'\|_2 \qquad (2.107)$$

Since $U_1^T [A;b] = U_1^T [\hat{A};\hat{b}]$ and $\hat{b} - \hat{A}x' \in R(U_1)$, (2.103) follows directly from (2.107).

(2.104) can be proven by noting that the angle θ between $R(A)$ and $R(\hat{A})$ is also given by the angle between their normals, given by the LS residual r' and the TLS residual $\hat{r}$ respectively. Since $\hat{r} \,/\!/\, u_{n+1}$ from (2.54), we obtain :

$$\cos\theta = |u_{n+1}^T r'| / \|r'\|_2 = |u_{n+1}^T [A;b][x'^T;-1]^T| / \|b - Ax'\|_2 \qquad (2.108)$$

Using the SVD (0.3) of [A;b], this yields :

$$\cos\theta = |\sigma_{n+1}\, v_{n+1}^T [x'^T;-1]^T| / \|b - Ax'\|_2 \qquad (2.109)$$

Substituting (1.16) and (2.35) into (2.109) gives (2.104). ■

Observe from (2.102) that **the distance between R(A) and R(Â) depends** on different parameters.

1. $\sigma_{n+1} = \|[\Delta\hat{A};\Delta\hat{b}]\|_F$. If $b \in R(A)$, then $\sigma_{n+1} = 0$ and dist $(R(A),R(\hat{A})) = 0$, i.e. the set $Ax = b$ is compatible. If σ_{n+1} is increasing, i.e. the set of equations becomes **less compatible**, then the distance between $R(A)$ and $R(\hat{A})$ **enlarges.**

2. The **orientation** of the TLS solution $\hat{x}$ with respect to the right singular vectors v_i' of A, especially those vectors associated with the smallest singular values of A. Or equivalently, the distance depends on the orientation of b with respect to the left singular vector u_i' of A. Indeed, substituting (2.8) into (2.102) gives :

$$\text{dist}\,(R(A),R(\hat{A})) = \sigma_{n+1} \sqrt{\sum_{i=1}^{n} (u_i'^T b)^2 / (\sigma_i'^2 - \sigma_{n+1}^2)^2} \Big/ \sqrt{\|\hat{x}\|_2^2 + 1} \qquad (2.110)$$

The distance enlarges if b is closer to the lower singular vectors u_n', u_{n-1}', ... The distance is **maximal** if $b \,/\!/\, u_n'$. For this case we **expect large differences between the LS and TLS solution and thus, a large distance between R(A) and R(Â), as shown in chapter 3.**

A nice relation between the subspaces R(A) and R($\hat{A}$) of LS and TLS and their respective residuals and solutions is given in the following corollary.

Corollary 2-20 :

Let (0.2) (resp. (0.3)) be the SVD of A (resp., [A;b]). Assume $\sigma'_n > \sigma_{n+1}$, d = 1 and m $\leq$ 2n. With r' = b - Ax' and $\hat{r}$ = b - A$\hat{x}$ the LS and TLS residual respectively, the distance between R(A) and R($\hat{A}$) is given by :

$$\text{dist}\,(R(A), R(\hat{A})) = \|\hat{r} - r'\|_2 \,/\, \|\hat{r}\|_2 \qquad (2.111)$$

$$= \sigma^2_{n+1}\,(1 + \hat{x}^T x') \,/\, (\|\hat{r}\|_2\,\|x'\|_2) \qquad (2.112)$$

Proof :

Substituting (2.67) and (2.58) into (2.110) gives (2.111).
Substituting (2.58) into (2.103) gives (2.112). ■

2.6. CONCLUDING REMARKS

In this chapter, the **TLS technique** is **compared** analytically with the ordinary **LS method** by proving various interesting **relationships between** their **solutions** (section 2.2), their **residuals** (section 2.3), their **approximation efforts** (section 2.4) and their **approximate subspaces** (section 2.5). All these properties enlighten different new aspects of the TLS approach and its relation to the LS approach. They are useful to get more **insight into the sensitivity** of both techniques with respect to perturbations on the data, discussed in the next chapter.

In particular, various upper bound expressions show that the **difference** between the TLS and LS solution of the set AX = B and their corresponding residuals, is **maximal** when B is parallel with the singular vector of matrix A, associated with its smallest singular value. In this case, the ratio of the TLS to the LS approximation effort reaches its minimum while the distance between the TLS and LS approximate subspaces is maximal. As will be illustrated in chapter 3, the superiority in accuracy of the TLS solution w.r.t. perturbations on the data is then most pronounced.

Several other properties clearly show how **the differences between the TLS and LS approach increase** when the equations AX = B are less compatible, when the norm of matrix B or the solution X is growing or when A tends to be rank-deficient. As will be shown in chapter 3, the most **influencing parameters** namely the minimal singular values of A and [A;B], the orientation of B within the range of A and the length of B or X, also determine the relative difference in sensitivity between the TLS and LS technique.

Furthermore, it was shown how TLS can be considered as a **weighted** LS problem (section 2.3). Also assumptions about the underlying **perturbation models** of both techniques were deduced. As TLS makes only assumptions about the noise distribution in the right singular subspace of the data [A;B], associated with the smallest singular values, many perturbation models correspond with the same TLS solution (section 2.4).

Finally, note that several properties proven in this chapter, require equality of the d smallest singular values of the data [A;B] in case of a d-dimensional TLS problem. This condition may appear too restrictive. However, if the data represent a measured sample of an exact but unobservable relation $A_o X = B_o$ and are rowwise perturbed by independently and identically distributed errors with zero mean and common covariance matrix σ^2 I, this condition is statistically satisfied (see chapter 3 and 4).

CHAPTER 3

THE SENSITIVITY OF THE TLS PROBLEM AND ITS RELATION TO THE LS PROBLEM

3.1. INTRODUCTION

The relationships between the TLS problem and the ordinary LS problem discussed in the previous chapter, are useful tools for the sensitivity study of this chapter. Hereto, we compare the **relative deviations** of the TLS and LS approach in the presence of errors with respect to the corresponding unperturbed problem $A_o x = b_o$. Since the TLS technique is especially devised for solving perturbed problems, this sensitivity study is of unmistakable value for anyone who wants to apply TLS. Indeed, a careful analysis of its **expected** and **worst** perturbation effects w.r.t. accuracy, reliability, stability, ... for all possible conditions of perturbation that (may) occur, improves undoubtedly the reader's insight into the **validity** of the TLS technique, i.e. the domain of applicability of its estimate. A **comparison** with the known sensitivity results of the LS method highly **elucidates the significance** of the TLS technique.

First of all, the **sensitivity of both solutions** is compared in section 3.2. This can be measured by computing the **condition number** of the problem $A_o x = b_o$, quantifying the maximal error effect on its solution by solving the corresponding perturbed problem $Ax = b$. Based on the classical worst perturbation results for LS problems, we derive the **worst perturbation** effects on the TLS solution w.r.t. the solution of the unperturbed problem $A_o x = b_o$. **Zero residual** (i.e. $b_o \in R(A_o)$) as well as nonzero residual (i.e. $b_o \notin R(A_o)$) problems are investigated and emphasize the impact of the residue on the sensitivity. The computed upper bounds for the **absolute** and **relative** error on the LS and TLS solution allow to deduce the conditions for which the bounds are attained and

shed new light on their significance. In particular, it is shown that their **worst effect is attained under different conditions** of perturbation and orientation of the data.

Second, the **sensitivity properties of the SVD** are discussed in section 3.3. First of all, we recapitulate the main **perturbation results** for the singular values and their associated singular subspaces. They allow to measure for the LS as well as for the TLS problem, the **maximal distance** between their perturbed singular subspaces and the corresponding ones of the unperturbed problem. These expressions are the key for understanding the relative differences in sensitivity between the TLS and the LS method. Additionally, we investigate the effects of some **special subspace perturbations** which **maximally** affect the accuracy of the TLS problem. We also compute the effect of singular subspace perturbations on the TLS solution for the one-dimensional problem.

Finally in section 3.4, the **expected differences** between the TLS and LS approach are illustrated by **simulations.** They not only **confirm the main results** of the sensitivity analysis discussed in the previous sections, but clearly show **the most influencing parameters** and their importance w.r.t. the applicability of the TLS technique. They certainly provide very valuable information for anyone who wants to evaluate the **practical** use of TLS for his own problems.

3.2. SENSITIVITY PROPERTIES OF THE SOLUTION

3.2.1. CONDITION OF THE PROBLEM

Our aim is to examine how **perturbations** in A and b **affect the solution** x. Throughout this section we consider the one-dimensional problem, i.e. d = 1 in (0.1). From the expressions (2.12) and (2.8) of the LS and TLS solution respectively, it is clear that small changes in A and b can induce relatively large changes in x if the minimal singular value σ'_n of A is small. This is not surprising when we recall from the Eckart-Young theorem (see theorem A.2) that σ'_n is a measure of the distance between A and the set of rank-deficient matrices. As we approach this set, it is

intuitively clear that the solution x should be increasingly sensitive to perturbations.

A precise measure of **square** linear system sensitivity can be obtained by considering the parametrized square system :

$$(A + \Delta A)\ x\ (\varepsilon) = b + \Delta b \qquad x(0) = x \tag{3.1}$$

where $A, \Delta A \in \mathbb{R}^{n\times n}$ and $b, \Delta b \in \mathbb{R}^{n}$. Assuming that A is invertible and defining the **condition number** $\kappa(A)$ by

$$\kappa(A) = \|A\|\,\|A^{-1}\| \tag{3.2}$$

it can be shown [48, section 2.5] that

$$\|x(\varepsilon) - x\| \,/\, \|x\| \leq \kappa(A)\,[\,\|\Delta A\| / \|A\| + \|\Delta b\| / \|b\|\,] + O(\varepsilon^2) \tag{3.3}$$

where $\|\Delta A\| / \|A\|$ and $\|\Delta b\| / \|b\|$ represent the relative errors in A and b respectively. Thus, the relative error in x can be $\kappa(A)$ times the relative error in A and b. In this sense, **the condition number $\kappa(A)$ quantifies the sensitivity of the problem Ax = b.** Using the **2-norm** in (3.2) and taking the SVD (0.2) of A with $\sigma'_n \neq 0$, the condition $\kappa(A)$ is given by

$$\kappa_2(A) = \|A\|_2\,\|A^{-1}\|_2 = \sigma'_1 / \sigma'_n \tag{3.4}$$

Throughout this thesis we always assume the 2-norm condition. Hence, the subscript in (3.4) can be omitted. We always have $\kappa(A) \geq 1$. Matrices with small (resp., large) condition numbers are said to be **well**-conditioned (resp., **ill**-conditioned). Orthogonal matrices Q are perfectly conditioned in that $\kappa(Q) = 1$.

If we consider now **rank-deficient** or **rectangular** matrices, e.g. in **overdetermined** linear equation problems, the notion of condition is **extended** as follows :

$$\kappa(A) = \|A\|\,\|A^{+}\| \tag{3.4}$$

with $A \in \mathbb{R}^{m\times n}$ and A^+ its **pseudo-inverse** [48, section 6.1]. In the important case of 2-norm condition and taking the SVD (0.2) of A, (3.4) gives :

$$\kappa(A) = \sigma'_1 / \sigma'_r \tag{3.5}$$

where r = rank (A). Analogously to the square linear equation problem, the condition number (3.4) or (3.5) measures the sensitivity of the solution to errors in the matrix A and right handside b. It shows that small changes in the elements of A can induce large changes in the solution x of (0.1), if A's columns are nearly dependent.

When studying the **sensitivity in overdetermined** linear equation sets, it is important to distinguish between **zero** and **nonzero residual problems.** Fig. 3.1. helps to clarify this relationship. Define the angle θ_o between the unperturbed b_o and $R(A_o)$ by

$$\sin\theta_o = \|b_o - A_o x\|_2 \,/\, \|b_o\|_2 \tag{3.6}$$

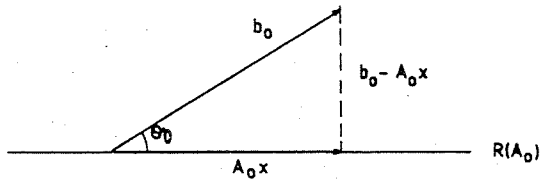


Fig. 3.1. : Geometric illustration of the angle θ_o between b_o and $R(A_o)$, given by $\sin\theta_o = \|b_o - A_o x\|_2 / \|b_o\|_2$.

Then a problem is said to be **zero residual** if $b_o \in R(A_o)$, i.e. θ_o is zero. In other words, in the absence of any perturbation on the data, an exact but **unobservable linear relationship** $A_o x = b_o$ **exists.** The **TLS problem typically applies to those cases** and gives consistent estimates of the solution x, as shown in chapter 4. The sensitivity of the solution depends **linearly** on the condition number and it is easily shown that LS and TLS may be **equally** sensitive.

In **nonzero residual** problems, $b_o \notin R(A_o)$ and thus, the angle θ_o in (3.6) is no longer zero. This means that, even in the absence of any perturbation on the data, **no exact** linear **relationship** $A_o x = b_o$ can be found. As shown in corollary 3-2, it is the **square** of the condition number that measures the sensitivity of nonzero residual LS problems and it is proven that the **TLS** problem may be **more sensitive** than the LS problem. Those problems occur, for example, in model fitting and prediction when

one wants to approximate nonlinear data by linear models or predict the system response by a simplified model of the system. In those cases LS must be applied rather than TLS.

3.2.2. UPPER BOUNDS OF THE LS SOLUTION PERTURBATION

For any given minimization problem (1.3) we shall investigate the **maximum (or worst) perturbation** caused in its solution. The importance of this notion of worst perturbation is that **in the absence of further information about ΔA and Δb**, the worst perturbation should be considered as an **attainable** or realistic perturbation of the minimization problem at hand.

Let the SVD of the unperturbed A_o be given by (0.2). We assume $\sigma'_n \neq 0$ throughout this section and investigate the effect of perturbations for given $\varepsilon > 0$ satisfying :

$$\|\Delta A\|_2 \leq \varepsilon \|A_o\|_2 = \varepsilon \sigma'_1 \quad \text{and} \quad \|\Delta b\|_2 \leq \varepsilon \|b_o\|_2 \tag{3.7}$$

It is reasonable to limit ourselves to $\varepsilon \kappa(A_o) < 1$ in order to ensure uniqueness of the solution of our perturbed LS problem. Let x' minimize (1.3). For given ΔA and Δb we define $\Delta x'$ as the vector which minimizes

$$\|(A_o + \Delta A)(x' + \Delta x') - (b_o + \Delta b)\|_2 \tag{3.8}$$

Since (3.8) equals $\|(A_o + \Delta A)\Delta x' - r' + \Delta A x' - \Delta b\|_2$ (3.9)

with r' given by $r' = b_o - A_o x'$, the minimizing $\Delta x'$ must be such that $(A_o + \Delta A)\Delta x'$ is the projection of $r' - \Delta A x' + \Delta b$ on $R(A_o + \Delta A)$. Writing $A = A_o + \Delta A$, we have :

$$\Delta x' = (A^T A)^{-1} A^T (r' - \Delta A x' + \Delta b) \tag{3.10}$$

Then the following **upper bounds of the absolute error** $\|\Delta x'\|_2$ in case of finite perturbations have been proven by A. Van der Sluis [118].

Theorem 3-1 : **Absolute error upper bounds for the LS solution** x' of a set $A_o x = b_o$.

Let (0.2) be the SVD of A_o and denote $\mu = \varepsilon \sigma'_1 / \sigma'_n < 1$. Then, for any perturbation $(\Delta A, \Delta b)$ satisfying (3.7) :

$$\|\Delta x'\|_2 \leq \frac{\varepsilon}{\sigma'_n}\left[\frac{\sigma'_1 \|r'\|_2}{\sigma'_n (1-\mu^2)} + \frac{\sigma'_1 \|x'\|_2}{1-\mu} + \frac{\|b_o\|_2}{1-\mu}\right] \tag{3.11}$$

For zero residual problems the same bound (3.11) is valid if we set $\|r'\|_2$ zero.

Theorem 3-1 assesses **the effect of the residue on $\Delta x'$** exactly, if the projection of r' onto A has the direction of the image of the right singular vector v'_n, associated with its minimal singular value, i.e.

$$A A^T r' \,/\!/\, A v'_n \quad \text{and} \quad \|\Delta A\|_2 = \varepsilon \sigma'_1 \tag{3.12}$$

This happens if $A = A_o + q v'^T_n$ with $q \perp \text{span}\{u'_1, \ldots, u'_{n-1}\}$. Then,

- $A v'_i = \sigma'_i u'_i$, $\forall i < n$; $A v'_n = \sigma'_n u'_n + q$
- $v'_1, \ldots, v'_n$ are singular vectors of A also
- and $R(A)$ may be obtained from $R(A_o)$ by a rotation around span $\{u'_1, \ldots, u'_{n-1}\}$.

This is illustrated in fig. 3.2 for the case n = 2. A. Van der Sluis [118] shows that this mechanism in fact **maximizes the effect of the residue on $\Delta x'$**. More generally we may say, that this is due to the coincidence of two events : **rotation of the projection plane around the image of a dominant singular vector produces a large projection of the residue and this projection has the direction of the image of the lowest singular vector.**

The **contribution of $\Delta b - \Delta A x'$ to $\Delta x'$**, i.e. the last 2 terms of (3.11), is **majorized** if :

$$\Delta A = -\varepsilon \sigma'_1 u'_n v'^T_n \subset R(A_o) \qquad \Delta b = \varepsilon \|b_o\|_2 u'_n \in R(A_o) \qquad x' = \|x'\|_2 v'_n \tag{3.13}$$

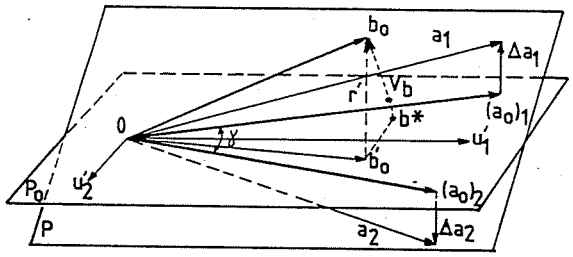


Fig. 3.2 : Geometric illustration of the maximal effect of the residue r' on the error $\Delta x'$ of the LS solution x' of $A_o x = b_o$, due to perturbations Δa_1 and Δa_2 on the columns $(a_o)_1$ and $(a_o)_2$ of A_o. b is the projection of b_o on plane P generated by a_1 and a_2. Rotating P back to P_o, moves b to b*. Observe that $b^* - b_o'$ is parallel with the lowest singular vector u_2' of A_o [118].

For **zero residual** problems, i.e. $\|r'\| = 0$, the **upper bound of** $\|\Delta x'\|$ in (3.11) reduces to :

$$\varepsilon(\sigma_1' \|x'\|_2 + \|b_o\|_2)\ \sigma_n'^{-1}\ (1-\mu)^{-1} \qquad (3.14)$$

and **is indeed attained** under the conditions given by (3.13). In this case, $\Delta x' = \|\Delta x'\|_2\ v_n'$ with $\|\Delta x'\|_2$ given by (3.14).

For **nonzero residual** problems, (3.12) and (3.13) **can not be simultaneously satisfied.** Hence, we can assess the effect of the residue exactly and the bounds of the other effects as nearly as possible, by majorizing the contribution of the projection of $\Delta b - \Delta A x'$ on A to $\Delta x'$. These upper bounds are quite close for $\mu < 0.5$. If r' is large, the first term in (3.11) would dominate $\|\Delta x'\|_2$, i.e. if the overdetermined system $A_o x = b_o$ is very incompatible (see example 3-1).

Observe from theorem 3-1 that if $\|A\|_2 \neq 1$, the coefficient of $\varepsilon \|r'\|_2$ is not the square of a condition number but something of the form $\sigma_1' / \sigma_n'^2$. This is not surprising since condition numbers are typically related to $\|\Delta x'\|_2 / \|x'\|_2$ and not to $\|\Delta x'\|_2$ alone, as shown in the next corollary [118].

Corollary 3-2 : **Relative error upper bounds for the LS solution** x' **of a set** $A_o x = b_o$.

Let (0.2) be the SVD of A_o and denote $\mu = \varepsilon \sigma_1' / \sigma_n' < 1$. Define $F(x') = \|Ax'\|_2 / \|x'\|_2$, hence $\sigma_n' \leq F(x') \leq \sigma_1'$. Also, let θ_o denote the angle between b_o and $R(A_o)$. Then $\|x'\|_2 = \|b_o\|_2 \cos\theta_o / F(x')$ and $\|r'\|_2 = \|b_o\|_2 \sin\theta_o$. Hence, (3.11) transforms into :

$$\frac{\|\Delta x'\|_2}{\|x'\|_2} \leq \varepsilon \left[\frac{\sigma_1'}{\sigma_n'} \frac{F(x')}{\sigma_n'} \frac{\tan\theta_o}{1-\mu^2} + \frac{\sigma_1'}{\sigma_n'} \frac{1}{1-\mu} + \frac{F(x')}{\sigma_n'} \frac{1}{\cos\theta_o} \frac{1}{1-\mu} \right] \qquad (3.15)$$

For zero residual problems, the same bound (3.15) is valid if we set $\theta_o = 0$. The upper bound (3.15) can be assessed under the same conditions (3.12) and (3.13) as given for the absolute error, but these conditions do not maximize the relative error effect. Indeed, the **contribution of the residue** to $\|\Delta x'\|_2 / \|x'\|_2$, i.e. the first term in (3.15), is **maximized if** :

(3.12) is satisfied and the LS solution $x' = \|x'\|_2\ v_1'$ (3.16)

The effect of **the contribution of** $-\Delta A x'$ to $\|\Delta x'\|_2 / \|x'\|_2$, i.e. the second term in (3.15) is **maximized** if :

$$\Delta A = -\varepsilon \sigma_1' u_n' v_n'^T \subset R(A_o) \quad \text{and} \quad x' = \|x'\|_2\ v_n' \qquad (3.17)$$

while the **effect of** Δb to $\|\Delta x'\|_2 / \|x'\|_2$, i.e. the third term in (3.15), is **maximized** if :

$$\Delta b = \varepsilon \|b_o\|_2\ u_n' \in R(A_o) \quad \text{and} \quad x' = \|x'\|_2\ v_1' \qquad (3.18)$$

We immediately see that (3.16), (3.17) and (3.18) can **not be satisfied simultaneously.** Even for zero residual problems the conditions (3.17) and (3.18) may not be satisfied together. Hence the **worst** possible **upper bound** (3.15) is **never attained.**

Observe that the effect of the residue and Δb on the relative error is maximized for $x' /\!/ v_1'$, while the effect of the residue on the absolute error is independent of the solution (see (3.12)) and the effect of b is maximal for $x' /\!/ v_n'$ (see (3.13)) (see example 3-1).

From (3.15) we see that, unless $\tan\theta_o$ is large, $\kappa^2(A_o)$ **plays no role**

for $\|\Delta x'\|_2 / \|x'\|_2$ if $F(x')$ is a negligible fraction of σ_1', i.e. **if x' has a negligible projection on all the v_i' associated with larger singular values.** This is obviously the case for only a small fraction of all vectors x'. It is however the case for a large fraction of all vectors b_o of given θ_o provided that $\kappa(A_o)$ is large. Indeed, as soon as the projection of b_o on u_n' is not small with respect to b_o, x' will belong approximately to the space spanned by the v_i', associated with the smaller singular values. Hence, if $\kappa(A_o)$ is large, then **$\kappa^2(A_o)$ is not realistic with respect to the relative error for most least squares problems** unless b_o is nearly orthogonal onto $R(A_o)$, i.e. $\theta_o \approx \pi/2$.

From corollary 3-2, we get a clearer picture of the **importance** of the various terms as functions of the **degree of incompatibility** of the system $A_o x = b_o$, for which θ_o may be taken as a measure. In particular, we see that for zero residual problems ($\theta_o = 0$), the second term in (3.15) (and therefore in (3.11)) is the most important one, and this virtually remains so as long as θ_o is small. If we let θ_o grow, however, then already from $\theta_o = 1/\kappa(A_o)$ on, the first term may become the most important one, and in fact is the most important one if b_o is such that $F(x') = \sigma_1'$, i.e. the projection of b_o on $R(A_o)$ is parallel with the singular vector u_1', corresponding to σ_1'. Hence, if $\kappa(A_o)$ is large, already a residue corresponding to a b_o with small θ_o (i.e. $A_o x = b_o$ is only slightly incompatible) may no longer be considered small in the sense of having no additional influence on the error. We also see that the full $\kappa^2(A_o)$ impact cannot occur until $\theta_o \approx \pi/4$.
If $\sigma_n' = 0$, then the **minimum norm** LS solution is computed. For those cases, worst perturbation bounds have been given by Hansen [52].

As said before, **the worst perturbation is obtained by rotating $R(A_o)$ around span $\{u_1', \ldots, u_{n-1}'\}$** i.e. $\Delta A = q\, v_n'^T$ with $q \perp \text{span}\{u_1', \ldots, u_{n-1}'\}$. **For those perturbation effects, we can compute the perturbed LS solution analytically**, as proven more generally in the following theorem.

Theorem 3-3 :

Let (0.2) be the SVD of A_o and $b_o = \sum_{i=1}^{n} \beta_i u_i'$. Consider perturbations

$\Delta A = q\, v_k'^T$, $q = \sum_{j=1}^{m} q_j u_j'$ and $\Delta b = \sum_{i=1}^{m} v_i u_i'$ on the exact equations $A_o x = b_o$. Then, the LS solution x' of the perturbed set $(A_o + \Delta A)\, x = b_o + \Delta b$ is given by $\sum_{i=1}^{n} \lambda_i v_i'$ and $\|x - x'\|_2^2 = \sum_{i=1}^{n} (\lambda_i - \beta_i / \sigma_i')^2$ with

$$\lambda_i = (\beta_i + v_i) / \sigma_i' - \lambda_k q_i / \sigma_i' \qquad \forall i \neq k$$

$$\lambda_k = \frac{\beta_k + v_k + \sum_{i=n+1}^{m} q_i \dfrac{v_i}{\sigma_k' + q_k}}{\sigma_k' + q_k + \dfrac{\|q^\perp\|_2^2}{\sigma_k' + q_k}} \quad \text{and} \quad \|q^\perp\|_2^2 = \sum_{i=n+1}^{m} q_i^2 \qquad (3.19)$$

Proof :

The proof is given in [122]. ■

From (3.19) we see that the **accuracy of x'** will be **worse** if the **perturbations are** more **oriented along the lowest singular vectors** (with large k), i.e. u_n', u_{n-1}', ..., and if the perturbations $\|q^\perp\|_2$ **in directions orthogonal on R(A)** become more important. The worst perturbation effects on the LS solution are now compared with the upper bounds (3.11) and (3.15) in the following example.

Example 3-1 :

(a) Consider the following **nonzero residual** problem, similar to [48, example 6.1-2] :

$$\text{If } A_o = \begin{bmatrix} 1 & 0 \\ 0 & 10^{-4} \\ 0 & 0 \end{bmatrix}, \quad \Delta A = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 10^{-8} \end{bmatrix}, \quad b_o = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \quad \Delta b = \begin{bmatrix} 0 \\ 10^{-8} \\ 0 \end{bmatrix} \qquad (3.20)$$

$$\text{then } x' = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } x' + \Delta x' = (A_o + \Delta A)^+ (b_o + \Delta b) = \begin{bmatrix} 1 \\ 1.0001 \end{bmatrix} \qquad (3.21)$$

Since $\kappa(A_o) = 10^4$ and $\varepsilon = 10^{-8}$, we have :

$$\|\Delta x'\|_2 / \|x'\|_2 = \|\Delta x'\|_2 = 1.0001 \leq 1 + 10^{-4} + \sqrt{2}\, 10^{-4}.$$

These are the upper bounds given by (3.11) and (3.15). Observe that the **square of the condition is attained** since the residue is large, i.e. the set of equations is very incompatible (see also fig. 3.3).

(b) Consider now the following **zero residual** problem :

$$A_o = \begin{bmatrix} 1 & 0 \\ 0 & 10^{-2} \\ 0 & 0 \end{bmatrix}, \quad \Delta A = \begin{bmatrix} 0 & 0 \\ 0 & -10^{-3} \\ 0 & 0 \end{bmatrix}, \quad \Delta b = \begin{bmatrix} 0 \\ 10^{-3} \\ 0 \end{bmatrix} \tag{3.22}$$

The condition $\kappa(A_o) = 10^2$ and $\varepsilon = 10^{-3}$.

Let (0.2) be the SVD of A_o.

First, take $b_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} // u'_n$, then $x'_1 = \begin{bmatrix} 0 \\ 10^2 \end{bmatrix} // v'_n$ and $x'_1 + \Delta x'_1 = \begin{bmatrix} 0 \\ 100.1/.9 \end{bmatrix}$ (3.23)

Hence, $\|\Delta x'_1\|_2 / \|x'_1\|_2 = .101/.9$ and $\|\Delta x'_1\|_2 = 10.1/.9$.

Observe that the bounds (3.11) and (3.15) are assessed exactly. Indeed :

$$\|\Delta x'_1\|_2 = (10^{-3} \cdot 10^2 \cdot 10^2 + 10^{-3} \cdot 1 \cdot 10^2) / .9 = 10.1/.9 \tag{3.24}$$

Take now $b_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} // u'_1$, then $x'_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} // v'_1$ and $x'_2 + \Delta x'_2 = \begin{bmatrix} 1 \\ .1/.9 \end{bmatrix}$ (3.25)

Hence $\|\Delta x'_2\|_2 / \|x'_2\|_2 = \|\Delta x'_2\|_2 = .1/.9$. Compare this with the upper bounds (3.11) and (3.15) :

$$\|\Delta x'_2\|_2 / \|x'_2\|_2 = \|\Delta x'_2\|_2 < (10^{-3} \cdot 10^2 \cdot 1 + 10^{-3} \cdot 10^2) / .9 = .2/.9 \tag{3.26}$$

Observe that $\|\Delta x'\|_2$ is maximal for $x' // v'_n$ in (3.24) while $\|\Delta x'\|_2 / \|x'\|_2$ is maximal for $x' // v'_1$ in (3.26). The absolute and relative error upper bound can indeed be assessed exactly (see (3.24)). Since $\mu = \varepsilon \kappa(A_o) = 10^{-1} < .5$, the upper bounds are quite close.

3.2.3. UPPER BOUNDS OF THE TLS SOLUTION PERTURBATION

Based on the results of section 3.2.2, we now deduce the **worst perturbation effects** on the TLS solution and investigate their importance. From the relationship (2.1), it is clear that a TLS solution can be considered as the LS solution of the related problem :

$$(A_o^T A_o - \sigma_{n+1}^2 I)\, \hat{x} = A_o^T b_o \tag{3.27}$$

Using (0.2) : $U'(\Sigma'^2 - \sigma_{n+1}^2 I)^{-1} \Sigma' V'^T \hat{x} = b_o$ (3.28)

whose condition number is given by :

$$\hat{\kappa} = \frac{\sigma'_n}{\sigma'_1} \cdot \frac{\sigma'^2_1 - \sigma^2_{n+1}}{\sigma'^2_n - \sigma^2_{n+1}} > \frac{\sigma'_1}{\sigma'_n} \tag{3.29}$$

Hence, the **same** upper bounds (3.11) and (3.15) may be used for measuring the worst TLS solution perturbation, if we replace σ'_1 by $(\sigma'^2_1 - \sigma^2_{n+1})/\sigma'_1$ and σ'_n by $(\sigma'^2_n - \sigma^2_{n+1})/\sigma'_n$.

For nonzero residual problems, i.e. $\sigma_{n+1} \neq 0$, **the TLS perturbation upper bounds are always larger than the LS perturbation upper bounds.** As already shown in corollary 2-2 and 2-3, the TLS problem becomes even **very** sensitive if σ_{n+1} approaches σ'_n. However, if $\sigma'_n - \sigma_{n+1}$ becomes very small, then we know from theorem 1-3 that $v_{n+1,n+1}$ can be almost zero, i.e. b is nearly orthogonal onto u_{n+1} and $u_{n+1} \approx u'_n$. In those cases, the TLS solution is computed in **nongeneric** sense (see section 1.6). The solution error is kept small and stability is maintained. This is illustrated in example 3-2. Moreover by comparing $v_{n+1,n+1}$ with the perturbation level we can easily detect whether or not b is orthogonal onto u_{n+1} and whether the TLS solution must be computed in the nongeneric sense. For LS problems, it is more difficult to detect insignificant solutions and restore the instability.

Observe also that the **worst perturbation effects only occur if the residue is large**, i.e. the set equations is highly incompatible. As said in section 1.6.1, it is then possible to have nongeneric TLS problems.

For **zero residual** problems, i.e. $\sigma_{n+1} = 0$, we have from (3.29) that

$$\hat{\kappa} = \kappa(A_o) \tag{3.30}$$

Hence, **the TLS and LS solution perturbations have the same bounds in case of zero residual problems** and attain their worst perturbation effect under the same conditions. For TLS, the **worst perturbations** ΔA and Δb occur when $R([\Delta A;\Delta b]) \subset R(A_o)$. Then σ_{n+1} remains zero. **No noise can be rejected since the perturbed set is compatible.** Thus the perturbations will entirely affect the solution accuracy. Under these conditions, span $\{v'_1, \ldots, v'_n\}$ = span $\{v_1, \ldots, v_n\}$. Hence, it follows that LS and TLS are equally affected with respect to their worst perturbations.

We can **illustrate** the TLS perturbation effect with an example of a nonzero residual problem. As mentioned before, the TLS and LS solution perturbation effects have the same bounds in case of zero residual problems.

Example 3-2 :

Consider the same nonzero residual problem (n = 2) as in example 3-1 (a) (see (3.20)), illustrated in fig. 3.3. Since the set of equations is highly incompatible, the nongeneric TLS solution is likely to occur. Taking the SVD (0.3) of $[A_o;b_o]$, we observe that $v_{n+1,n+1} = 0$. Hence, $b_o \perp u'_n$ and $v_{n+1} = v'_n$ by theorem 1-3 (see fig. 3.3) and the TLS solution $\hat{x}$ of the unperturbed problem $A_o x = b_o$ must be computed in the nongeneric sense. Using theorem 1-4, we obtain $\hat{x} = \begin{bmatrix} 2/(\sqrt{5}-1) \\ 0 \end{bmatrix}$ from v_2.

Perturb A_o and b_o with ΔA and Δb as given in (3.20). Taking the SVD (0.3) of $[A;b] = [A_o+\Delta A; b_o+\Delta b]$, we now obtain $v_{n+1,n+1} \approx 10^{-8}$ and $\sigma'_n - \sigma_{n+1} = 0\ (10^{-8})$. Computing the generic TLS solution (1.16) from v_{n+1} produces a very sensitive, unstable solution, given by $\begin{bmatrix} 1 \\ 10^8 + 10^4 \end{bmatrix}$. However, since the perturbation level is 10^{-8}, we may conclude that $b \perp u'_n$. Hence the TLS solution must be computed in the nongeneric sense. Using theorem 1-4, we obtain $\hat{x} + \Delta\hat{x} = \begin{bmatrix} 2/(\sqrt{5}-1) \\ 2/(\sqrt{5}-3)\times 10^{-8} \end{bmatrix}$ from v_n (n = 2). Observe that $\|\Delta\hat{x}\|_2 = 10^{-8}$ is very small and the perturbed TLS solution very stable !

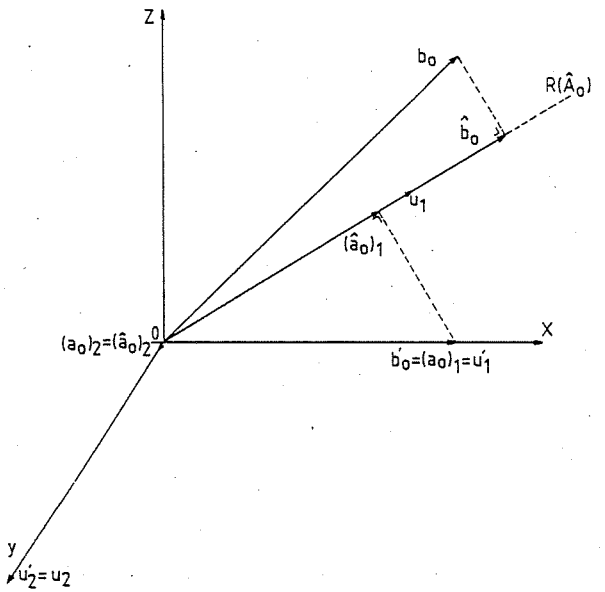


Fig. 3.3 : Geometric illustration of example 3-1 (a) and 3-2.
X, Y and Z are the coordinate axes. A_o has 2 columns $(a_o)_1$ and $(a_o)_2$ and b_o is the observation vector. The TLS approximation is given by $(\hat{a}_o)_1$, $(a_o)_2$, $\hat{b}_o$ and the LS approximation by b'_o. u'_1, u'_2 (resp., u_1, u_2) are the left singular vectors of A_o (resp., $[A_o;b_o]$).

Another relative error upper bound for the TLS solution perturbations is given and proven in [47].

Theorem 3-4 :

Let (0.2) (resp., (0.3)) be the SVD of A_o (resp. $[A_o;b_o]$). If $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$ are such that $\eta = \|[A-A_o;b-b_o]\|_F < \varepsilon/6$

where $\varepsilon = \sigma'_n - \sigma_{n+1} > 0$,
then the perturbed TLS problem $A\,(\hat{x} + \Delta\hat{x}) = b$ has a unique solution $\hat{x} + \Delta\hat{x}$. Moreover, if $\hat{x} + \Delta\hat{x} \neq 0$, then

$$\frac{\|\Delta\hat{x}\|_2}{\|\hat{x}\|_2} \leq \frac{9\eta\,\sigma_1}{\sigma_n - \sigma_{n+1}} \left\{1 + \frac{\|b_o\|_2}{\sigma'_n - \sigma_{n+1}}\right\} \frac{1}{\|b_o\|_2 - \sigma_{n+1}} \qquad (3.31)$$

For most TLS problems, the upper bound (3.31) is very crude and can never be attained. From the proof we deduce that the worst perturbation $[\Delta A;\Delta b]$ occurs if it maximizes the angle between the perturbed singular vector v_{n+1} of $[A;b]$ and the corresponding unperturbed one of $[A_o;b_o]$, associated with σ_{n+1} (see also section 3.3) and if $\hat{x} \,/\!/\, v'_n$. Observe from (3.31) that the TLS solution is less sensitive with increasing $\sigma'_n - \sigma_{n+1}$ and $\|b_o\|_2$ and decreasing σ_1.

From the previous it can be concluded that TLS is **not better** than LS, even worse for nonzero residual problems. However, it must be stressed that we only investigated the **worst** perturbations which are only attained for a **special choice** of b_o and the perturbations ΔA and Δb. Under these conditions only, the upper bounds given by (3.11) and (3.15) are realistic for LS and generic TLS solutions. However, if further **information about ΔA and Δb is given**, e.g. their distribution function or covariance matrix, those upper bounds may be unrealistic. Other more important properties making TLS superior to LS from a **practical** point of view, can be deduced (see section 3.4 and chapter 4).

3.3. SENSITIVITY PROPERTIES OF THE SVD

In section 3.2, upper bounds for the TLS and LS solution errors, due to perturbation effects on the data A_o and B_o, were derived. These upper bounds correspond with a special orientation of B_o with respect to $R(A_o)$ and may not be realistic for the other orientations. In this section, the **perturbation effects on the singular values and associated singular subspaces** of a given matrix C_o, e.g. $[A_o;B_o]$, due to perturbations on its matrix entries, are investigated and bounds are given. Contrary to section 3.2, these bounds do not assume a special choice of B_o w.r.t. $R(A_o)$. Hence, they are more **generally** applicable.

Let's start with a **general description** of the perturbation effects. Consider an m by p matrix C_o of rank r, $m \geq p \geq r$, whose SVD is given by :

$$C_o = U_o\,\Sigma_o\,V_o^T = \underset{r \quad m-r}{[U_1^o;U_2^o]} \begin{bmatrix} \Sigma_1^o & 0 \\ 0 & \Sigma_2^o \\ 0 & 0 \end{bmatrix}_{m-p} \underset{r \quad p-r}{[V_1^o;V_2^o]^T} = \underset{r \quad p-r}{\qquad\qquad}$$

$$\sum_{i=1}^{r} \sigma_i^o\,u_i^o\,v_i^{oT} \quad \text{with} \quad \Sigma_2^o = 0 \qquad (3.32)$$

Add perturbations ΔC to C_o and let the SVD of $C = C_o + \Delta C$ be given by :

$$C = C_o + \Delta C = U\,\Sigma\,V^T = [U_1;U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \\ 0 & 0 \end{bmatrix} [V_1;V_2]^T = \sum_{i=1}^{p} \sigma_i\,u_i\,v_i^T \qquad (3.33)$$

Then the relationship between C, C_o and their SVD is given by :

$$C_o + \Delta C = U\,\Sigma\,V^T = (U_o\;G)\;(\Sigma_o + \Delta\Sigma)\;H^T\,V_o^T \qquad (3.34)$$

and

$$\|\Delta C\|_F = \|G\,(\Sigma_o + \Delta\Sigma)\,H^T - \Sigma_o\|_F \qquad (3.35)$$

with G and H orthonormal matrices.

(3.34) says that the perturbation ΔC of C_o **causes a variation** $\Delta\Sigma$ in the singular spectrum of C_o and **a rotation** G and H of its left and right singular subspace respectively. Indeed G and H can be considered as **products of elementary Givens rotations** [48, section 3.4], which describe the simultaneous rotation of two singular vectors over a certain angle within their own span.

Bounds on the perturbation effects $\Delta\Sigma$ of the singular spectrum have been given in [105] and [48, section 8.3]. Using (3.32) and (3.33), we have :

$$\|\Delta\Sigma\|_F \leq \|\Delta C\|_F \quad \text{and} \quad |\sigma_k - \sigma_k^o| \leq \|\Delta C\|_2 \qquad k = 1, \ldots, p \qquad (3.36)$$

(3.36) says that each singular value is **"perfectly conditioned"**. That

is, the absolute change in a singular value is not more than the absolute change in the matrix.

For the singular subspaces U_o and V_o, the perturbation effects G and H are more delicate, as shown below. First of all we note that the matrices G and H are directly related to the **canonical angles** θ_i (see section 2.5). These angles separate the i-th principal subspaces of C_o and C, i.e. the subspaces generated by their first i singular vectors. Since $G = U^t U_o$ (resp., $H = V^T V_o$), its singular values characterize the **cosine** of the canonical angles between the left (resp., right) principal singular subspaces of C_o and C. These angles are directly related to the so called **"direct rotation"** : among all unitary transformations rotating one space into another, the direct rotation differs minimally from the identity [22].

Bounds on the perturbation effects related to singular subspaces have been given by Stewart [108], Davis and Kahan [22], Staar [105], ... More interesting for the sensitivity study of the TLS problem, are the bounds given by Wedin [133]. Hereto we use the important **relation** (2.97) between the **largest canonical angle** and the concept of **distance** between any two subspaces. Then, the following interesting **upper bounds for the distance between perturbed singular subspaces and their corresponding unperturbed ones**, can be derived.

Theorem 3.5 :

Let the SVD of the m by p matrix C_o ($m \geq p$) be given and partitioned as in (3.32). Call $C_1^o = U_1^o \Sigma_1^o V_1^{oT}$ and $C_2^o = U_2^o \Sigma_2^o V_2^{oT}$. Add perturbations ΔC to C_o and let the SVD of $C = C_o + \Delta C$ be given and partitioned as in (3.33). Call $C_1 = U_1 \Sigma_1 V_1^T$ and $C_2 = U_2 \Sigma_2 V_2^T$.
If $\delta_1 = \sigma_r - \sigma_{r+1}^o > 0$ and $\varepsilon_1 = \max (\|R_{11}\|_2 , \|R_{21}\|_2)$ with R_{11}, R_{21}, defined by

$$R_{11} = C_o V_1 - U_1 \Sigma_1 = - \Delta C V_1 \tag{3.37}$$

$$R_{21} = C_o^T U_1 - V_1 \Sigma_1 = - \Delta C^T U_1 \tag{3.38}$$

then

$$\text{dist}(R(C_1), R(C_1^o)) \leq \varepsilon_1 / \delta_1 \tag{3.39}$$

$$\text{dist}(R(C_1^T), R(C_1^{oT})) = \text{dist}\ (R(C_2^T), R(C_2^{oT})) \leq \varepsilon_1 / \delta_1 \tag{3.40}$$

And if $\delta_2 = \sigma_r^o - \sigma_{r+1} > 0$ and $\varepsilon_2 = \max (\|R_{12}\|_2 , \|R_{22}\|_2)$ with R_{12}, R_{22} defined by :

$$R_{12} = C_o V_2 - U_2 \begin{bmatrix} \Sigma_2 \\ 0 \end{bmatrix} = - \Delta C V_2 \tag{3.41}$$

$$R_{22} = C_o^T U_2 - V_2 [\Sigma_2 ; 0] = - \Delta C^T U_2 \tag{3.42}$$

then

$$\text{dist}\ (R(C_1), R(C_1^o)) \leq \varepsilon_2 / \delta_2 \tag{3.43}$$

$$\text{dist}\ (R(C_1^T), R(C_1^{oT})) = \text{dist}\ (R(C_2^T), R(C_2^{oT})) \leq \varepsilon_2 / \delta_2 \tag{3.44}$$

$$\text{dist}\ (R(C_2), R(C_2^o)) \leq \sqrt{2}\ \varepsilon_2 / \delta_2 \tag{3.45}$$

Theorem 3-5 is a slightly modified version of the generalized sin θ-theorem and the sin θ-theorem with complementary residuals in Wedin's paper [133]. It is restricted to m by p matrices, $m \geq p$ and dim $(R(C_1)$ + dim $(R(C_2)) = p$ and the 2-norm is used. Since the subspaces C_1^T and C_2^T are orthogonal complements, we can state that :

$$\text{dist}\ (R(C_2^T), R(C_2^{oT})) = \text{dist}\ (\text{Ker}\ (C_1), \text{Ker}\ (C_1^o)) = \text{dist}\ (R(C_1^T), R(C_1^{oT})) \tag{3.46}$$

and use the upper bounds (3.40) or (3.44). The subspaces C_1^o and C_2^o are not orthogonal complements. In order to estimate dist $(R(C_2), R(C_2^o))$, complementary residuals were defined to make a similar theorem possible. Notice that the **upper bounds** in theorem 3-5 **depend on the perturbation effects $\|\Delta C\|_2$ and the gap** between the least singular value of C_1 and largest singular value of C_2^o. This means that the **sensitivity of the singular subspaces grows when the perturbations are larger and when the separation between the concerned singular values narrows.** In other words, theorem 3-5 says that ε changes in C_o can alter a singular subspace by an amount ε/δ where δ measures the isolation of the involved singular values.

Theorem 3-5 has important consequences for the **perturbation of the singular subspaces related to the TLS and LS problem.** Let us first consider the TLS problem $A_o X = B_o$. Let the m by n+d matrix $[A_o ; B_o]$ be perturbed by $[\Delta A ; \Delta B]$ such that $[A ; B] = [A_o ; B_o] + [\Delta A ; \Delta B]$. Denote by r ($\leq n$) the

rank of the TLS approximation $[\hat{A}_o;\hat{B}_o]$ of $[A_o;B_o]$. Let (0.3) be the SVD of $[A;B]$ and $[A_o;B_o] = \sum_{i=1}^{n+d} \sigma_i^o u_i^o v_i^{oT}$. Applying theorem 3-5 to $[A_o;B_o]$ and $[A;B]$, we obtain :

$$\text{dist}\,(R(\hat{A}), R(\hat{A}_o)) =$$

$$\text{dist}\,(R([\hat{A};\hat{B}]), R([\hat{A}_o;\hat{B}_o])) \leq \|[\Delta A;\Delta B]\|_2 / (\sigma_r - \sigma^o_{r+1}) \qquad (3.47)$$

$$\text{dist}\,(R([\hat{A};\hat{B}]^T), R([\hat{A}_o;\hat{B}_o]^T)) =$$

$$\text{dist}\,(\text{Ker}\,([\hat{A};\hat{B}]), \text{Ker}\,([\hat{A}_o;\hat{B}_o])) \leq \|[\Delta A;\Delta B]\|_2 / (\sigma_r - \sigma^o_{r+1}) \qquad (3.48)$$

For **zero residual** problems, $A_o X = B$ represents an **exact** relationship. In this case, $[\hat{A}_o;\hat{B}_o] = [A_o;B_o]$ and thus $\sigma^o_{r+1} = 0$ in (3.47) and (3.48). Mostly, we have **rank** $(A_o) = n$. Then (3.47) and (3.48) reduce to :

$$\text{dist}\,(R(\hat{A}), R(A_o)) \leq \|[\Delta A;\Delta B]\|_2 / \sigma_n \qquad (3.49)$$

$$\text{dist}\,(\text{Ker}\,[\hat{A};\hat{B}], \text{Ker}\,[A_o;B_o]) \leq \|[\Delta A;\Delta B]\|_2 / \sigma_n \qquad (3.50)$$

This means that **the TLS solution space approaches the exact solution if the perturbations $\|[\Delta A;\Delta B]\|_2$ are kept small and if the gap** between the r-th and (r+1)-th singular value of $[A;B]$ **is large.**

Analogously we can use theorem 3-5 for **LS problems.** Let the m by n matrix A_o of rank r ($r < n$) be perturbed by ΔA such that $A = A_o + \Delta A$. Let (0.2) be the SVD of A and $A_o = \sum_{i=1}^{r} \sigma_i^{o'} u_i^{o'} v_i^{o'T}$. Applying theorem 3-5 to A_o and A, we obtain :

$$\text{dist}\,(R(A), R(A_o)) \leq \|\Delta A\|_2 / \sigma'_r \qquad (3.51)$$

$$\text{dist}\,(R(A^T), R(A_o^T)) \leq \|\Delta A\|_2 / \sigma'_r \qquad (3.52)$$

Mostly, rank $(A_o) = n$. Then (3.51) reduces to

$$\text{dist}\,(R(A), R(A_o)) \leq \|\Delta A\|_2 / \sigma'_n \qquad (3.53)$$

If the matrix entries in A_o and B_o are equally perturbed, then $\|\Delta A\|_2 \approx \|[\Delta A;\Delta B]\|_2$.

Comparing now the TLS and LS subspace sensitivity from the expressions above, some important observations can be made :

- Based on the interlacing property for singular values (see theorem A.3), the **TLS fundamental subspaces are less noise sensitive than the LS subspaces for zero residual problems.** This implies that the former are expected to be "closer" to their corresponding unperturbed subspaces. Hence, the TLS solution is expected to be more accurate than the LS solution. This is also the case **for nonzero residual** problems when $\sigma_r - \sigma^o_{r+1} > \sigma'_r$ in (3.47) and (3.51). Or in other words, the corresponding unperturbed problem $A_o X = B_o$ **must be sufficiently compatible.** This is **almost the case.** Indeed, problems for which $\sigma_r - \sigma^o_{r+1} \leq \sigma'_r$ are usually irrelevant from a modelling point of view and must be rejected.
- **Since the difference in sensitivity depends on the ratio $(\sigma_r - \sigma^o_{r+1})/\sigma'_r$ the superiority of TLS w.r.t. LS will be more pronounced with increasing ratio.** This ratio is mainly influenced by the length (Frobenius norm), the column dimension and the orientation of the observation matrix B_o (or equivalently, of the solution X) w.r.t. the singular vectors of A_o. Those observations will be confirmed by simulations in section 3.4.

Furthermore, also interesting **upper bounds for the perturbation of singular vectors,** associated with **isolated** singular values, can be deduced from theorem 3-5. Just partition C_o and C such that dim (C_2) = dim (C_2^o) = 1.

Corollary 3-6 :

Let the SVD of C_o and C be given by (3.32) and (3.33) respectively. If $\delta = \min\,(\sigma_i, \min_{j \neq i} |\sigma_i - \sigma_j^o|) > 0$ and $\varepsilon = \max\,(\|r_{12}\|_2, \|r_{22}\|_2)$ with r_{12}, r_{22} defined by

$$r_{12} = C_o v_i - \sigma_i u_i = - \Delta C\, v_i$$

$$r_{22} = C_o^T u_{i2} - \sigma_i v_i\,[1, 0, \ldots, 0] = - \Delta C^T u_{i2}$$

with $U_{i2} = [u_i, u_{p+1}, \ldots, u_m]$

then we have ($\measuredangle$ means angle) :

$$|\sin \measuredangle (v_i, v_i^o)| \leq \varepsilon / \delta \tag{3.54}$$

$$|\sin \measuredangle (u_i, u_i^o)| \leq \sqrt{2}\ \varepsilon / \delta \tag{3.55}$$

and $\varepsilon \leq \|\Delta C\|_2$

Observe that without a gap singular vectors can be very sensitive functions of the data. This suggests that the proper object to compute, when a few singular values are **tightly clustered**, is any orthonormal basis for the invariant subspace belonging to the whole cluster rather than separate singular vectors for each singular value. Furthermore, corollary 3-6 clearly shows that **the singular vectors associated with the smallest singular values close to zero are severely affected by perturbations.** Hence, it reduces the accuracy of the solution of (0.1), especially, if B_o is closer to the lowest singular vectors $u_n^{o'}$, $u_{n-1}^{o'}$, ... of A_o. The **worst case** happens if $B_o \mathbin{/\!/} u_n^{o'}$. In this case $\sigma_n^o = \sqrt{\sigma_n^{o'2} + \|B_o\|_F^2}$ if $A_o X = B$ holds exactly. Then the ratio $\sigma_n^o/\sigma_n^{o'}$ reaches its maximal value for a given $\|B_o\|_F$, corresponding to theorem 2-15. Since the sensitivity-to-noise of the n-th right singular vector is inversely proportional to its singular value from corollary 3-6, the **superiority of TLS** w.r.t. LS is **maximal** in this case. This is confirmed by the simulations in section 3.4.

Finally, let us now investigate **subspace perturbations which maximize the sensitivity of the TLS problem.** From chapter 1, we know that the TLS solution of (0.1) is entirely deduced from the right singular subspace, associated with the smallest singular values of the matrix $[A;B]$. Hence, the worst perturbations for TLS occur by rotating away this subspace as far as possible without affecting the others. Let us first consider simultaneous left and right singular subspace rotations.
The largest angle z over which two singular vectors in the left and right singular subspace can rotate simultaneously within their own span, is given in the next theorem :

Theorem 3-7 : **Left and right singular subspace rotation of maximal sensitivity.**

Let the SVD of C_o be given by (3.32). The largest angle z over which a pair of right singular vectors v_i^o and v_j^o, and left singular vectors u_i^o and u_j^o of C_o can rotate simultaneously for a fixed variation $\|\Delta C\|_F$ is given by

$$z = \arcsin \left(\|\Delta C\|_F \;/\; (\sqrt{2}\ |\sigma_i^o - \sigma_j^o|) \right) \tag{3.56}$$

The proof is given in [105].
In fact, we obtain a maximal forward effect on the matrices U_o and V_o, for a minimal effect on C_o. Therefore, we use all the energy of ΔC to rotate away as far as possible one pair of singular vectors, without affecting the others. It is clear from (3.56) that bending a singular vector in the direction of another singular vector is much easier when the involved singular values are close.
W.r.t. the accuracy of the TLS problem, the **left and right singular subspace perturbation with worst sensitivity** is obtained when applying theorem 3-7 to the singular vectors associated with σ_n^o and σ_{n+1}^o of $[A_o;B_o]$ or ($\measuredangle$ means angle) :

$$z = \max \left\{ \gamma \,|\, \gamma = \measuredangle (u_{n+1}^o, u_{n+1}) = \measuredangle (v_{n+1}^o, v_{n+1}) \right\} = \arcsin \frac{\|[\Delta A;\Delta B]\|_F}{\sqrt{2}|\sigma_n^o - \sigma_{n+1}^o|} \tag{3.57}$$

If we allow $\measuredangle (u_{n+1}^o, u_{n+1})$ to be **different** from $\measuredangle (v_{n+1}^o, v_{n+1})$, then **worse** perturbations than (3.57) w.r.t. the accuracy of the TLS solution are possible. Indeed, since the TLS solution is computed from the right singular subspace of C_o only, we can concentrate the variation ΔC on C_o entirely in the rotation of the two right singular vectors associated with the singular values σ_n^o and σ_{n+1}^o. The maximal angle z over which these vectors can rotate within their own span for a given $\|\Delta C\|_F$, and without affecting the corresponding left singular vectors, is derived in the following theorem.

Theorem 3-8 :

Let the SVD of C_o be given by (3.32). The largest angle z over which a pair of right singular vectors v_i^o and v_j^o of C_o can rotate for a fixed variation $\|\Delta C\|_F$, without affecting the corresponding left singular vectors u_i^o and u_j^o, is given by

$$z = 2 \arcsin \left(\|\Delta C\|_F \; / \; (2\sqrt{\sigma_i^{o2} + \sigma_j^{o2}}) \right) \tag{3.58}$$

The proof is given in [122].

For a given $\|\Delta C\|_F$, the angle z will be maximal for the smallest singular values. Taking j = n and i = n+1 and assuming $\sigma_{n+1}^o = 0$, one obtains

$$z = 2 \arcsin \left(\|\Delta C\|_F \; / \; 2\,\sigma_n^o \right) \tag{3.59}$$

(3.59) is the **worst subspace perturbation for zero residual TLS problems, i.e. the accuracy** of the TLS solution is **maximally affected.** However if $\sigma_{n+1}^o \neq 0$, it is no longer evident which subspace perturbation is worst w.r.t. the TLS solution accuracy. Indeed, if σ_{n+1}^o approaches σ_n^o, the angle given in (3.57) may be much larger than the angle given in (3.58). For each fixed variation $\|\Delta C\|_F$, there must exist an angle $\measuredangle(u_{n+1}^o, u_{n+1})$ and an angle $\measuredangle(v_{n+1}^o, v_{n+1})$ such that the accuracy of the TLS solution is most affected. This problem has not yet been solved.

In any case the **perturbation of the singular vector** v_{n+1} (from which the solution of the one-dimensional TLS problem is computed), can be **related to the perturbation of the TLS solution** $\hat{x}$, as proven in the following theorem :

Theorem 3-9 :

Let σ_{n+1}^o and σ_{n+1} be the minimal singular values of $[A_o; b_o]$ and $[A;b]$ respectively and v_{n+1}^o and v_{n+1} their respective corresponding right singular vectors.

Let σ_n' be the minimal singular value of A. Let x and $\hat{x}$ be the solution of the (un)perturbed TLS problem $A_o x = b_o$ and $Ax = b$ respectively.

If $x^T \hat{x} \geqslant 0$, then $\cos \measuredangle(x, \hat{x}) = \dfrac{\sqrt{(\|x\|_2^2 + 1)(\|\hat{x}\|_2^2 + 1)}\, \cos \measuredangle(v_{n+1}^o, v_{n+1}) - 1}{\|x\|_2 \; \|\hat{x}\|_2}$ (3.60)

Proof :

From (1.16) we obtain using the cos-formula and assuming $x^T \hat{x} > 0$

$$\cos \measuredangle \left(\begin{bmatrix} x \\ -1 \end{bmatrix}, \begin{bmatrix} \hat{x} \\ -1 \end{bmatrix} \right) = \frac{x^T \hat{x} + 1}{\sqrt{\|x\|_2^2 + 1}\; \sqrt{\|\hat{x}\|_2^2 + 1}} =$$

$$\cos \measuredangle(v_{n+1}^o, v_{n+1}) = v_{n+1}^T v_{n+1}^o \tag{3.61}$$

Substituting $\cos \measuredangle(x, \hat{x}) = \dfrac{x^T \hat{x}}{\|x\|_2 \; \|\hat{x}\|_2}$ into (3.61) gives (3.60). ■

Theorem 3-9 has important consequences. If $\|x\|_2 = \|\hat{x}\|_2$, theorem 3-9 reduces to :

$$\cos \measuredangle(x, \hat{x}) = \cos \measuredangle(v_{n+1}^o, v_{n+1}) - \frac{1 - \cos \measuredangle(v_{n+1}^o, v_{n+1})}{\|x\|_2^2} \tag{3.62}$$

This means that, if $\|x\|_2 = \|\hat{x}\|_2$, **the angle between x and $\hat{x}$ approaches the angle between v_{n+1}^o and v_{n+1} more and more with increasing $\|x\|_2$ and smaller angle $\measuredangle(v_{n+1}^o, v_{n+1})$ but remains always larger**

e.g. if $\|x\|_2 = 1$, then $\measuredangle(x, \hat{x}) = 43°$ if $\measuredangle(v_{n+1}^o, v_{n+1}) = 30°$

and $\measuredangle(x, \hat{x}) = 14.16°$ if $\measuredangle(v_{n+1}^o, v_{n+1}) = 10°$

and $\measuredangle(x, \hat{x}) = 90°$ if $\measuredangle(v_{n+1}^o, v_{n+1}) = 60°$,

i.e. $x \perp \hat{x}$ and the solution error is 100 %.

If $\|x\|_2 \neq \|\hat{x}\|_2$, we have :

$$\cos \measuredangle(x, \hat{x}) \begin{cases} < \text{ expression (3.62) if } \|x\|_2 > \|\hat{x}\|_2 & (3.63) \\ > \text{ expression (3.62) if } \|x\|_2 < \|\hat{x}\|_2 & (3.64) \end{cases}$$

This implies that for a given $\measuredangle(v_{n+1}^o, v_{n+1})$, the deviations between the angle $\measuredangle(x, \hat{x})$ and the angle $\measuredangle(v_{n+1}^o, v_{n+1})$ decrease (resp., increase) w.r.t. the case $\|x\|_2 = \|\hat{x}\|_2$, if the length of the perturbed solution $\hat{x}$ is larger (resp., smaller) than the length of the unperturbed solution x.

3.4. SIMULATION RESULTS

In this section we describe a set of simulations confirming the main results of the sensitivity analysis, discussed in the previous sections. The **expected differences in accuracy between the TLS and LS solution in the presence of perturbations** are shown. Only **zero residual** problems are considered.

SIMULATION PROCEDURE

We start from an exact, overdetermined set of m equations $A_o X = B_o$ in n x d unknowns X with d observation vectors B_o and A_o of full rank. Random errors Δa_{ij} and Δb_{ij} with zero mean and variance σ_v^2 are then added to the entries of the exact data matrix $[A_o;B_o]$. Call the perturbed matrix $[A;B] = [A_o;B_o] + [\Delta A;\Delta B]$. We then solve the perturbed set of equations AX = B to obtain the TLS solution $\hat{X}$ and the LS solution X'. The relative errors

$$\|X' - X\|_F / \|X\|_F \quad \text{and} \quad \|\hat{X} - X\|_F / \|X\|_F \tag{3.65}$$

are averaged over 100 sets of equations affected by random errors with zero mean and equal variance σ_v^2. We repeat the simulation for progressively larger noise variances σ_v^2. The average relative errors are then plotted versus the noise variance σ_v^2 (see fig. 3.4, 3.5 and 3.6).

Let (0.2) (resp., (0.3)) be the SVD of A_o (resp., $[A_o;B_o]$). Since $B_o \in R(A_o)$, it follows that $\sigma_{n+i} = 0$, i = 1, ..., d. From the theorems above we can deduce a number of parameters which influence the relative capabilities of LS and TLS to reject the noise from the data.
First of all from the theorems 3-5 and 3-6 and since $\sigma_{n+1} = 0$, it is clear that **the difference in accuracy depends on the ratio σ_n/σ'_n of the n-th singular values σ_n and σ'_n of $[A_o;B_o]$ and A_o respectively.** By increasing this ratio, the TLS singular subspaces become relatively less sensitive w.r.t. the LS singular subspaces and thus the superiority of TLS w.r.t. LS will be more pronounced. Let us **investigate the parameters which influence this ratio.**

- First, **the orientation of each observation vector b_o, or equivalently the orientation of each solution x w.r.t. the vectors u'_n or v'_n of A_o respectively,** plays an important role. From the interlacing property for singular values, it is evident that the singular values of $[A_o;b_o]$ are always larger than the corresponding ones of A_o, i.e. $\sigma_i \geq \sigma'_i$ i = 1, ..., n. However the extent to which the σ_i will be larger, depends on the orientation of b_o w.r.t. the u'_i of A_o. It is not difficult to see that those directions u'_i of A_o onto which b_o has a larger projection, will be **"favored"** and have the largest ratio σ_i/σ'_i. From this, it is evident that **σ_n/σ'_n will be maximal if b_o lies entirely in the direction of u'_n.** In this case however, the accuracy of the LS solution is also mostly affected. Indeed, if $b_o = \sum_{i=1}^{n} \beta_i u'_i$, the exact solution $x = \sum_{i=1}^{n} (\beta_i/\sigma'_i)\, v'_i$. By adding noise the smaller singular values of A_o and their singular vectors are mostly affected, according to corollary 3-6. Hence it reduces the accuracy of the solution especially if x is "closer" to the lowest singular vectors v'_n, v'_{n-1}, ... This happens not only if b_o is "close" to the lowest singular vectors u'_n, u'_{n-1}, ... but also if the **condition number σ'_1/σ'_n of A_o is large** and the projection β_n of b_o onto u'_n is not too small since then, the component β_n/σ'_n of x in the direction of v'_n is large. Hence, x will belong to the space spanned by v'_n, v'_{n-1}, ... (see also fig. 3.6 further on). The worst case happens of course when $b_o = \beta_n u'_n$. Then σ_n/σ'_n also reaches its maximal value for a given $\|b_o\|_2$, according to theorem 2-18. This is easily seen if we compute the SVD of $[A_o;b_o]$ with $b_o \,/\!/\, u'_n$. Let the SVD of A_o and $[A_o;B_o]$ be given by (0.2) and (0.3) respectively and choose the j-th column of B_o as $(b_o)_j = \|(b_o)_j\|_2 \, u'_i$ for j = 1, ..., d, then

  * the left singular vectors of $[A_o;B_o]$: $u_j = u'_j$ j = 1, ..., n
  * the singular values of $[A_o;B_o]$: $\sigma_j = \sigma'_j$ j = 1, ..., i - 1, i + 1, ..., n

    $\sigma_i = \sqrt{\sigma_i'^2 + \|B_o\|_F^2}$

    $\sigma_{n+j} = 0$ j = 1, ..., d
  * the right singular vectors of $[A_o;B_o]$:

$$v_j = [v_j'^T,\ 0,\ \ldots,\ 0]^T \qquad j = 1,\ \ldots,\ i-1,\ i+1,\ \ldots,\ n$$

$$v_i = [(\sigma_i'/\sigma_i)\ v_i'^T,\ \|(b_o)_1\|_2 / \sigma_i,\ \ldots,\ \|(b_o)_d\|_2 / \sigma_i]^T$$

$$v_{n+j} = [\,(-\|(b_o)_j\|_2 / \sqrt{\sigma_i'^2 + \|(b_o)_j\|_2^2})\ v_i'^T,\ 0,\ \ldots,$$

$$0,\ \sigma_i' / \sqrt{\sigma_i'^2 + \|(b_o)_j\|_2^2},\ 0,\ \ldots,\ 0]^T \qquad j = 1,\ \ldots,\ d$$

$$\uparrow$$

$$(n+j)\text{-th component} \qquad (3.66)$$

From (3.66) we see that for $b = \|b\|\ u_n'$, the ratio $\bar{\sigma}_n/\sigma_n' = \sqrt{\sigma_n'^2 + \|b\|_2^2} / \sigma_n'$ is indeed maximal. Since the sensitivity to noise of the n-th singular vector is inversely proportional to its corresponding singular value according to corollary 3-6, v_n will be less noise sensitive than v_n' with growing ratio σ_n/σ_n' and hence the superiority of TLS w.r.t. LS will be more pronounced. This is illustrated in fig. 3.4.

- Second, **the length of each observation vector** b_o influences the ratio σ_n/σ_n'. This is evident from (3.66). Making $\|b_o\|_2$ larger, increases the ratio σ_n/σ_n' if $b_o \,/\!/\, u_n'$ (as long as $\sigma_{n-1} < \sqrt{\sigma_n'^2 + \|b_o\|^2}$) and hence, the superiority of TLS w.r.t. LS will be more pronounced, as shown in fig. 3.5.

- Third, the ratio σ_n/σ_n' depends on the **number d of observation vectors** B_o in (0.1). From the previous discussion it is evident that the orientation of B_o, or equivalently the orientation of X, is important. If now all d vectors of B_o (instead of one) are close to the lower singular vectors u_n', u_{n-1}', ... of A_o, one immediately sees that the singular values σ_j of $[A_o;B_o]$, associated with directions u_j which are close to u_n', u_{n-1}', ..., are growing. In the extreme case where $B_o \,/\!/\, u_n'$, σ_n equals $\sqrt{\sigma_n'^2 + \|B_o\|_F^2}$ from (3.66) (as long as $\sigma_{n-1} < \sqrt{\sigma_n'^2 + \|B_o\|_F^2}$) and the ratio σ_n/σ_n' is maximal. Since d observation vectors together have a larger norm than one vector, the ratio σ_n/σ_n' is growing with increasing number d of observation vectors. The relative effect on the solution accuracy of TLS w.r.t. LS is most pronounced, not only if all observation vectors are equally oriented along the lowest singular vector u_n' of A_o, but also if the condition number of A_o is large and the projection of B_o onto u_n' is not too small, as illustrated in fig. 3.6. Observe that the accuracy of the TLS solution improves with increasing number d of observation vectors while the accuracy of LS is independent of d !

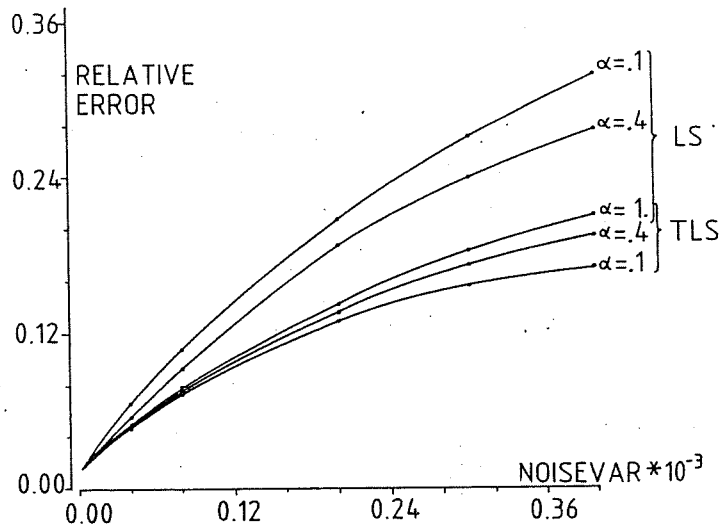


Fig. 3.4 : Comparison of the average relative error of the LS and TLS solution versus the variance of the noise added to the set $(A_o)_{18x5}\ x = b_o$ with $\sigma'(A_o) = \{10, 8, 4, 1, .1\}$ and $b_o = \sum_{i=1}^{5} \beta_i\ u_i'(A_o)$ **for different coefficient ratios** $\alpha = \max_{i=1,4} (\beta_i/\sigma_i') / (\beta_5/\sigma_5')$. The values 1, .4 and .1 of α correspond with a ratio $\sigma_5[A_o;b_o] / \sigma_5'(A_o)$ equal to 1.1, 1.6 and 3.6 respectively.

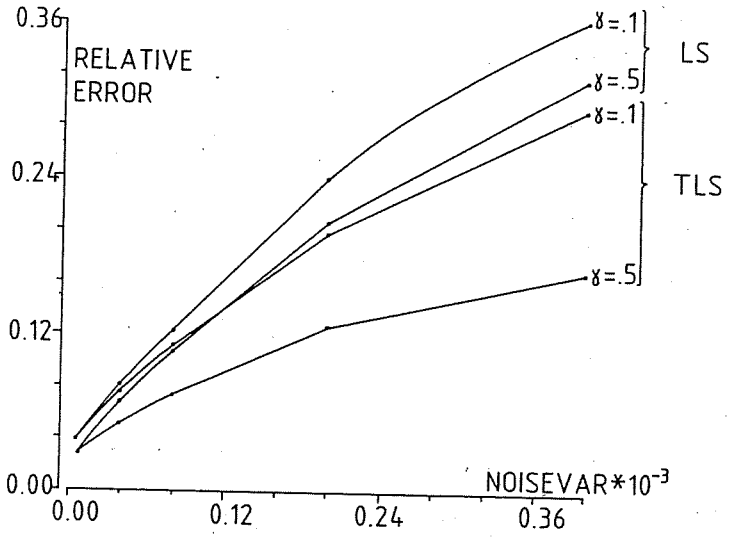


Fig. 3.5 : Comparison of the average relative error of the LS and TLS solution versus the variance of the noise added to the set $(A_o)_{18x5}\ x = b_o$ with $\sigma'(A_o) = \{10, 8, 4, 1, .1\}$ and $b_o = \gamma u'_5$ **for different lengths γ of b_o.** The values .1 and .5 of γ correspond with a ratio $\sigma_5[A_o;b_o] \ / \ \sigma'_5\ (A_o)$ equal to 1.4. and 5.1 respectively.

Furthermore, **the difference in accuracy between TLS and LS also depends on the number m of equations,** i.e. the degree of overdetermination. This will be statistically proven in chapter 4 and confirmed by simulations (see fig. 4.1).

Note finally that, although the TLS solution error may equal the LS solution error for a particular set (0.1) (see section 3.2), this is generally not the case. The **TLS solution is expected to have better accuracy** than LS especially when $b_o \ /\!/ \ u'_n$ and all data are equally perturbed.

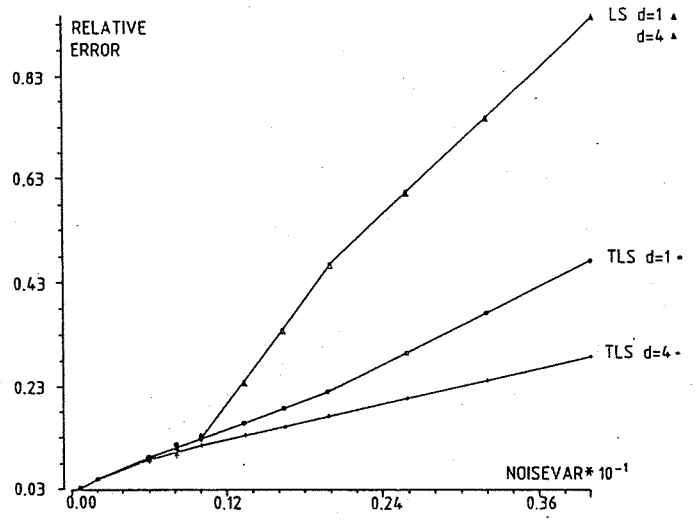


Fig. 3.6. : Comparison of the average relative error of the LS and TLS solution versus the variance of the noise added to the set $(A_o)_{14x4}\ X = B_o$ with $\sigma'(A_o) = \{264, 9.7, 2.7, .75\}$ **for different numbers d of observation vectors.** All d columns of B_o are equal to $\sum_{i=1}^{4} u'_i\ (A_o)$. The values 1 and 4 of d correspond with ratio $\sigma_4\ [A_o;B_o] \ / \ \sigma'_4\ (A_o)$ equal to 1.6 and 2.2 respectively.

We repeat that the upper bounds given in section 3.2, are only realistic for the worst type of perturbation $[\Delta A;\Delta B]$ of the data and the worst orientation of B_o. They are rather pessimistic if more information about the perturbations is available (see chapter 4).

3.5. CONCLUDING REMARKS

In this chapter, a comparison is presented of the **sensitivity** of the TLS problem w.r.t. the LS problem when all data are perturbed. This is done by evaluating the relative changes in their solutions and related subspaces w.r.t. the **unperturbed** problem $A_o x = b_o$, due to errors in all data.

First of all, the **worst perturbation effects caused in their solutions** are investigated for zero and nonzero residual problems (section 3.2). Hereby, the **condition number** can be used as a good tool to quantify the worst relative error effects on the solution (section 3.2.1). The importance of this notion of worst perturbation is that in the absence of further information, the worst perturbation should be considered as an attainable perturbation of the solution. Upper bounds for the absolute and relative errors are given and it is shown that they are maximized for different conditions of perturbation and orientation of the data (section 3.2.2). In case of **zero residual** problems the TLS and LS solution perturbations attain **the same upper bounds** under the same conditions but for nonzero residual problems the **perturbation effects on the TLS solution** may be even **worse** (section 3.2.3). However it must be stressed that those worst perturbations are only attained for a special choice of b_o and the perturbations on the data. They are quite unrealistic for other conditions.

Second, the **sensitivity properties of the SVD** are investigated (section 3.3). The perturbation results for singular values and their associated singular subspaces allow to measure for the TLS and LS problem the **maximal distance** between their perturbed singular subspaces and the corresponding ones of the unperturbed problem. It is concluded that the TLS subspaces are **less noise sensitive** than the LS subspaces. This is always the case for any zero residual problem and sufficiently compatible nonzero residual problems. This **superiority** of TLS w.r.t. LS is **more pronounced** if the **number of observation vectors** B_o and their **length** is increasing but especially if B_o, or equivalently the solution X, is **"close"** to the singular vectors of A_o, associated with its **smallest** singular values. For those cases it is **expected that the accuracy** of the

TLS solution w.r.t. LS **is maximal**, as **confirmed by our simulations** (section 3.4).

Furthermore, subspace **perturbations** which **maximally affect the accuracy of the TLS problem** are studied by computing the largest possible angle over which subspaces can rotate. Also, the effect of singular subspace perturbations on the TLS solution accuracy is derived for the one-dimensional problem (section 3.3).

CHAPTER 4

THE STATISTICAL PROPERTIES OF THE TLS PROBLEM

4.1. INTRODUCTION : THE ERRORS-IN-VARIABLES MODEL

In the previous chapters, the TLS technique is introduced and its algebraic and geometrical properties are presented from the viewpoint of a numerical analyst. No assumptions are made about the distribution of the errors added to the data. **If now some information about the error distribution is available**, e.g. its covariance matrix, then statistical properties of the solution can be derived. They prove whether a certain solution technique is **expected** to be better and more accurate than others. In this chapter, it is shown that, **if all errors on the data in (0.1) are rowwise independently and identically distributed with zero mean and common covariance matrix of the form $\sigma_v^2 I$, then the TLS technique offers the best estimate** and is superior to the LS solution. This will be statistically proven and confirmed by simulations. As we shall see in section 4.5, these assumptions are not too restrictive.

In this case of noise superposition, **the statistical properties** in this chapter clearly **show the reader the expected quality of the TLS solution** of (0.1) as estimator of the unknown parameters of a linear relation (see e.g. (I.1)) present in the data, for any number m of observations. Note however that contrary to chapter 3, these properties don't inform the reader about the quality of the TLS estimator for a **particular** choice of the error matrix. Instead, they **judge the merit** of the TLS solution, as parameter estimator, **by the distribution of estimates** to which it gives rise (see appendix B). Additionally, estimators of other parameters in the model such as the error variance σ_v^2 or an intercept, are derived from the TLS results and provide very valuable information to the user. Moreover, **all the properties** given in this chapter, **also apply to problems whose common error covariance matrix has the more general form $\sigma_v^2 \Sigma_o$**, Σ_o known and positive definite. As shown in section 4.5, the data can always be transformed appropriately in this case.

The statistical properties of a solution technique, called an **estimator**, are always **related to a model** whose parameters have to be estimated. In order to appreciate the statistical properties of the TLS problem we must first of all derive its most suitable model. The most successful approach directly related to the TLS concept, is the **"Errors-in-Variables" model**, which considers all observations as coming from some unknown true values plus measurement errors. Knowledge of the error covariance matrix (or an estimate of it) is required.

It is well-known that the presence of errors in all variables in univariate linear regression makes the ordinary LS estimators inconsistent and biased. An extensive literature dating back to Adcock [3], deals with estimation in models of univariate regression which incorporate "errors in variables" (see introduction). **Less is known about the estimation in multivariate errors-in-variables models.** Gleser and Watson [42] presented results for general linear transformations. Using an eigenvalue-eigenvector analysis, Gleser [41] also derived various estimators of the parameters in a linear multivariate errors-in-variables model and proved many statistical properties of them. From their close relation to the TLS problem, we could derive most of the results, stated in this chapter.

The multivariate linear errors-in-variables model can be described as follows : a vector $(b_o)_i$ of unobservable values of d dependent variables and a vector $(a_o)_i$ of unobservable values of n independent variables generated by some phenomenon of interest for i = 1, ..., m, are related by the linear model :

$$(b_o)_i = \alpha + X^T (a_o)_i \qquad i = 1, \ldots, m \tag{4.1}$$

The d-dimensional vector α and the n by d matrix X are the true, unknown parameters to be estimated. α is the vector of intercepts and X is the matrix of slopes. Actually one can only observe $(b_o)_i$ and $(a_o)_i$ with errors, i.e.

$$a_i = (a_o)_i + \Delta a_i$$

$$b_i = (b_o)_i + \Delta b_i \quad i = 1, \ldots, m \tag{4.2}$$

$\Delta_i = \begin{bmatrix} \Delta a_i \\ \Delta b_i \end{bmatrix}$ is a vector of random errors with known covariance matrix $\Sigma_\Delta = E(\Delta_i \Delta_i^T)$. To condense the notation, let A, A_o and ΔA be the m by n matrices whose rows are a_i, $(a_o)_i$ and Δa_i, i = 1, ..., m respectively. Let B, B_o and ΔB be the m by d matrices whose rows are b_i, $(b_o)_i$ and Δb_i, i = 1, ... m respectively. In terms of these matrices, (4.1) and (4.2) become

$$A = A_o + \Delta A$$

$$B = B_o + \Delta B$$

$$B_o = 1_m \alpha^T + A_o X \tag{4.3}$$

where $1_m = [1, \ldots, 1]^T$ and the row vectors of $\Delta = [\Delta A; \Delta B]$ have common covariance matrix Σ_Δ. The intercept vector α is either known to be zero (**no-intercept model**) or else is unknown (**intercept model**) and must be estimated.

In the "**classical**" errors-in-variables model (4.3), an exact but unobservable relation is assumed. Hence, (4.3) describes a zero residual problem (see section 3.2.1). Nonzero residual problems are described by the more **general** error-in-variables regression model, which introduce an error ε in the equation (4.1) (see section 5.2). Consider now the additional assumption :

Assumption 4-1 :

The rows of $\Delta = [\Delta A; \Delta B]$ in (4.3) are independently and identically distributed with common zero mean vector and common covariance matrix $\Sigma_\Delta = \sigma_v^2 I_{n+d}$ where $\sigma_v^2 > 0$ is unknown.

It is shown in the following sections that the **TLS technique is able to estimate the unknown parameters X, A_o, α and σ_v of the errors-in-variables model (4.3) consistently.** The reader is referred to appendix B for the definition of the elementary statistical concepts, used throughout this chapter.

In section 4.2 estimators $\hat{X}$, $\hat{A}$, $\hat{\alpha}$ and $\hat{\sigma}$ of the parameters X, A_o, α and σ_v are derived from the TLS results. In section 4.3, it is shown that the TLS estimators $\hat{X}$ and $\hat{\alpha}$ are **strongly consistent estimators** of X and σ_v respectively in both intercept and no-intercept models. In the intercept model, $\hat{\alpha}$ is also a strongly consistent estimator of α. The strong consistency results of section 4.3 are obtained without any assumptions about the distribution of Δ beyond those stated in assumption 4-1. In particular, it is not assumed that the rows of Δ have a multivariate normal distribution. Strong consistency results (which assert convergence with probability one) provide deeper knowledge of the large-sample properties of estimators than do weak consistency results (which assert only convergence in probability) [41]. Section 4.4 considers **asymptotic distributional** results for $\sqrt{m}(\hat{X} - X)$ and $\sqrt{m}(\hat{\sigma}^2 - \sigma_v^2)$ in both the intercept and no-intercept models, and for $\sqrt{m}(\hat{\alpha} - \alpha)$ in the intercept model. If the common distribution of the rows of Δ has finite fourth moments, these quantities have asymptotic multivariate normal distributions. Finally, section 4.5 discusses the **practical relevance** of the assumptions needed to carry through the asymptotic theory, and presents some simulation results. It is shown that when $\Sigma_\Delta = \sigma_v^2 \Sigma_o$ with Σ_o known and positive definite, then the data can be transformed linearly so that (4.3) and assumption 4-1 hold for the transformed data.

4.2. PARAMETER ESTIMATORS

It is well-known in regression analysis that the **ordinary LS solution** x' of (4.3) with one observation vector (d = 1) **is an inconsistent estimator** of the true parameters x. This implies that LS is asymptotically **biased.** If the error covariance matrix Σ_Δ is known, the asymptotic bias can be removed and a consistent estimator, called **corrected least squares (CLS)**, can be derived [96-65]. We can easily show that **TLS is no more than a special case of CLS estimation.** Indeed, it was proven in [124] that, under the assumption 4-1, CLS and TLS yield the same consistent estimator $\hat{x}$ of the parameters x in (4.3). Using the same assumption, Gleser [41] proved consistency of the TLS solution $\hat{X}$ for the multi-dimensional problem (d > 1), as well as consistency of the estimators of other parameters α and σ_v in (4.3) (see section 4.3). These estimators are derived from the TLS results, as shown below.

Because the analyses of the **no-intercept and intercept** form of the model (4.3) involve similar steps, these two forms can be **treated simultaneously**. To this end, we premultiply the data $[A;B]$ with a matrix C_m, given by :

$$C_m = I_m \qquad \text{for the no-intercept model}$$
$$= I_m - m^{-1} 1_m 1_m^T \qquad \text{for the intercept model with } 1_m = [1, \ldots, 1]^T \tag{4.4}$$

This means that the **variables of the intercept model are centered.** This is done by subtracting from each entry in A and B the arithmetic mean of its corresponding column. The columns of the transformed data $C_m[A;B]$ have then zero mean. Compute now the SVD (0.3) of $C_m[A;B]$ and assume that V_{22} and V_{11} are nonsingular. We then consider the following estimators $\hat{X}$, $\hat{A}$, $\hat{\alpha}$ and $\hat{\sigma}$ of the respective parameters X, A_o, α and σ_v in (4.3) and assumption 4-1 :

- $\hat{X}$ is the TLS solution given by (1.25) or (1.27).
- $\hat{A}$ is the TLS approximation of A, given by $U_1 \Sigma_1 V_{11}^T$ in (1.21).
- $\hat{\alpha}^T = 0$ for the no-intercept model
  $= [\bar{b}_1, \ldots, \bar{b}_d] - [\bar{a}_1, \ldots, \bar{a}_n]\ \hat{X}$ for the intercept model
  $\bar{a}_i$ and $\bar{b}_i$ denote the arithmetic means of the i-th column in A and B respectively. (4.5)
- $\hat{\sigma}^2 = (md)^{-1} \|\Sigma_2\|_F^2$ (4.6)

It is interesting to note that the estimators $\hat{X}$ and $\hat{\alpha}$ of the intercept model can be computed straightforwardly from the original data $[A;B]$ (no centering is required). Indeed, $\begin{bmatrix}\hat{\alpha}\\ \hat{X}\end{bmatrix}$ is the TLS solution of the following set :

$$\left[\begin{array}{c|ccc} 0 & 0 & \ldots & 0 \\ \hline 1_m & & [A;B] & \end{array}\right] \begin{bmatrix}\alpha\\ X\\ -I_d\end{bmatrix} = 0 \tag{4.7}$$

Since the first column is exactly known, we must apply algorithm 1-2 (see section 1.8.2). The QR factorization (1.124) of (4.7) yields :

$$\left[\begin{array}{c|ccc} -\sqrt{m} & & -(\sqrt{m})^{-1} 1_m^T [A;B] & \\ \hline 0 & & & \\ \vdots & & & \\ 0 & & C_m [A;B] & \end{array}\right] \begin{bmatrix}\alpha\\ X\\ -I_d\end{bmatrix} = 0 \tag{4.8}$$

and thus $\hat{X}$ is the TLS solution of $C_m[A;B][X^T;-I_d]^T = 0$ while $\hat{\alpha}$, given by $m^{-1} 1_m^T [A;B][-\hat{X}^T;I_d]$, equals (4.5).

Contrary to our algebraic approach (see theorem 1-1), Gleser [41] used a statistical approach to prove under which conditions V_{22} is nonsingular. Very interesting in his proofs is the following property. If the assumptions 4-1, 4-2 and 4-3 (see further on) hold, there always exists a finite m_o such that $V_{11}^{(m)}$ and $V_{22}^{(m)}$ are nonsingular. This means that the generic TLS solution $\hat{X}^{(m)} = -V_{12}^{(m)} V_{22}^{(m)-1}$ exists, for all $m \geq m_o$ (the superscript m on a sample quantity, e.g. $\hat{X}^{(m)}$, indicates that the quantity is calculated from the first m observations a_i and b_i, given by (4.2), of a process under study). Hence, the property guarantees that **a TLS solution can always be computed in the generic sense when the set of equations (0.1) is sufficiently overdetermined ($m > m_o$)** and the assumptions hold.

Furthermore, Gleser [41] proved that the estimators $\hat{X}$, $\hat{\alpha}$, $\hat{A}$ and $d\hat{\sigma}^2/(n+d)$ defined above, are the unique (with probability one) **maximum likelihood estimators** of X, α, A_o and σ_v^2 in (4.3) respectively, if $m > n+d$ and the common distribution of the rows of $[\Delta A;\Delta B]$ is multivariate normal.

4.3. STRONG CONSISTENCY PROPERTIES

In this section, strong consistency properties of the estimators $\hat{\alpha}$, $\hat{X}$ and $\hat{\sigma}^2$ of the parameters in model (4.3) are discussed. No conditions are imposed on the distribution of the rows of $\Delta = [\Delta A;\Delta B]$ beyond those appearing in assumption 4-1. Reference is made to the following assumptions on A_o given in (4.3) :

Assumption 4-2 :

$\lim_{m\to\infty} \frac{1}{m} A_o^T A_o$ exists.

Assumption 4-3 :

the matrix $\Omega = \lim_{m\to\infty} \frac{1}{m} A_o^T C_m A_o$ is positive definite.

Assumption 4-2 also implies the existence of

- an asymptotic mean $\mu = \lim_{m\to\infty} \frac{1}{m} [A_o;B_o]^T 1_m = \begin{bmatrix} \mu_1 \\ \alpha + X^T \mu_1 \end{bmatrix}$ with $\mu_1 =$

$$\lim_{m\to\infty} \frac{1}{m} A_o^T 1_m \qquad (4.9)$$

- and the $\lim_{m\to\infty} \frac{1}{m} [A_o;B_o]^T C_m [A_o;B_o] = \begin{bmatrix} I_n \\ X^T \end{bmatrix} \Omega [I_n;X]$ (4.10)

Using the notations defined above and the SVD (0.3) of $C_m[A,B]$, the following strong consistency properties can be proven (see [41] for proofs). Under the assumptions 4-1 and 4-2 :

- $\lim_{m\to\infty} \frac{1}{m} [A;B]^T C_m [A;B] = \Theta = \sigma_v^2 I_{n+d} + \lim_{m\to\infty} \frac{1}{m} [A_o;B_o]^T C_m [A_o;B_o]$

w.p. 1 (4.11)

- $\lim_{m\to\infty} \frac{1}{m} [A;B]^T 1_m = \mu$ w.p. 1 (4.12)

- $\lim_{m\to\infty} \frac{1}{m} \Sigma^T \Sigma = \text{diag}\,(\theta_1, \ldots, \theta_{n+d}) =$

$$\text{diag}\,(\sigma_v^2 + \gamma_1, \ldots, \sigma_v^2 + \gamma_n, \sigma_v^2, \ldots, \sigma_v^2) \quad \text{w.p. 1}$$

where θ_i and γ_i are the (ordered) eigenvalues of Θ in (4.11) and of $\Omega(I_n + X X^T)$ respectively. (4.13)

- $\lim_{m\to\infty} \frac{1}{m} \Sigma_2^T \Sigma_2 = \sigma_v^2 I_d$ w.p. 1 (4.14)

- $\lim_{m\to\infty} \hat{\sigma}^2 = \sigma_v^2$ w.p. 1 (4.15)

Under the assumptions 4-1, 4-2 and 4-3 :

- $\lim_{m\to\infty} \hat{X} = X$ w.p. 1 (4.16)

- $\lim_{m\to\infty} \hat{\alpha} = \alpha$ w.p. 1 (4.17)

- $\lim_{m\to\infty} \frac{1}{m} V_{11} (\Sigma_1^T \Sigma_1 - m \hat{\sigma}^2 I) V_{11}^T = \Omega$ w.p. 1 (4.18)

- $\lim_{m\to\infty} V_{11} V_{11}^T = (I_n + X X^T)^{-1}$ w.p. 1 (4.19)

- $\lim_{m\to\infty} V_{22} V_{22}^T = (I_d + X^T X)^{-1}$ w.p. 1 (4.20)

- $\lim_{m\to\infty} \frac{1}{m} (B - A\hat{X})^T (B - A\hat{X}) = \sigma_v^2 (I_d + X^T X)$ w.p. 1 (4.21)

if V_{22} nonsingular

- $\lim_{m\to\infty} \frac{1}{m} \Delta\hat{B}^T \Delta\hat{B} = \sigma_v^2 (I_d + X^T X)^{-1}$ w.p. 1 (4.22)

if V_{22} nonsingular

Comments :

- (4.11) means that the **error matrix $[\Delta A;\Delta B]$ becomes uncorrelated with the exact data $[A_o;B_o]$ with increasing overdetermination.** By subtracting $\sigma_v^2 I_{n+d}$ from the observations $[A;B]^T C_m [A;B]$ we obtain a strongly consistent estimate of the exact data $[A_o;B_o]^T C_m [A_o;B_o]$.

- (4.12) implies that the arithmetic mean of each column in A and B is a strongly consistent estimator of the asymptotic mean of the corresponding unobservable column in A_o and B_o.

- From (4.11), (4.13) and (4.14), it follows that the first n squared singular values σ_i^2 of $C_m[A;B]$ are strongly consistent estimators of $(\sigma_i^o)^2 + m\sigma_v^2$ with $(\sigma_i^o)^2$ the corresponding squared singular values of the exact data $C_m[A_o;B_o]$. The d smallest squared singular values σ_{n+i}^2 of $C_m[A;B]$ are strongly consistent estimators of $m\sigma_v^2$ or :

$$\lim_{m\to\infty} \frac{1}{m} \sigma_i^2 = \frac{1}{m} (\sigma_i^o)^2 + \sigma_v^2 \quad \text{w.p. 1} \quad \text{for } i = 1, \ldots, n$$

$$\lim_{m\to\infty} \frac{1}{m} \sigma_{n+i}^2 = \sigma_v^2 \quad \text{w.p. 1} \quad \text{for } i = 1, \ldots, d \qquad (4.23)$$

- The estimator $\hat{\sigma}^2$, given by **the mean square of the d smallest singular values of $[A;B]$ divided by m** (see (4.6)), **yields a strongly consistent estimate of the unknown variance σ_v^2 of the errors on the data $[A;B]$**

(see (4.15)). Under the assumptions 4-1 and 4-2, TLS is thus able to estimate the **covariance matrix of the errors** $[\Delta A;\Delta B]$ on the data.

- Furthermore from (4.16) and (4.17), we have that the TLS solution $\hat{X}$ and $\hat{\alpha}$ are strongly consistent estimators of the exact solution X and the intercept α of model (4.3). This means that **the TLS solution $\hat{X}$ and the intercept $\hat{\alpha}$ converge to the true parameters X and α of (4.3) with increasing overdetermination.**

- From (4.18), it might be conjectured that $\frac{1}{m} \hat{A}^T C_m \hat{A} = \frac{1}{m} V_{11} \Sigma_{11}^T \Sigma_{11} V_{11}^T$ is a consistent estimator for Ω given in assumption 4-3. However from (4.15), (4.18) and (4.19) we have :

$$\lim_{m \to \infty} \frac{1}{m} \hat{A}^T C_m \hat{A} = \Omega + \sigma_v^2 (I_n + X X^T)^{-1} \qquad \text{w.p. 1} \qquad (4.24)$$

so that this conjecture is false. This implies that **the exact data matrix $\frac{1}{m} A_o^T C_m A_o$ can not be consistently estimated by the TLS estimator** $\frac{1}{m} \hat{A}^T C_m \hat{A}$, as mentioned in corollary 2-7.

- (4.19), (4.20), (4.21) and (4.22) follow directly from the expressions (2.38), (2.39), (2.56) and (2.87) respectively, proven in chapter 2, by taking the asymptotic limits.

4.4. ASYMPTOTIC DISTRIBUTIONS

From (4.11), it follows that the mean of $[A;B]^T C_m [A;B]$ is :

$$E([A;B]^T C_m [A;B]) = m \sigma_v^2 I_{n+d} + [A_o;B_o]^T C_m [A_o;B_o] \qquad (4.25)$$

where E(.) is the expected value operator. Assume that the assumptions 4-1, 4-2 hold and that the common distribution of the rows of $\Delta = [\Delta A;\Delta B]$ has finite fourth moments. Gleser [41] then proved that the elements of $\frac{1}{\sqrt{m}} ([A;B]^T C_m [A;B] - E([A;B]^T C_m [A;B]))$, together with the elements of $\frac{1}{\sqrt{m}} ([A;B]^T 1_m - [A_o;B_o]^T 1_m)$ in case of an intercept model, have a **symmetric, asymptotic multivariate normal distribution** with zero means. He also derived expressions for the asymptotic covariance between each pair of elements.

The **asymptotic distributions** of $\sqrt{m}(\hat{X} - X)$ and $\sqrt{m}(\hat{\sigma}^2 - \sigma_v^2)$ in both the no-intercept and intercept model, and the asymptotic distribution of $\sqrt{m}(\hat{\alpha} - \alpha)$ in the intercept model are **all (multivariate) normal** with zero means. However, the covariance matrices of these distributions involve third-order and fourth-order cross moments, and thus are both complicated in form and hard to estimate from data. In **order to simplify the results, additional assumptions** are imposed :

Theorem 4-1 :

Let the estimators $\hat{X}$, $\hat{\alpha}$ and $\hat{\sigma}$ be given by (1.25), (4.5) and (4.6) repectively. Let Ω and μ_1 be given by assumption 4-3 and (4.9) respectively. Assume that assumptions 4-1, 4-2 and 4-3 hold and assume further that the cross moments of the common distribution of the rows of $\Delta = [\Delta A;\Delta B]$ are identical, up to and including moments of order four, to the corresponding moments of the multivariate normal distribution with zero mean and covariance matrix $\sigma_v^2 I_{n+d}$.

Under both the intercept and no-intercept models, we have that

1. $\sqrt{m}(\hat{\sigma}^2 - \sigma_v^2)$ has an asymptotic normal distribution with zero mean and variance $2\sigma_v^4 / d$.

2. The elements of $\sqrt{m}(\hat{X} - X)$ have an asymptotic n x d-variate normal distribution with zero means and covariance between the (i,j)-th and (k,l)-th elements given by :

$$\sigma_v^2 [\sigma_v^2 \Omega^{-1} (I_n + X X^T)^{-1} \Omega^{-1} + \Omega^{-1}]_{ik} [I_d + X^T X]_{jl} \qquad (4.26)$$

3. In the intercept model, $\sqrt{m}(\hat{\alpha} - \alpha)$ has an asymptotic multivariate normal distribution with zero mean and covariance matrix.

$$\rho (I_d + X^T X) \quad \text{with}$$

$$\rho = \sigma_v^2 \{1 + \mu_1^T [\sigma_v^2 \Omega^{-1} (I_n + X X^T)^{-1} \Omega^{-1} + \Omega^{-1}] \mu_1\} \qquad (4.27)$$

and further the elements of $\sqrt{m}(\hat{X} - X)$ and $\sqrt{m}(\hat{\alpha} - \alpha)^T$ have an asymptotic (n+1) x d-variate normal distribution where the asymptotic covariance between the (i,j)-th element of $\sqrt{m}(\hat{X} - X)$ and the 1-th element of $\sqrt{m}(\hat{\alpha} - \alpha)^T$ is :

$$\sigma_v^2 (I_d + X^T X)_{jl} \{[\sigma_v^2 \Omega^{-1} (I_n + X X^T)^{-1} \Omega^{-1} + \Omega^{-1}] \mu_1\}_i \qquad (4.28)$$

The proof has been given by Gleser [41]. The same assumptions of theorem 4-1 permit the construction of large-sample approximate confidence regions for the parameters σ_v, X and α of the model (4.3) (see [41]). Other expressions of the asymptotic covariance matrix of the TLS estimator can be deduced in case the model includes trend-like variables and for several other special cases of noise distribution [124].

From the previous, it follows that the **covariance matrix of the TLS estimator** can not be easily obtained in the general case and remains rather complicated for practical use. However, **when the amount of additive noise is low, it can be approximated** as follows.
Assume the one-dimensional problem d = 1 and let Σ_Δ be the covariance matrix of the row vectors of $\Delta = [\Delta A; \Delta b]$. Koopmans [66] has shown that

$$\text{if} \quad \Delta_{ii} << (M_o)_{ii} \qquad \text{for all} \quad i = 1, \ldots, n+d \tag{4.29}$$

with $M_o = \frac{1}{m} [A_o; b_o]^T C_m [A_o; b_o]$, then the covariance matrix of the TLS estimator $\hat{x}$ can be approximated by :

$$\text{cov}(\hat{x}) \approx \frac{1}{m} [x^T; -1] \, \Sigma_\Delta \, [x^T; -1]^T \, (\frac{1}{m} A_o^T A_o)^{-1} \tag{4.30}$$

Under assumption 4-1, Σ_Δ equals $\sigma_v^2 I_{n+1}$. Hence (4.30) yields :

$$\text{cov}(\hat{x}) \approx (1 + \|x\|_2^2) \, \sigma_v^2 \, (A_o^T A_o)^{-1} \tag{4.31}$$

Observe that (4.31) can also be easily deduced from (4.26) by neglecting the term in σ_v^4. For finite data, cov $(\hat{x})$ can be approximated by the estimates $\hat{x}$, σ_{n+1}^2 / m and $A^T A - \sigma_{n+1}^2 I_n$ of the true parameters x, σ_v^2 and $A_o^T A_o$ respectively :

$$\text{cov}(\hat{x}) \approx (1 + \|\hat{x}\|_2^2) \, \sigma_v^2 \, (A^T A - m \, \sigma_v^2)^{-1} \quad \text{with} \quad \sigma_v^2 \approx \sigma_{n+1}^2 / m \tag{4.32}$$

The approximate covariance is found to be reasonably accurate [101].

The matrix $(\frac{1}{m} A_o^T A_o)^{-1}$ is proportional to the covariance matrix of the estimates that would be obtained if the errors occurred along just one coordinate axis. In this case, the ordinary LS estimates are appropriate. Indeed, if A is noise-free, it is well-known that the LS estimate x' is consistent with covariance matrix, given by [50] :

$$\text{cov}(x') = \sigma_v^2 \, (A_o^T A_o)^{-1} \tag{4.33}$$

where σ_v is the variance of the errors on b.

The scale factor $\frac{1}{m} [x^T; -1] \Sigma_\Delta [x^T; -1]^T$ in (4.30) depends upon the true parameter values x and the noise covariance matrix Σ_Δ and is inversely proportional to the number m of observations or equivalently the degree of overdetermination.

Furthermore, it has been established that **for Gaussian noise**, the **bias** of the TLS estimator $\hat{x}$ **is negligible** compared with its standard deviation, and that (4.30) equals the **covariance matrix given by the Cramer-Rao lower bound** for joint unbiased estimates [69]. This means that the TLS solution $\hat{x}$ has minimum variance w.r.t. all joint unbiased estimates when the amount of additive noise is low and Gaussian.

Finally, it can be observed from (4.31) and (4.33) that the **covariance matrix** of the TLS estimator $\hat{x}$ is **larger** than the covariance matrix of the LS estimator x', even if A is noisy. This is confirmed by our simulations in section 4.5 and is also clear from the asymptotic variance expressions of TLS and LS in the case of univariate regression [124-65]. Hence, **despite its asymptotic bias** owing to observational errors, **ordinary LS estimation is thus not necessarily inferior** to TLS estimation since the asymptotic distribution of the TLS estimator is characterized by a larger variance. This is clear when we consider the **asymptotic mean squared error** (AMSE), which is the sum of the asymptotic variance and squared asymptotic bias. The differences between the AMSE of $\hat{x}$ and x' are only significant for sufficiently large noise variances σ_v^2 when the bias of LS becomes more important. This is confirmed by simulations in section 4.5.
Based on the asymptotic MSE, Ketellapper [65] investigated the choice between ordinary LS and corrected least squares (CLS) estimation of parameters in a univariate linear errors-in-variables model under conditions relevant in practice. He found that CLS, which equals TLS under the assumption 4-1 (see section 4.2), is in general the preferable estimation procedure.

4.5. PRACTICAL RELEVANCE AND SIMULATION RESULTS

The major results of this section require two assumptions for their validity :

1. $\Omega = \lim_{m \to \infty} \frac{1}{m} A_o^T C_m A_o$ exists and is positive definite (assumption 4-3).

2. the covariance matrix Σ_Δ of the errors equals $\sigma_v^2 I_{n+d}$ (assumption 4-1).

It is, of course, open to argument whether either of these assumptions can be even approximately valid in practical contexts.

Loosely speaking, **assumption 4-3** requires that the row vectors $(a_o)_i$ of A_o sample all directions in the n-dimensional space with relative frequencies of commensurate magnitude. Certainly, any well-designed experiment would require that the observations on the vector of independent variables, i.e. the rows of A_o, have this property. Mathematically, assumption 4-3 implies that the **rank of A_o** must equal the dimension of the space spanned by the independent variables. Indeed, using simulations for the case n = d = 3, Gleser and Watson [42] under-sampled one dimension in 3-dimensional space. The resulting estimates of X bore little resemblance to the true value. On the other hand, when all dimensions were nearly equally sampled, the estimates of X were reasonably accurate even in moderately small samples.
With respect to parameter estimation in system identification and signal processing, assumption 4-3 implies the necessity for the input signal in A_o to be persistently exciting i.e. the property to sufficiently excite all the modes of the system. As a measure for **persistent excitation** the condition number of the input matrix A can be used [105]. If this number is small, the data are sufficiently excited. A high degree m of overdetermination increases the probability that assumption 4-3 will be satisfied.

With respect to assumption 4-1, it should be noted that the requirement $\Sigma_\Delta = \sigma_v^2 I_{n+d}$ may be generalized to :

$$\Sigma_\Delta = \sigma_v^2 \Sigma_o \qquad \Sigma_o = \begin{bmatrix} \Sigma_{11}^o & \Sigma_{12}^o \\ \Sigma_{21}^o & \Sigma_{22}^o \end{bmatrix} \begin{matrix} n \\ d \end{matrix} \quad \text{known and positive definite.} \tag{4.34}$$

Transformation of the data $[A;B]^T$ to new data $\Sigma_o^{-\frac{1}{2}} [A;B]^T$ where $\Sigma_o^{-\frac{1}{2}}$ has the form :

$$\Sigma_o^{-\frac{1}{2}} = \begin{bmatrix} T_{11} & 0 \\ T_{21} & T_{22} \end{bmatrix} \begin{matrix} n \\ d \end{matrix} \quad \text{with} \quad T_{11} = (\Sigma_{11}^o)^{-\frac{1}{2}}, \; T_{21} = - T_{22} \Sigma_{21}^o T_{11} \text{ and}$$

$$T_{22} = \left[\Sigma_{22}^o - \Sigma_{21}^o (\Sigma_{11}^o)^{-1} \Sigma_{12}^o\right]^{-\frac{1}{2}} \tag{4.35}$$

yields a model for the transformed data of the form (4.3) with covariance matrix $\Sigma_\Delta = \sigma_v^2 I_{n+d}$ [40]. For computational purposes, triangular square roots are recommended. (4.35) can, for example, be obtained from the Cholesky decomposition of Σ_o [48, section 5.2]. If $\hat{\alpha}^*$, $\hat{X}^*$ and $\hat{A}^*$ are the estimators of the parameters α, X and A_o in the new model, then the identities

$$\hat{\alpha} = T_{22}^{-1} \hat{\alpha}^* \; ; \quad \hat{X}^T = T_{22}^{-1} (\hat{X}^{*T} T_{11} - T_{21}) \quad \text{and} \quad A^T = T_{11}^{-1} \hat{A}^{*T} \tag{4.36}$$

allow to deduce the estimators $\hat{\alpha}$, $\hat{X}$ and $\hat{A}$ of the parameters α, X and A_o of the original (untransformed) model. The estimator $\hat{\sigma}^2$ of σ_v^2 is the same under both models. If (0.3) is the SVD of $[A;B] (\Sigma_o^T)^{-\frac{1}{2}}$, then $\hat{\sigma}^2$ is given by (4.6). However, forming $[A;B] (\Sigma_o^T)^{-\frac{1}{2}}$ explicitly and computing its SVD can lead to unnecessary and large numerical errors when $\Sigma_o^{\frac{1}{2}}$ is nearly singular. To avoid this inverse, we better compute the generalized SVD [80].

Thus the basic question that must be raised about **the applicability of using assumption 4-1 is whether or not the form of the error covariance matrix Σ_Δ can be known up to a constant scalar multiple.** This may still appear too restrictive. Indeed, if consistent sequences of estimators for X are desired, the **parametric form** of Σ_Δ can not be completely unspecified. For example, if Σ_Δ is allowed to be a general unknown, positive definite covariance matrix, Gleser [41] showed that the maximum of the likelihood is infinite and no maximum likelihood estimators of the unknown parameters α, X and A_o exist.

Summarizing, it may be concluded that **TLS yields a strongly consistent estimator for the parameters of an errors-in-variables model of the form (4.3) in all cases, provided the covariance matrix Σ_Δ of the error variables is known up to a factor of proportionality.**

Finally it should be noted that we used criteria based on **asymptotic properties** to evaluate the quality of estimates for the parameters of the errors-in-variables model. Hence, our analysis and conclusions are **valid for large-sample models whereas applications often involve moderate sample sizes.** However, there is evidence based on Monte Carlo simulation experiments, that the **moments of the asymptotic distributions** of the LS and CLS (which equals TLS under the assumption 4-1) estimators are good approximations for the corresponding simulated finite-sample moments at least for samples having 20 observations or more [64]. This is also confirmed by our simulations.

As a final result, we therefore compare the accuracy of the TLS and LS solution with respect to their squared bias, total variance and mean squared error (MSE), defined by Gunst, Webster and Mason [51] (see also appendix B) :

$$MSE\ (\hat{x}) = E\left\{(\hat{x} - x)^T\ (\hat{x} - x)\right\} \tag{4.37}$$

$$= E\left\{(\hat{x} - E\ (\hat{x}))^T\ (\hat{x} - E\ (\hat{x}))\right\} + (E\ (\hat{x}) - x)^T\ (E\ (\hat{x}) - x) \tag{4.38}$$

$$= \text{trace}\left\{\text{covariance}\ (\hat{x})\right\} + \text{bias}^2\ (\hat{x})$$

$$= \text{total variance}\ (\hat{x}) + \text{squared bias}\ (\hat{x})$$

with x the exact solution of $A_o x = b_o$. We use the same simulation procedure as explained in section 3.4 but instead of relative errors, the squared bias, total variance and MSE of the solution are estimated. The expected values above are estimated by the sample mean (see (B.2)), using the solution of 100 sets of equations affected by noise with the same variance. We divide those estimated values of the squared bias, total variance and MSE by $\|x\|_2^2$ and then plot those values in a log-log diagram versus the noise variance.

Fig. 4.1 illustrates the **difference in accuracy between TLS and LS with**

increasing degree m of overdetermination. A linear plot of the same example comparing the TLS and LS relative errors with increasing m, has been given in [123]. From fig. 4.1 the following observations are made.

- The **bias of TLS is much smaller** than the bias of LS and decreases with increasing m, conformal to the asymptotic unbiasedness property (4.16) of TLS.
- The **total variance of TLS is always larger** than that of LS. This has been proven for the one-dimensional univariate model [124-65] and can be deduced for the multivariate model from (4.31) and (4.33). The decrease in variance for the case m = 10 and m = 14 which occur at larger noise variances, are due to **rank reduction** of the data. Indeed, using the rank determinator (1.117) in the TLS algorithm 1-1, and equivalently in the LS algorithm, the data matrix reduces its rank as soon as its squared minimal singular value is smaller than the rank determinator given by (1.117). In this case, the minimum norm solution is computed. As it is well-known in principal component analysis [49], this rank reduction causes an increase in bias but a decrease in variance of the solution error.
- And finally, **the MSE combine the effects of bias and variance.** At the smallest noise variances, the LS bias is rather low (and usually substantially smaller than the variance) and the variance of TLS is somewhat larger than the LS variance. These two effects annihilate each other, producing comparable MSE for TLS and LS. By increasing the noise on the data, the bias of LS becomes more important and even the dominating term in (4.38). Hence, the differences in MSE are more pronounced, showing the superiority of TLS. It is evident that the TLS **solution will be more accurate** especially **when the sets of equations are more overdetermined.** Indeed, the TLS estimator then approaches its asymptotic properties. Fig. 4.1 confirms that these properties are already fairly well approximated for **moderate sample size** m. Observe from fig. 4.1.a that the squared bias of TLS does not exceed the noise variance for m = 22.

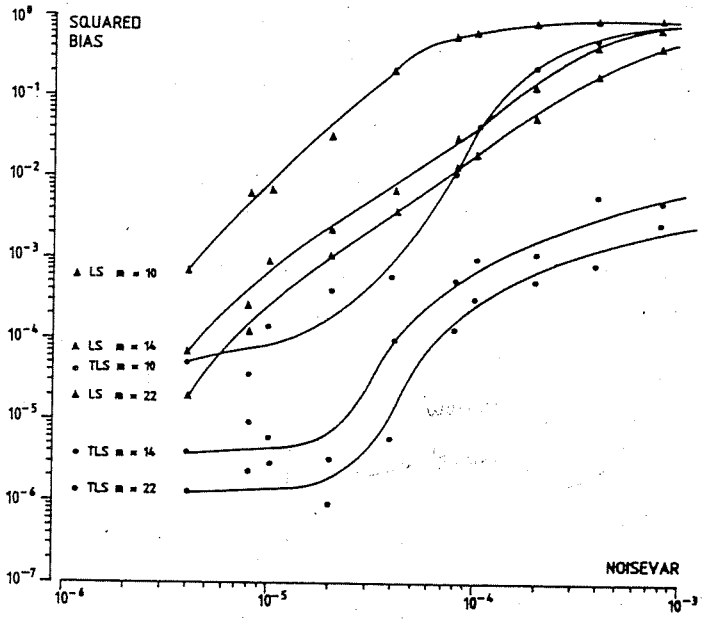
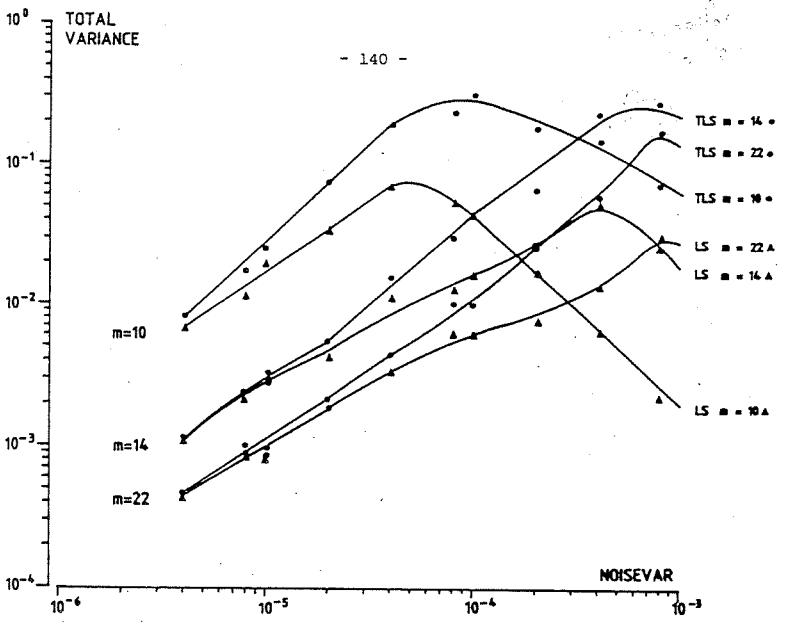
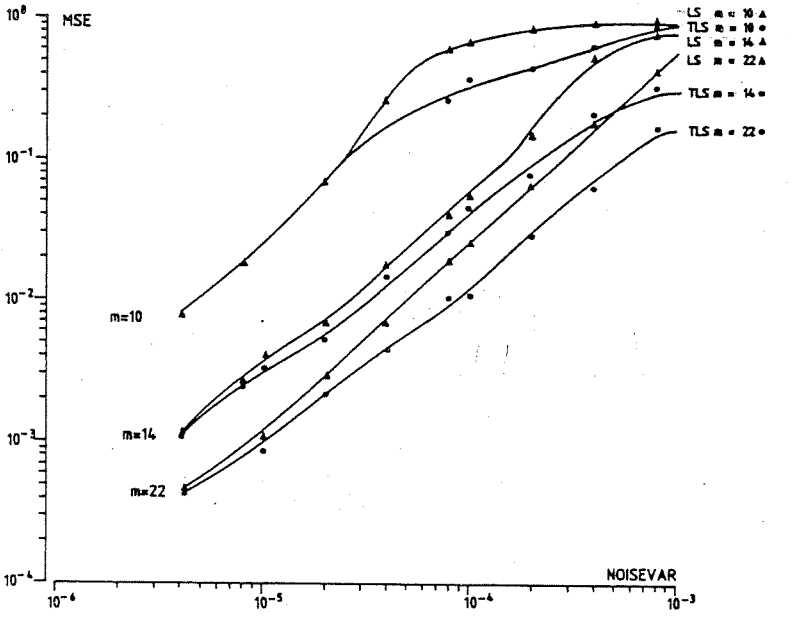


Fig. 4.1 : Comparison of (a) the squared bias, (b) the total variance and (c) the mean squared error (MSE) of the LS and TLS solution (all values divided by $\|x\|_2^2$) versus the variance of the noise added to the set $(A_o)_{mx5}x = b_o$ for different row dimensions m = 10, 14 and 22. The 22 x 5 matrix A_o has singular values 10, 8, 4, 1, .1 and $b_o = .5\ u_5$ where u_5 is the left singular vector of A_o corresponding to .1. By deleting the last rows in this 22 x 6 matrix $[A_o;b_o]$, the other m x 6 matrices $[A_o;b_o]$ are derived.





4.6. CONCLUDING REMARKS

As soon as some **prior information about the distribution of the errors** on the data **is available**, statistical properties of the TLS problem can be derived. They prove **the expected quality** of TLS estimating the parameters of a model where all variables involved contain errors. The most successful approach, directly related to the TLS concept, is the "**errors-in-variables**" model, which considers all observations as coming from some unknown true values plus measurement errors. The true values are assumed to follow a linear relation. It is further assumed that the errors are independently and identically distributed with zero mean and common covariance matrix Σ_Δ. When $\Sigma_\Delta = \sigma_v^2 I_{n+d}$, it is shown that the **TLS solution** of the model is a **strongly consistent** estimator of the true but unknown parameters X of the model (section 4.3). As it is well-known in linear regression, the **ordinary LS solution** is always **inconsistent** in this case (section 4.2).

Beyond an estimate for the parameters X, TLS also provides estimators for the intercept of the model and the variance σ_v^2 of the errors. Under reasonable assumptions, all these estimators are shown to be strongly consistent estimators of the true model parameters, regardless of the common distribution of the errors on the data (section 4.3). When the common error distribution has finite fourth moments, these estimators are also shown to be **asymptotically normally distributed** (section 4.4). Finally the **practical relevance** of the assumptions is evaluated and shown to be fairly reasonable (section 4.5). In particular, it is shown that when $\Sigma_\Delta = \sigma_v^2 \Sigma_o$ with Σ_o known and positive definite, TLS still computes consistent estimates after an appropriate transformation of the data. It may be concluded that **TLS yields a strongly consistent estimator in all cases, provided the covariance matrix Σ_Δ of the error variables is known up to a factor of proportionality.** Moreover it should be noted that, although asymptotic properties are used to evaluate the quality of the estimates for the parameters of an "errors-in-variables" model, these properties are **fairly well approximated in applications of moderate sample sizes.** This is confirmed by simulations.

CHAPTER 5

RELATIONSHIPS BETWEEN TLS ESTIMATION AND CLASSICAL LINEAR REGRESSION

5.1. INTRODUCTION

Multivariate linear least squares regression (LS) has been in use for a long time as the major statistical technique for estimating the regression parameters of a linear regression model. However, the full implications, limitations and inherent problems associated with it have been treated in literature only recently. Despite processing the very desirable property of having minimum variance among all linear unbiased estimators under the usual conditions imposed on the model, the LS estimators can, nevertheless, have extremely large variances when the data are multicollinear, i.e. nearly linearly dependent. Most of the modifications of the ordinary least squares approach are attempts to deal with the problem of **multicollinearity** and many **biased estimators** have been proposed : principal component regression (PC) [30], latent root regression (LR) [132], shrinkage [107], ridge regression (RR) [30-56], ... Despite their better performance in the presence of a multicollinearity, all the techniques cited above, **assume the independent variables to be free of error.** They are **no longer adequate** when **all data are observed with errors.** New techniques have been introduced for estimating the parameters of the linear regression model **with errors-in-the variables.** If the errors on all data are independently and identically distributed with zero mean and common covariance matrix $\sigma_v^2 I$, the total least squares approach yields strongly consistent estimators for the parameters of the linear errors-in-variables model, regardless of the common distribution of the errors (see section 4.3).

It is **the aim** of this chapter to **present some new, remarkable algebraic equivalences and interrelations between the classical linear regression**

estimators (LS, PC, RR and LR) **and total least squares estimation, in the presence of multicollinearities.** The SVD technique, proven to be an excellent tool in regression analysis [74], is used to elucidate those equivalences and geometric concepts clarify the difference in approach of classical regression versus TLS estimation [126].

This chapter is organized into four sections. Section 5.2 clarifies the essential **differences between a regression model and the errors-in-variables model** (4.3). At first sight, the estimation of the parameters in (4.3) looks like a problem in regression analysis and indeed, this resemblance has given rise to much confusion. Next in section 5.3, the classical linear regression techniques are briefly described : least squares (LS), principal component (PC), ridge regression (RR) and latent root regression (LR). The SVD is used to elucidate their equivalences and relation with TLS. Finally in section 5.4, new **algebraic relationships** between TLS and the classical regression techniques are presented. The **equivalence between PC and LR** in the presence of exactly defined multicollinearities is proven and **the difference between LR and TLS estimation** is clarified.

We only consider here one-dimensional problems, i.e. with only one observation vector. The classical regression techniques never treat the multidimensional case. Nevertheless, most of these (except LR) can be straightforwardly extended. Their performance is invariant w.r.t. the number of observation vectors, just like LS (see also fig. 3.6).

5.2. REGRESSION VERSUS ERRORS-IN-VARIABLES MODEL

The differences between the well-known classical regression model and the classical errors-in-variables model, discussed in section 4.1, are best illustrated by considering the univariate, one-dimensional case of the **more general regression model with errors in the variables** :

$$b_o = \alpha + a_o x + \varepsilon \quad ; \quad b = b_o + \Delta b \quad ; \quad a = a_o + \Delta a \qquad (5.1)$$

a_o and b_o are the true, but unobservable variables, Δa and Δb are their

measurement or observation errors, and a and b are the corresponding observable variables. In addition to the errors in the variables Δa and Δb, there is an error in the equation ε. As is usual in econometric work, ε will be interpreted as summarizing all the factors that have an influence on b_o but are not represented by explicit observable variables in the equation. ε is due to the inherent structure of the relationship between the variables. For example, body weight varies with height in an intrinsic way quite unconnected with any errors of measurement. It is assumed that the errors Δa, Δb, ε are uncorrelated with a_o and b_o, their variances $\sigma^2_{\Delta a}$ and $\sigma^2_{\Delta b}$ and their covariance $\sigma_{\Delta a \Delta b}$ (which is zero in case of independence) are known, while σ^2_ε is an unknown parameter to be estimated along with α and x. Model (5.1) typically describes **nonzero residual** problems (see section 3.2.1).

If we assume ε to be zero in (5.1), we obtain the **classical errors-in-variables model.** There exists an exact linear relation between a_o and b_o, in which we are interested. This model thus typically describes **zero residual** problems. a_o and hence b_o, may be either random or nonrandom variables. If a_o is a random variable, the relation $b_o = \alpha + a_o x$ is called a **structural** relation. Otherwise, if a_o is not random the relation is called a **functional** relation [75]. This latter situation is a degenerate case of the regression situation in that here $\varepsilon = 0$ (see further). Note that this situation is **symmetric** in that the expression $a_o = (b_o - \alpha)/x$ is an equally meaningful way of writing the relation $b_o = \alpha + a_o x$ in this context.

In a **classical regression situation**, however, we are concerned with the dependence of b_o upon a_o **which is not subjet to error** : the error variable Δa in (5.1) is identically zero in value, so that $\sigma^2_{\Delta a} = 0$. Thus the **classical regression model is essentially a special case of the more general errors-in-variables regression model** (5.1) but may **not** be considered as a special case of the **classical** errors-in-variables model. Indeed, in contrast to the latter model, the **variation** of the dependent variable b_o in regression analysis is **not** necessarily, or even usually, **due to the error Δb alone**, i.e. $\varepsilon \neq 0$. Hence, one important property of the regression situation when a_o is a random variable is its **asymmetry**

i.e. the expression $a_o = (b_o - \alpha)/x$, a result of algebraic manipulation of the original equation, is not a meaningful relation in this context. Also note that when we write $b = b_o + \Delta b + \varepsilon$, we must distinguish between Δb and ε. The variable Δb is an error of observation which we presumably may eliminate by making finer and finer observations, whereas even if we get rid of Δb, we could never eliminate ε. The variable ε is inextricably tied to the distribution of b_o given a_o. The variable Δb is called a "**fluctuation**" while ε is called the "**individual part**" of an observed quantity [73].

We may easily convince ourselves that the **existence of errors in both a_o and b_o poses a problem quite distinct from that of regression.** A useful consequence of (5.1) is :

$$b = \alpha + ax + u \qquad \text{with } u = \Delta b + \varepsilon - \Delta a x \qquad (5.2)$$

This is not a classical regression situation : a is a random variable and it is correlated with the error term u [62]. Finally, one should note that, in contrast to the errors-in-variables model, it makes no difference in the case of regression analysis whether we fix a_o e.g. at equal intervals along its effective range, or emerge a_o as a result of some random selection, for in either case, a_o is not treated as a random variable in considering the linear relationship $b_o = \alpha + a_o x$ of interest.

5.3. UNIFIED SVD APPROACH OF THE CLASSICAL REGRESSION TECHNIQUES

5.3.1. MULTICOLLINEARITIES

The **most general linear regression model is** of the form :

$$b = \alpha 1_m + Ax + \varepsilon \qquad (5.3)$$

where b is an m-dimensional vector of response variables; 1_m is an m-dimensional vector of ones. $A = [a_1, \ldots, a_n]$ is a full column rank matrix of nonstochastic **error-free regressor variables** and is usually **standardized**, i.e. all columns of A have zero mean and unit length; ε is

an m-dimensional vector of unobservable random error variables which are normally distributed with zero mean and covariance $\sigma^2 I$. α is zero for no-intercept models and an unknown constant to be estimated in intercept models. x is an n-dimensional vector of unknown regression coefficients.

The major statistical technique for estimating x has been LS estimation. However, in spite of the highly desirable estimator properties of unbiasedness, minimum variance (among unbiased estimators of regression coefficients) and known probability distributions, **multicollinearities** can produce extremely **large variances** and thus **destroy the effectiveness of LS estimation.**

Multicollinearities are approximate linear dependencies between the columns of A, and are essentially constant for all responses b (zero, when expressed in terms of standardized variables). They are a consequence of the duplication of information that is provided by the variables. More formally, if there exist k linearly independent nonzero vectors $w_j = [w_{1j}, \ldots, w_{nj}]^T$ such that

$$\sum_{q=1}^{n} w_{qj} a_q \approx 0 \qquad j = 1, \ldots, k \qquad (5.4)$$

then the columns of A are said to be "multicollinear". The existence of a linear relation such as (5.4), is however usually referred to as "collinearity" (which is thus used as a generic term, including a generalization of the more limited case of points on the same line) [74]. The closer the linear combinations in (5.4) are to zero, the stronger are the multicollinearities and the more damaging are their effects on the LS estimator.

This has prompted research into **biased regression estimators.** These estimators attempt to **introduce a small bias** into the regression estimator **while greatly reducing the variance** from that of LS. Despite their relative paucity of exact distributional theory [50], biased estimators can provide more realistic parameter estimates. They are in general more accurate when the data are multicollinear.

5.3.2. BIASED REGRESSION ESTIMATORS

Our discussion is facilitated by expressing each of the various estimators (except latent root) as a linear combination of the right singular vectors of A. Denote the singular value decomposition of A as in (0.2). Small singular values and the corresponding right singular vectors identify multicollinearities. This is because

$$\sigma_j'^2 = v_j'^T A^T A v_j' = (A v_j')^T (A v_j') \qquad j = 1, \ldots, n \qquad (5.5)$$

So if the last (n-r) singular values of A are suitably small, this relation is

$$0 \approx (A v_j')^T (A v_j') \qquad \text{and hence} \qquad A v_j' \approx 0 \qquad (5.6)$$

The elements of the **right singular vectors** can thus be **used to determine the coefficients** used in (5.4). The general form of the estimators is :

$$x_e = \sum_{j=1}^{n} a_j c_j v_j' \qquad (5.7)$$

The subscript e denotes the used regression technique. a_j depends on the particular estimator and $c_j = \sigma_j' u_j'^T b$ is the same for all estimators. Thus the estimators differ only in the values of a_j used in (5.7). Although the latent root estimator can not be expressed as a linear combination of the v_j', its form is similar to (5.7).

The least squares (LS) estimator of x in (5.3) can be written as :

$$x_{LS} = \sum_{j=1}^{n} \sigma_j'^{-2} c_j v_j' \qquad (5.8)$$

This form of the estimator requires that $a_j = \sigma_j'^{-2}$ in (5.7), thereby tending to place large weights on singular vectors associated with small singular values of A.

The principal component (PC) estimator of x in (5.3) is given by :

$$x_{PC} = \sum_{j=1}^{r} \sigma_j'^{-2} c_j v_j' \qquad (5.9)$$

This is equivalent to setting $a_j = 0$, $j = r+1, \ldots, n$ and $a_j = \sigma_j'^{-2}$ for $j = 1, \ldots, r$ in (5.7). PC estimators eliminate multicollinearities from the LS estimator, thereby greatly reducing estimator variances while attempting to introduce only a small amount of bias.

The ridge regression (RR) estimator of x in (5.3) proposed by Hoerl and Kennard [56], can be written using (5.7) with $a_j = (\sigma_j'^2 + k)^{-1}$ for some $k > 0$, i.e.,

$$x_{RR} = \sum_{j=1}^{n} (\sigma_j'^2 + k)^{-1} c_j v_j' \qquad (5.10)$$

Ridge regression adjusts the LS estimator by reducing the influence of multicollinearities on the parameter estimates. Unlike PC and LR regression, no singular vectors v_j' defining multicollinearities are eliminated from the resulting estimator but their effect is greatly diminished. Relatively the largest reductions occur with terms having the smallest singular values. The major controversy surrounding the use of (5.10) concerns the appropriate method for choosing k.

The latent root (LR) regression estimator was developed by Hawkins [53] and Webster, Gunst and Mason [132] as alternative for the PC estimator. Two major differences exist between PC and LR regression : (1) the matrices from which the eigenvalues and eigenvectors are extracted differ and (2) the techniques for deciding whether to delete multicollinear vectors from the respective estimators differ. Let [A;b] be the mx(n+1) matrix whose first n columns contain the (standardized) regressor variables and whose last column contains the standardized values of the response variable. x_{LR} is then calculated from the eigenvalues and eigenvectors of $[A;b]^T [A;b]$, the "correlation" matrix of regressor and response variables or, similarly from the singular values and right singular vectors of [A;b]. Through geometrical arguments the developers show that the response variable aids in the determination of the final estimator and can reveal when LS should be preferred to LR. To be more precise, denote the singular values and corresponding right singular vectors of [A;b] as in (0.3) and let $v_j^T = [v_j^{1T}; v_{n+1,j}]$. Then the LR estimator of x in (5.3) is given by :

$$x_{LR} = \sum_{j=1}^{n+1-s} f_j v_j^1 \qquad (5.11)$$

where $f_j = -\eta v_{n+1,j} \sigma_j^{-2} / \left(\sum_{q=1}^{n+1-s} v_{n+1,q}^2 \sigma_q^{-2} \right)$, s is the number of non-predictive multicollinearities (defined below) and $\eta^2 = \sum_{i=1}^{m} (b_i - \bar{b})^2$ with $\bar{b} = \sum_{i=1}^{m} b_i / m$ ($\eta = 1$ if all columns in [A;b] are standardized).

The form (5.11) is similar to (5.7) but the vectors v_j^1 are not the right singular vectors v_j'. LR regression **extends** PC regression by utilizing an assessment of whether multicollinearities have predictive value and should be retained in the resulting estimator (5.11). When σ_j is nearly zero but $|v_{n+1,j}|$ differs significantly from zero, the multicollinearity is referred to as a **predictive multicollinearity**. However, when both $|v_{n+1,j}|$ and the associated singular value σ_j are nearly zero, the multicollinearity is referred to as a **nonpredictive multicollinearity**, i.e. a multicollinearity among the predictor variables in A that is of **little value in predicting the response variable** b. The **LR estimator** (5.11) **eliminates all nonpredictive multicollinearities**, corresponding to the s terms deleted in (5.11). To accomplish this, the coefficients f_j in (5.11) are determined by first setting $f_j = 0$ if v_j identifies a nonpredictive multicollinearity and then selecting the remaining values to minimize the residual sum of squares. If $s = 0$ then $x_{LR} = x_{LS}$. Gunst, Webster and Mason [51] report that in many cases x_{LR} behaves similarly to x_{PC} so that MSE $(x_{LR}) \approx$ MSE (x_{PC}) for these situations. This result agrees well with our theorems proven in section 1.6 (see section 5.4).

Many other biased estimators, e.g. a shrunken estimator of the form Φx_{LS} where $0 < \Phi < 1$ [107] have been proposed but they are not relevant for our comparison with the TLS estimator (see [54-55] for a list of references).

5.4. ALGEBRAIC COMPARISON OF THE CLASSICAL REGRESSION ESTIMATORS W.R.T. THE TLS ESTIMATOR

In the comparison of the estimators, the presence or absence of (non)-predictive multicollinearities is crucial. In order to allow an adequate algebraic comparison, let us consider the **exact case** throughout this section. This means that nonpredictive multicollinearities are defined by vectors v_j whose (n+1)-th component $v_{n+1,j}$ is exactly zero and the associated singular value σ_j is suitably small. Predictive multicollinearities are defined by vectors v_j, associated with zero singular values σ_j and $v_{n+1,j} \neq 0$.
Let (0.2) (resp., (0.3)) be the SVD of A (resp., [A;b]). Throughout this section, s denotes the number of nonpredictive multicollinearities and p denotes the total number of multicollinearities (predictive and nonpredictive).

If [A;b] has rank n+1-p, then its p smallest singular values are zero and thus, [A;b] has p multicollinearities. Hence, if at least one multicollinearity is predictive, the **TLS estimator** is directly obtained from (1.40) :

$$x_{TLS} = \sum_{j=n+2-p}^{n+1} \lambda_j v_j^1 \qquad \text{with} \qquad (5.12)$$

$$\lambda_j = - v_{n+1,j} \Big/ \sum_{i=n+2-p}^{n+1} v_{n+1,i}^2 \quad \text{and} \quad v_j^1 = [v_{1,j}, \ldots, v_{n,j}]$$

If however all s multicollinearities which appear in [A;b] are nonpredictive ($s \geq 1$), then all $v_{n+1,i} = 0$ for i = n+2-s, ..., n+1. Using theorem 1-3, this means that the vector b is orthogonal onto the singular vectors u_j' of A with j = n+1-s, ..., n. TLS then approximates [A;b] with $[\hat{A};\hat{b}]$ obtained by making the next larger singular value σ_{n+1-s} zero (see section 1.6.1). This means that v_{n+1-s} is made predictive if $v_{n+1,n+1-s} \neq 0$. Hence if $\sigma_{n-s} > \sigma_{n+1-s}$, we have from (2.8) :

$$x_{TLS} = \sum_{j=1}^{n-s} (\sigma_j'^2 - \sigma_{n+1-s}^2)^{-1} \sigma_j' (u_j'^T b) v_j' \qquad (5.13)$$

with s the number of nonpredictive multicollinearities.

Since we approximate [A;b] with $[\hat{A};\hat{b}]$ by making v_{n+1-s} predictive, the expression (5.12) may also be used if we consider the number p of multicollinearities of the TLS approximation $[\hat{A};\hat{b}]$, namely p = s + 1.

(5.13) allows to compare, at least formally, the expressions of the **PC, LS and RR estimators with TLS estimation.** From (5.10) and (5.13) one observes that $x_{TLS} = x_{RR}$ if one takes $k = - \sigma_{n+1}^2$ and assumes no multicollinearities. Ridge regression is in fact a way of regularizing the solution to an ill-conditioned LS problem. TLS is a deregularizing procedure, a kind of **reverse** ridge regression (see section 2.1). This implies that TLS can only be useful in errors-in-variables problems for which it can be proven that the expected value of the perturbed $\sigma_i'^2 - \sigma_{n+1}^2$ equals the i-th singular value of the error-free matrix A_o corresponding to A. In those cases, the TLS solution estimates the solution of the error-free problem (4.3) consistently (see section 4.3).
If we subtract σ_{n+1-s}^2 from each squared singular value $\sigma_i'^2$ of A, we transform the LS estimator (resp., PC estimator) into the TLS estimator if the number s of nonpredictive multicollinearities is zero (resp., > 0). Hence, the TLS estimator (5.13) differs from the PC (resp., LS) estimator (5.9) (resp., (5.8)), depending on the size of the singular value σ_{n+1-s} of [A;b], corresponding to v_{n+1-s} which is made predictive by TLS. They coincide if $\sigma_{n+1-s} = 0$ i.e. if [A;b] = $[\hat{A};\hat{b}]$ has one predictive multicollinearity (see theorem 5-1 below).

The comparison between **LR and TLS estimation** is based on (5.12). Assuming s (s < p) nonpredictive multicollinearities, we observe from (5.11) and (5.12) the strong, though different relation between LR and TLS estimation in contrast to the assertion of Stewart [109]. The relation between LR and TLS depends primarily on the size of the singular value σ_{n+1-s}, corresponding to v_{n+1-s} which is predictive or made predictive by TLS ($v_{n+1,n+1-s} \neq 0$). Indeed, $\sigma_{n+1-s} = 0$ implies that [A;b] has one predictive multicollinearity and thus, the estimates of TLS and LR coincide (see theorem 5-1). Otherwise, TLS makes v_{n+1-s} predictive by making σ_{n+1-s} zero in its approximation $[\hat{A};\hat{b}]$, in order to obtain its estimate (5.12) with p = s + 1. LR however, minimizes the residual sum of squares (like LS), after eliminating the s nonpredictive multicollinearities, in order to obtain its estimate (5.11). Observe that, in (5.11)

the contribution of v_{n+1-s} will dominate and thus x_{LR} approaches x_{TLS}, depending on the size of the corresponding singular value σ_{n+1-s}.

In generalizing the relations between all estimators, the presence or absence of (non)predictive multicollinearities (see definition above) in [A;b] is crucial. First of all, one can prove that, **in case there is an exact solution, it is found by all methods.**

Theorem 5-1 :

If [A;b] has one or more predictive multicollinearities then

$$x_{LS} = x_{PC} = x_{LR} = x_{TLS} = x \qquad (5.14)$$

with x the minimum norm exact solution of Ax = b.

Proof :

As soon as [A;b] has one predictive multicollinearity, assume v_{n+1}, there exists a linear relation between the columns of A and b, namely $[A;b]\, v_{n+1} = 0$. As $v_{n+1,n+1} \neq 0$ it follows immediately that $b \in R(A)$ and $R(A) = R([A;b])$. This means that model (5.3) has an exact solution x. Hence the minimal residual norm must be zero. As σ_{n+1} is also zero, we obtain (5.14) from the definitions of the respective estimators (5.8-5.9 - 5.11 - 5.12). If [A;b] has more predictive multicollinearities, assume s + 1, then A must also have s multicollinearities, or equivalently, s zero singular values according to the interlacing property for singular values (see theorem A.3). Taking the minimum norm solution implies transforming the basis of the right singular subspace of [A;b], corresponding to those s + 1 multicollinearities, to s nonpredictive base vectors and one predictive base vector [47, p. 892], [122, p. 31]. Since $\sigma'_{n-s+1} = \ldots = \sigma'_n = \sigma_{n-s+1} = \ldots = \sigma_{n+1}$, we obtain the unique minimum norm solution (5.14) from the definition of the estimators (5.8-5.9 - 5.11 - 5.13). ■

The difference between the estimators becomes apparent when the model does not have an exact solution, i.e. $R(A) \neq R([A;b])$ and $b \notin R(A)$. This implies that the number of predictive multicollinearities equals zero. In this case all multicollinearities, if any, are nonpredictive. Using the interesting theorem 1-3, we can prove the following important **relation between LR regression and PC.**

Theorem 5-2 :

If there are only nonpredictive multicollinearities, then

$$x_{PC} = x_{LR} \qquad (5.15)$$

Proof :

Assume s nonpredictive multicollinearities corresponding to isolated singular values. By theorem 1-3 (b) the s smallest singular triplets in the SVD (0.2) of A and (0.3) of [A;b] then coincide, i.e.

$$u_{j+1} = \pm u'_j,\ \sigma_{j+1} = \sigma'_j \quad \text{and} \quad [\pm v'^{T}_j;0]^T = v_{j+1} = [v^{1T}_{j+1};0]^T$$

$$\text{for } j = n-s+1, \ldots, n \qquad (5.16)$$

If s = 0 LS, PC and LR all minimize the residual norm $\|Ax-b\|_2$ and thus $x_{LS} = x_{PC} = x_{LR}$. If s > 0 then $v_{n+1,j} = 0$ for j = n-s+2, ..., n+1. Using theorem 1-3 this implies that $u'^{T}_j b = 0$ for j = n-s+1, ..., n. As $v'_j = v^1_{j+1}$ for j = n-s+1, .., n from (5.16), PC and LR will delete the same terms in the expression of their respective estimators (5.9) and (5.11). Hence $x_{PC} = x_{LR}$. In case of nonpredictive multicollinearities corresponding to singular values with multiplicity r > 1, the proof still holds if one deals with the corresponding r-dimensional singular subspaces instead of the singular vectors. ■

Using theorem 1-3 (e), we can also immediately prove that $x_{LS} = x_{PC}$ in case of s nonpredictive multicollinearities.
Geometrically theorem 5-2 implies that b is orthogonal onto the vectors u'_j of A for j = n-s+1, ..., n. In order to find a nonzero approximation b' such that b' belongs to the column space R(A) of A, b must be projected orthogonally onto the (n-s)-dimensional space $R(A_{n-s}) = \text{span}\{u'_1, \ldots, u'_{n-s}\}$ (see fig. 1.2.a). Hence, b' = b" with b" the orthogonal projection of b onto $R(A_{n-s})$. As such, LS, PC and LR obtain their estimate of x in (5.3) which minimizes the residual norm $\|A_{n-s}\, x-b\|_2$.

Combining now the theorems 5-1 and 5-2 we conclude immediately :
If [A;b] has s nonpredictive multicollinearities, then

$$x_{LS} = x_{PC} = x_{LR} = \sum_{j=1}^{n-s} \sigma_j'^{-1} (u_j'^T b).\ v_j' \qquad (5.17)$$

Comparing (5.13) with (5.17) we observe that $x_{TLS} \neq x_{PC}$ and $x_{TLS} \neq x_{LR}$ if $\sigma_{n-s+1} > 0$. **This difference in estimation depends on the size of the singular value σ_{n-s+1}** and can be best understood when we present the problem geometrically. Like LS, PC and LR, TLS also tries first to find an approximation $[\hat{A};\hat{b}]$ orthogonal onto the vectors $u_{j+1} = u_j'$ for $j = n-s+1, \ldots, n$ in order to make $\hat{b} \in R(\hat{A})$. As b is orthogonal onto $u_{j+1} = u_j'$ for $j = n-s+1, \ldots, n$, b remains unchanged after the orthogonal projection. Hence $\hat{b}$ still $\notin R(A_{n-s}) = R(\hat{A})$. In order to obtain a solution, TLS makes the TLS approximation $[\hat{A};\hat{b}]$ of $[A;b]$ orthogonal onto u_{n-s+1} (see fig. 1.2.b) by making the next larger singular value σ_{n-s+1} of $[A;b]$ zero, i.e. v_{n-s+1} is made predictive. If now $v_{n+1,n-s+1} \neq 0$, $\hat{b} \in R(\hat{A})$. As such the total approximation effort $\|[\Delta\hat{A};\Delta\hat{b}]\|_F$, equal to σ_{n-s+1}, is minimized such that $\hat{b} \in R(\hat{A})$. The TLS estimate x_{TLS} of x in (4.3) is obtained from (1.4). Observe that **σ_{n+1-s} measures the degree of incompatibility of the set of linear equations Ax = b and thus, indicates how closely the data A, b fit the linear model** (5.3) or (4.3).

We now summarize the main **conclusions.**

- If **$b \in R(A)$**, $[A;b]$ has **at least one predictive multicollinearity** and the **exact solution** of $Ax = b$ is **found by LS, PC, LR and TLS estimation.**
- If **$b \notin R(A)$, all multicollinearities** of $[A;b]$ must be **nonpredictive.** Assume this number is equal to s $(s > 0)$.
  The **difference in estimation** becomes now apparent and **is proportional with the size of the next larger singular value σ_{n+1-s}** of $[A;b]$, measuring how closely the data fit the model (5.3) or (4.3).
  LS, PC and LR take the orthogonal projection b' of b onto R(A) in order to make $b' \in R(A)$ and obtain their estimate (5.17) by solving $Ax = b'$. As such, the residual norm $\|Ax-b\|_2$ is minimized.
  TLS however, approximates $[A;b]$ with $[\hat{A};\hat{b}]$ by making σ_{n+1-s} zero, i.e. TLS makes one v_{n+1-s} predictive such that $[\hat{A};\hat{b}]$ has now one predictive multicollinearity. In this way, $\hat{b}$ is made $\in R(\hat{A})$ and TLS obtains its estimate (5.13) by solving $\hat{A}x = \hat{b}$. As such, the total approximation effort $\|[A;b] - [\hat{A};\hat{b}]\|_F$, such that $\hat{b} \in R(\hat{A})$, is minimized and given by σ_{n+1-s}.
- Hence, we obtain the **minimal residual norm with LS, PC or LR,** i.e. $\|Ax_{TLS} - b\|_2 > \|Ax_{LS} - b\|_2$ but the **minimal total approximation effort with TLS,** i.e. $\|[A;b] - [\hat{A};\hat{b}]\|_F < \|b - b'\|_2$, as proven by Van Huffel, Vandewalle and Staar [130].

Note that in practice the multicollinearities are not exact, i.e. one has numerically $\sigma_j \approx 0$ and $v_{n+1,j} \approx 0$. Depending on the strength of the multicollinearities the same relations hold approximately for the inexact cases.

Finally we mention one **chemical example in multivariate regression** [135]. In this example, the ordinary LS, the PC, the RR, the shrunken estimator and the TLS estimator are compared w.r.t. their prediction accuracy and the stability of their estimates, in the presence of multicollinearities among the independent variables A. In many chemical applications of multivariate regression, like the present example of relationships between chemical structure and biological activity, **the predictive properties** of the model are of prime importance and the regression estimates therefore often **need to be stabilized.** The example [135] shows that TLS gives a solution to the multivariate regression problem which is stabilized in comparison with the LS solution and which has (at least in the example investigated) minimal prediction error. Stabilization is performed by reducing the matrix $[A;b]$ **to a matrix of smaller rank. All variables remain included in the model.** In applied work, the multicollinearity problem is often handled by selecting a subset of variables such that the size of the subset equals the rank of the matrix $[A;b]$ (see chapter 6).

5.5. CONCLUDING REMARKS

In this chapter, the **TLS estimator of the parameters in a linear errors-in-variables model is compared with the classical regression estimators of the parameters in a linear regression model.** First of all, the main

differences between both models are outlined. The **classical linear errors-in-variables model** always **assumes an unknown but exact linear relation** between the true variables which can only be observed with errors. The **linear regression model**, however, assumes **the independent variables** A to be **error-free** but, in addition, allows the dependent variable b to be influenced by other factors than observation errors alone (section 5.2). The major statistical technique for estimating the parameters of a linear regression model has been LS estimation. However, despite possessing the very desirable property of having minimum variance among all unbiased estimators, it can, nevertheless, have extremely large variances, when the data are multicollinear (section 5.3.1). By introducing a small bias while greatly reducing the variance from that of LS, **biased estimators are shown to perform better in the presence of multicollinearities.** Three biased regression estimators are described using a SVD approach : principal component (PC) regression, ridge regression (RR) and latent root (LR) regression (section 5.3).

Algebraic equivalences and interrelations between these regression estimators and TLS estimation are presented, based on the SVD and geometric concepts. The **equivalence between PC and LR** in the presence of exactly defined multicollinearities, is proven and **the difference between LR and TLS** is clarified (section 5.4).

CHAPTER 6

SUBSET SELECTION USING THE TLS APPROACH IN MULTICOLLINEARITY PROBLEMS WITH ERRORS IN THE VARIABLES

6.1. INTRODUCTION

As mentioned in the previous chapter, the notion of multicollinearity refers to the situation in which there is a nearly exact linear relation among the variables in the model. The consequences are well-known; in particular, coefficient estimates obtained with ordinary LS tend to be inflated and predicted values may be unreasonable. One way of handling the multicollinearity problem and stabilizing the coefficients is reducing the LS estimator with a factor k, called ridge regression or, less used, shrinking the LS estimator with a shrinking factor Φ ($0 < \Phi < 1$) (see section 5.3.2). Another way is based on reducing the rank of the matrix of variables. Replacing the original data matrix A, whose SVD is given by (0.2), by its rank r approximation ($r \leq$ rank (A)) :

$$A_r = \sum_{i=1}^{r} \sigma_i' u_i' v_i'^T \qquad (6.1)$$

in the LS problem amounts to filtering out the smallest singular values and can make a great deal of sense in those situations where A is derived from noisy data. The r first singular vectors can then be regarded as new independent predictor variables, i.e. principal component regression. A similar approach is found in latent root regression (see section 5.3.2). All those approaches offer an estimate and predictor involving **all** variables of the original model. Indeed, although principal component regression reduces the original matrix to rank r, each singular vector is still a linear combination of all its original variables. Thus, although the dimensionality of the space may be reduced by selecting singular vectors, one must still interpret results about the original number of variables. **In many applications it is desirable not only**

to reduce the dimensionality of the space but also to reduce the number of variables that are to be considered or measured in the future. For example, the model builder may not be interested in a predictor such as $A_r x$ that involves all redundant variables. Instead, a predictor Az may be sought where z has at most r nonzero components. The position of the nonzero entries determines which columns of A, i.e. which variables in the model are to be used in approximating the response vector b. How to pick these columns is the problem of **subset selection.**

Procedures for discarding or selecting variables in multivariate analysis are available in a variety of settings. The regression area has been investigated extensively and a variety of selection criteria have been proposed (see [54-55-114-115] for a review). Many of these criteria are simple functions of the mean squared error of estimation (or prediction) which trades off estimated variance against estimated squared bias on a one-to-one or weighted basis. Another criterion, efficient in the elimination of collinearities is **maximizing the independence between the selected variables.** This can be expressed quantitatively by using a QR decomposition of the matrix of all variables with column pivoting. However in settings that involve near rank deficiency, it can fail to obtain a set of linearly independent columns [44]. Hence it is advisable to use the **singular value decomposition** (SVD). A subset selection procedure that is based on this decomposition has been proposed by Golub, Klema and Stewart [48, section 12.2]. This approach is based on a least squares approximation and considers the columns of the data matrix A as values of the real variables in the model. Hence they are obtained without error. However in many applications, the desired variables as well as the response variables are unobservable and can't be obtained without error. For those cases, only perturbed information about the variables involved in the model, e.g. by taking measurements, is available. Hence, **a more global approximation of the problem which takes into account the errors on all variables**, is adequate here and can improve the reliability and stability of the selection. Moreover, by **using the information provided by the response variable, the accuracy** of the selected subset model w.r.t. parameter estimation and prediction **can be improved.** With the properties of TLS in mind, a **new subset selection procedure based on TLS** [125], is presented in this chapter. It will be compared with the

subset selection procedure of Golub, Klema and Stewart [44] w.r.t. its **stability** in subset choice and its **accuracy in parameter estimation and prediction, when errors occur in all variables.**

The key to a **successful prediction of future responses** is the careful examination of the needs. If the objective is to predict the response from new, noisy measurements of the predictor variables, e.g. input - output data taken from the system, the predictor performs best if it models the noise. This implies that the predictor must contain a perturbation model and must be used in the same conditions as during estimation of parameters, e.g. in real-time prediction [39]. In those cases, the bias may be beneficial for the prediction performance of the model. This is clear from [72, example 5.1] showing that biased estimators are strictly better predictors than the true parameters of the system. In other words, this is just the case handled by LS regression because we do not want to model the true but the observed values of the predictor variables (which can thus be considered without error).
However, if **the objective is to predict responses of the true system,** we are interested in the parameters of the true system. Hence, we want to filter out the observation errors. For example, in **hypotheses** one utilises hypothetical values of the predictor variables to study the response behaviour. Also, when **one wants to simulate the behaviour of the true system,** the predictor must converge to an exact description of the interesting part of the true system (despite the fact that it does not yield the best predictor w.r.t. the first objective) [39]. In this case, the predictor may even be used when the environment is different from that which prevailed during the data collection period, e.g. with different inputs, different noise sources or even no noise at all. Because of its important property (see section 4.3) to estimate the unobservable true parameters in a model consistently from noisy data, TLS will be of primary importance, if the latter objective of prediction is pursued. Only those latter needs of prediction are considered in our analysis. It is shown in this chapter that **the new subset selection procedure presented here, must be applied whenever only noisy data are available and one wants to know the exact relationship between the response and the most independent variables of a given model** or study the exact properties of the selected subset model.

This chapter is organized into four sections. Section 6.2 presents the new subset selection algorithm based on TLS, and discusses some variants. In section 6.3 the subset selection algorithms and their variants are evaluated in a simulation set up w.r.t. their selection abilities and accuracy in parameter estimation and prediction. Finally, section 6.4 summarizes the conclusions.

Throughout this chapter, we consider the one-dimensional model i.e. $d = 1$ in (0.1). All the algorithms and their variants discussed in this chapter, can be easily generalized to handle multidimensional problems.

6.2. SUBSET SELECTION ALGORITHMS

Before starting, we define some additional, subset related notations used throughout this chapter.

- Let A_1 be the selected r-dimensional subset of A and let z be the r-dimensional solution vector of the subset model $A_1 z = b$. To specify the method, use the subscript α where α denotes the used subset selection algorithm, i.e. A_α, b_α, z_α. The subscript 0 is used to denote the exact case.
- The SVD of the subset matrix A_1 (resp., $[A_1;b]$) is also denoted by (0.2) (resp., (0.3)).
- Denote by V_r (resp., V_r') the submatrix containing the first r right singular vectors of $[A;b]$ (resp., A).

The method used by Golub, Klema and Stewart [48, section 12.2] to select their independent columns, is based on the following theorem.

Theorem 6-1 :

Let the SVD of $A \in \mathbb{R}^{m\times n}$ be given by (0.2) and define the matrix $A_1 \in \mathbb{R}^{m\times r}$, $r \leq \text{rank}(A)$, by $AP = [A_1;A_2]$ where $P \in \mathbb{R}^{n\times n}$ is a permutation matrix. If

$$P^T V' = \begin{bmatrix} \tilde{V}'_{11} & \tilde{V}'_{12} \\ \tilde{V}'_{21} & \tilde{V}'_{22} \end{bmatrix} \begin{matrix} r \\ n-r \end{matrix}$$

$\quad\quad\quad

As A has rank r, it has exactly r independent columns. We will select the most independent ones in A_1 from (6.4). We can state that

$$R(A_r) = R(A_1) = \text{span}\{u_1, \ldots, u_r\}$$

and the unique solution $z \in \mathbb{R}^r$ is obtained from

$$z = A_1^{-1} b = R_{11}^{-1} Q^T v_{1b} \qquad (6.6)$$

Hence z can be directly obtained from the 2nd step of SAB-TLS by back substitution (R_{11} is upper triangular) [48, section 4.1] :

$$R_{11} z = Q^T v_{1b} \qquad (6.7)$$

If errors are introduced in all variables then $\sigma_{r+1} \neq \ldots \neq \sigma_{n+1} > 0$ and hence the r selected columns of A will also have components in the direction of $u_{r+1}, \ldots, u_{n+1}$, i.e. $R(A_r) \neq R(A_1)$. We can however still compute the solution (6.7) and compare it with the TLS solution of $A_1 z = b$. If we assume that the n-r+1 smallest singular values of [A;b] are nonzero only due to the perturbation of the exact but unobservable model, then $R(A_r)$ will be more stable than $R(A_1)$ and hence (6.7) might predict the stable components of b, i.e. $\sum_{i=1}^{r} u_i^T b$, more accurately.

This computationally more efficient variant of the SAB-TLS algorithm is called **SAB-VTS** in the simulation study of section 6.3.

Finally, we summarize the main characteristics of the subset selection algorithms and their variants, in the table on p. 166.

6.3. OVERALL COMPARISON OF SUBSET SELECTION ALGORITHMS

Let us describe an extensive set of experiments to evaluate the subset selection algorithms and their variants, discussed in the previous section. With respect to prediction, we only consider here the need of predicting future responses in the true system, i.e. in cases where one deals with exact values of the predictor variables, e.g. in hypotheses and simulation studies (see section 6.1). The TLS concept typically applies to those cases.

method	1st step : SVD of	2nd step : subset selection of r columns A_α of A via QR with column pivoting based on	3rd step : solution method
SA-LS	A	$V_r'^T = [v_1', \ldots, v_r']^T$	LS solution of $A_\alpha z = b$
SA-TLS	A	$V_r'^T = [v_1', \ldots, v_r']^T$	TLS solution of $A_\alpha z = b$
SAB-TLS	[A;b]	$V_r^T = [v_1, \ldots, v_r]^T$	TLS solution of $A_\alpha z = b$
SAB-VTS	[A;b]	$V_r^T = [v_1, \ldots, v_r]^T$	solution of $(A_\alpha)_r z = b_r$ with $(A_\alpha)_r$ and b_r the projection of A_α and b onto span $\{u_1, \ldots, u_r\}$ of [A;b]

SIMULATION SET UP

Starting from an exact set of m equations $A_0 x = b_0$ in n unknowns with n-r multicollinearities, i.e. rank $(A_0) = r$, we add noise with zero mean and variance σ_v^2 to the exact data A_0 and b_0. Call the noisy data matrix A and noisy response vector b. We compute the r-dimensional subset A_α of A where α denotes one of the following subset selection algorithms : SA-LS, SA-TLS, SAB-TLS or SAB-VTS. Subset selection based on $V_r'^T$ (resp., V_r^T) of A (resp., [A;b]) is denoted by SA (resp., SAB). We then calculate the corresponding subset solution z_α of the noisy subset model $A_\alpha z_\alpha = b$. The prediction accuracy is measured by computing the mean squared error of prediction (MSEP), which trades off estimated variance against estimated squared bias on a one-to-one basis [115, pp. 133] :

$$\text{MSEP}(b_\alpha) = E\{\|b_\alpha - b_0\|_2^2\} = \text{total variance}(b_\alpha) + \text{bias}^2(b_\alpha) \qquad (6.8)$$

$$\text{total variance}(b_\alpha) = E\{\|b_\alpha - E(b_\alpha)\|_2^2\} \qquad (6.9)$$

$$\text{squared bias}(b_\alpha) = \|E(b_\alpha) - b_0\|_2^2 \qquad (6.10)$$

where the predicted response $b_\alpha = (A_0)_\alpha z_\alpha$; $(A_0)_\alpha$ is the exact subset of

A_0 selected by α and α denotes the used subset selection algorithm. The expected values are estimated by the sample mean (see (B.2)) using a set of 100 noisy models $Ax = b$ affected by noise with the same noise variance σ_v^2. The simulation is repeated for larger noise variances and the estimated values of the squared bias, total variance and MSEP are plotted versus the noise variance in a log-log diagram (see fig. 6.1 and 6.2).

Note that, since $R(A_0) = R((A_0)_\alpha)$ and $b_0 \in R(A_0)$, also $b_0 \in R((A_0)_\alpha)$. Hence, if z_0 is the exact subset solution of $(A_0)_\alpha z_0 = b_0$, then (6.8) can also written as ("tr" means trace) :

$$\text{MSEP}\ (b_\alpha) = \text{tr}\ [(A_0)_\alpha\ E\left\{(z_\alpha - z_0)(z_\alpha - z_0)^T\right\}\ (A_0)_\alpha^T] \tag{6.11}$$

$$= \sum_{i=1}^{m} (a_i^0)_\alpha\ E\left\{(z_\alpha - z_0)(z_\alpha - z_0)^T\right\} (a_i^0)_\alpha^T \tag{6.12}$$

with $(a_i^0)_\alpha$ the i-th row of the exact subset $(A_0)_\alpha$ and

$$\underbrace{E\left\{(z_\alpha - z_0)(z_\alpha - z_0)^T\right\}}_{\text{MSE-term of } z_\alpha} = \underbrace{E\left\{(z_\alpha - E(z_\alpha))(z_\alpha - E(z_\alpha))^T\right\}}_{\text{covariance } (z_\alpha)} +$$

$$\underbrace{(E(z_\alpha) - z_0)(E(z_\alpha) - z_0)^T}_{\text{bias-term of } z_\alpha} \tag{6.13}$$

Hence, the prediction accuracy is determined by the accuracy of the solution of the subset equations $A_\alpha z_\alpha = b$. Several conclusions can be made now :

- First of all **the estimation of the coefficients of the selected subset**, i.e. 3rd step of the subset selection algorithm, **by TLS is always more accurate than by LS**. Hence, it confirms the superiority of TLS in the study of the exact relationships between the selected variables and the response. This is evident by comparing the results obtained with SA-LS and SA-TLS (fig. 6.1). Indeed, from (6.12) and (6.13) we know that the difference in prediction accuracy between TLS and LS depends on the difference in accuracy of their solutions of the subset equations $A_\alpha z_\alpha = b$. Our results of chapter 3 and 4 are thus directly

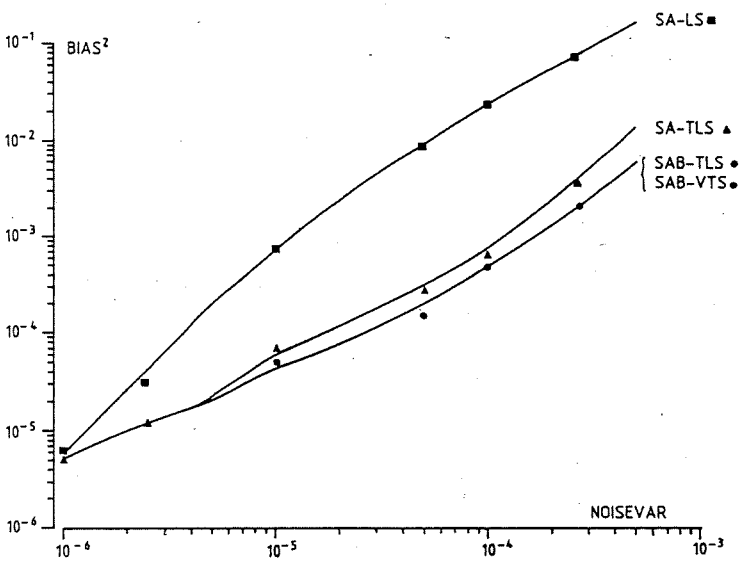


Fig. 6.1 : Comparison of SA-LS, SA-TLS, SAB-TLS and SAB-VTS with respect to their prediction accuracy, computed over 100 noisy sets of equations. The exact original model is $(A_0)_{30\times10}x = b_0$ with $\sigma'(A_0) = \{10, 5, 1, .5, .1, 0, 0, 0, 0, 0\}$ and $b_0 = \sum_{i=1}^{5} (1/\sqrt{5})\ u_i'$ in which 5 columns a_i must be deleted.

(a) the squared bias of prediction; (b) the prediction variance; (c) the mean squared error of prediction.

applicable. Based on the asymptotic properties of TLS and LS estimation and the fact that the errors on the data are rowwise independently and identically distributed with zero mean and common covariance matrix $\sigma_v^2 I$, it follows that the TLS solution yields a strongly consistent estimate of the exact model parameters whereas LS looses its optimal properties and is biased, i.e.

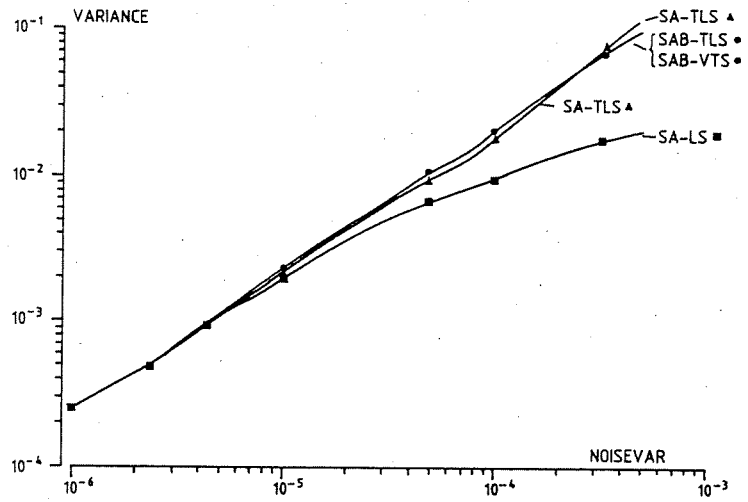


Fig. 6.1.b : The prediction variance.

$$\lim_{m\to\infty} \hat{z}_a = z_0 \quad \text{and} \quad \lim_{m\to\infty} z'_a \neq z_0 \qquad \text{w.p. 1} \qquad (6.14)$$

$\hat{z}_a$ (resp., z'_a) is the TLS (resp., LS) solution of $A_a z_a = b$. (fig. 6.1.a)

Hence, TLS will produce a higher prediction accuracy than LS when $m \gg r$. Although the (asymptotic) variance of the TLS solution remains slightly larger (fig. 6.1.b) causing a comparable prediction accuracy for very small perturbation, the TLS solution enables quickly a better prediction accuracy w.r.t. the LS solution as soon as the noise variance σ_v^2 increases (fig. 6.1.c).

This is evident when the set of equations is more overdetermined since the TLS estimator then approaches its asymptotic properties. This is certainly the case when selecting a subset of r variables as $m/r > m/n$. Hence the TLS technique is recommended. Observe that this conclusion is independent from the subset selection procedure itself (2nd step).

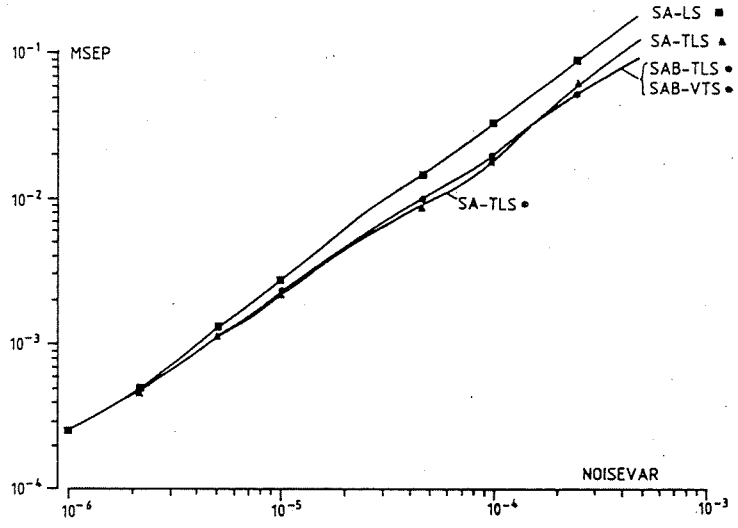


Fig. 6.1.c : The mean squared error of prediction.

Hence it still holds when the subset is selected by any other variable selection method described in literature, provided the final solution of the set of equations corresponding to the selected subset is calculated with TLS (3rd step). Conformal with our analysis in section 4.5 (fig. 4.1), this effect increases as m/r is increasing, e.g. when the number r of selected columns of A is smaller w.r.t. the original number n of variables in the model. On the other hand, the effect decreases as m/r converges to one and the norm of the solution z_0 to zero. This is evident from theorem 2-16 in section 2.4. If the norm of the solution is small, b is bent towards R(A). Hence the TLS solution comes close to the LS solution where b is completely bent towards R(A). By scaling b in such a way that $\|z_0\|_2$ converges to zero, TLS approaches the LS solution. In this sense TLS is more general than

LS via the scaling factor. LS can be considered as a special case of TLS.

- Secondly, the accuracy of the SAB-VTS solution is in all experiments equally or even slightly more accurate than the SAB-TLS solution. Hence **the use of SAB-VTS is preferred because of its computational efficiency** : the computation time is reduced by 20 % to 50 % depending on the parameters of the model. The only disadvantage is perhaps its adaptivity to noise superposition. SAB-VTS can reject noise and thus stabilize its subset solution only by further reducing the subset size, while SAB-TLS can reject noise either by reduction of the subset size or the rank of the selected subset model [124].

- Third, by comparing the properties of subset selection performed on $V_r'^T$ (A)(SA) and subset selection performed on V_r^T of [A;b] (SAB), we observe that, **in cases of sufficiently small perturbations of badly conditioned datasets, SA enables a slightly better prediction accuracy (SA-TLS). In all other cases,** i.e. when the datasets are well-conditioned or the perturbations are no longer small, **SAB is superior in prediction accuracy (SAB-TLS). Moreover, SAB guarantees a higher and longer stability and reliability of its subset choice in the presence of errors in all variables.** All those facts are justified now.

First, observe that a difference between SA and SAB is only due to the influence of the response b, causing possibly different subset choices. If the norm of b is small, then b has little influence and SAB and SA will always choose the same subset A_α. Moreover, small $\|b\|_2$ implies a small norm $\|z_0\|_2$ of the subset solution. As discussed previously, the TLS solution z_α then also approaches the LS solution. Hence as the norm of the response b converges to zero, all subset selection algorithms converge to each other and compute identical results. A difference in prediction accuracy is apparent when SA and SAB choose different subsets A_α and can be evaluated by comparing the results of SA-TLS with those of SAB-TLS.

For sufficiently small perturbations, this difference in prediction accuracy depends mainly on the **condition of the subset** $(A_0)_\alpha$. Indeed, the

bias term in (6.13) is negligible and the covariance of z_α will determine the prediction accuracy. From (4.31), it follows that cov (z_α) depends on the variance σ_v^2 of the noise on the data, the norm of the exact solution z_0 and the condition of $(A_0)_\alpha$, mainly determined by its minimal singular value. W.r.t. the difference between SA-TLS and SAB-TLS, this latter parameter is of primary importance. By maximizing the minimal singular value $\sigma'_{min}(A_\alpha)$ of the subset A_α, the sensitivity of the subset solution z_α is minimized. From theorem 6-1, we know that SA selects the subset which maximizes $\sigma'_{min}(A_\alpha)$. Hence :

$$\sigma'_{min}\ (A_{SA}) > \sigma'_{min}\ (A_{SAB}) \qquad (6.15)$$

Assuming **bad condition**, the subset solution is very sensitive. We expect a large covariance for the subset solution in (6.13) which essentially affects the prediction accuracy (6.12). Hence the subset selection procedure which selects A_α such that the sensitivity of its subset solution z_α is minimized, i.e. such that $\sigma'_{min}(A_\alpha)$ is maximal, minimizes also the prediction accuracy. Therefore from (6.15), we usually obtain a higher prediction accuracy with SA-TLS in case of bad subset condition and sufficiently small perturbations (fig. 6.1). This means that we assume $\sigma'_r(A)$ and $\sigma_r[A_\alpha;b]$ to be sufficiently larger than the noise variance such that the r-th singular triplet of A_0 and $[(A_\alpha)_0;b_0]$ can still be accurately defined. This is confirmed by experiments.

However in case of **good** subset **condition**, the sensitivity of the subset solution is no longer of primary importance. We expect good subset solution accuracy. Hence the advantages of a selection which takes into account the extra information provided by the response b, can now influence the prediction accuracy and improve it as shown in figure 6.2. Moreover, a good condition of the subset equations also implies that the r largest singular values of the data matrix A_0 are of the same order of magnitude and hence, many well-conditioned subset of A are acceptable candidates for selection. It is evident that in this case, the extra information provided by the response b might help to choose the best subset A_α for b. This is clear from table 6.1 where SA deletes a variable (column of A) arbitrarily with little relationship between the variables deleted and the magnitude of the estimated coefficients, i.e. the orientation of the response b with respect to the columns of A. For example, one would

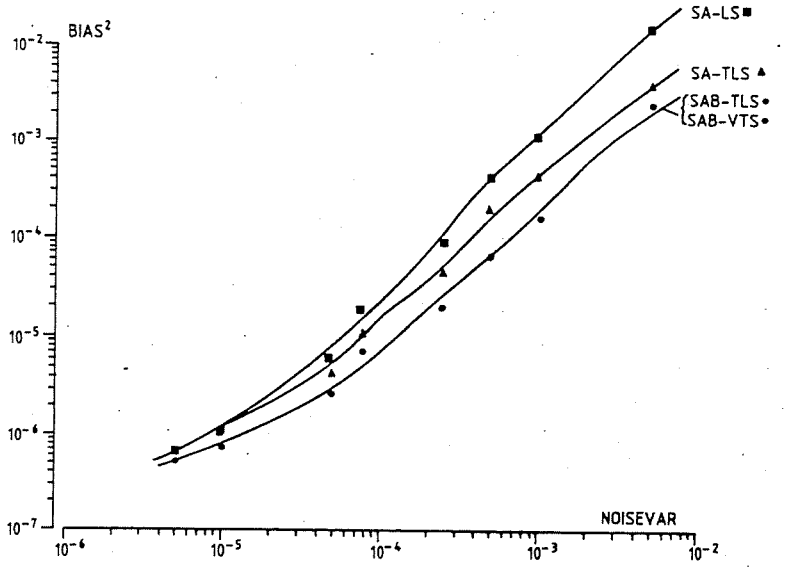


Fig. 6.2 : Comparison of SA-LS, SA-TLS, SAB-TLS and SAB-VTS with respect to their prediction accuracy, computed over 100 noisy sets of equations. The exact original model is $(A_0)_{20\times 8} x = b_0$ in which each column has length one; the six first columns of A_0 are essentially orthogonal to each other and $b_0 = (10a_1 - a_5 - a_6 + a_7 + a_8)/\sqrt{104}$. The two multicollinearities are defined by $-5a_1 - a_2 + a_3 + a_4 - 10a_7 = 0$ and $7a_1 + 4a_4 - 10a_8 = 0$. Two columns a_i have to be deleted.

(a) the squared bias of prediction; (b) the prediction variance; (c) the mean squared error of prediction.

hope that a_1 would not be deleted as this variable provides important information about the response b (large coefficient z_1). This is not the case here. SAB on the other hand, takes into account the extra information provided by the response b, hence it will not delete a_1. SAB is thus more systematic in its deletion of variables. This effect is most

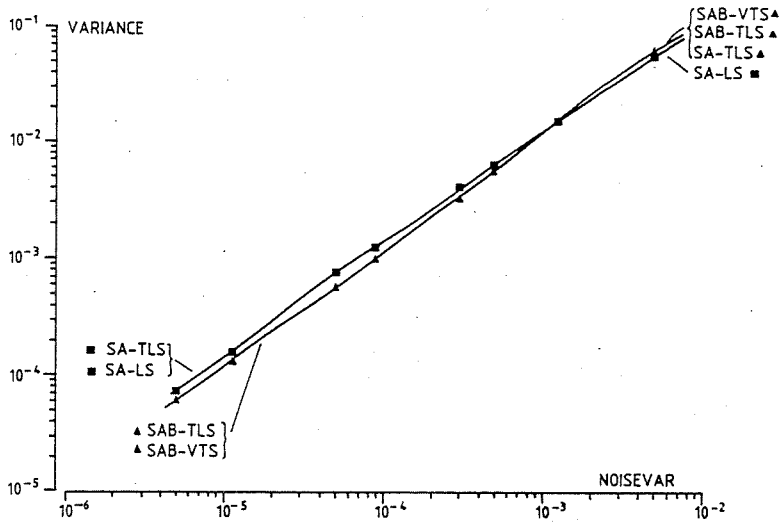


Fig. 6.2.b : The prediction variance.

pronounced when the selected subset is well-conditioned, e.g. when one deals with orthogonal variables in standardized datasets or datasets with a multiple r-th singular value, in the presence of multicollinearities.

The difference in bias of prediction between SA-TLS and SAB-TLS (see fig. 6.1.a) can be explained as follows. From theorem 3-5 in section 3.3.1, we know that the sensitivity-to-noise of the r-th principal right singular subspace V_r of $[A_\alpha;b]$ and its orthogonal complement, is inversely proportional with the size of its corresponding minimal singular value σ_r of $[A_\alpha;b]$. As the TLS solution $[z_\alpha^T;-1]^T$ lies in the orthogonal complement of $V_r[A_\alpha;b]$ (see section 1.3), the subset selection procedure which selects its subset A_α such that its r-th right singular subspace

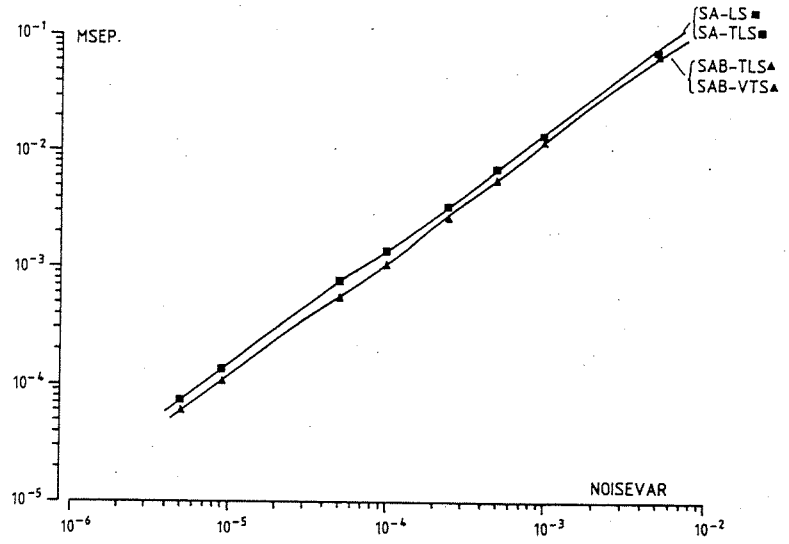


Fig. 6.2.c : The mean squared error of prediction.

$V_r^T [A_\alpha;b]$ has minimal noise sensitivity, approaches on the average more closely the exact solution z_0 and thus minimizes the bias of prediction in (6.12). Hence, the bias is minimized if $\sigma_r[A_\alpha;b]$ is maximal. From theorem 6-1 and (6.5) we know that SAB maximizes $\sigma_r[A_\alpha;b]$ and thus :

$$\sigma_r [A_{SAB};b] \geq \sigma_r [A_{SA};b] \qquad (6.16)$$

Therefore we usually obtain a smaller bias of prediction with SAB-TLS with increasing perturbation rate as shown in figure 6.1.a. As the variance term is still considerably larger than the bias, this improvement in bias only minimally influences the mean squared error of prediction.

If we now increase the noise variance σ_v^2, the r-th singular subspace V_r' of A and V_r of [A;b] are also more and more perturbed. As SA (SA-LS or SA-TLS) and SAB (SAB-TLS or SAB-VTS) base their subset selection on

Table 6.1 : Comparison of the subset selection procedures SA and SAB with respect to the number of times (out of 100) variables are deleted in models Ax = b, perturbed by noise with variance σ_v^2. The exact original model is $(A_0)_{20\times8}x = b_0$ in which each column has length one; a_1, a_2, a_3, a_4, a_5 and a_6 are essentially orthogonal to each other and $b = (10\,a_1 - a_5 - a_6 + a_7 + a_8) / \sqrt{104}$. The two multicollinearities are defined by $-5a_1 - a_2 + a_3 + a_4 - 10a_7 = 0$

$7a_1 + 4a_4 - 10a_8 = 0$

and two columns a_i have to be deleted.

noise variance σ_v^2		columns deleted : a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
.5D-05	SA	100	0	0	0	0	0	0	100
↓	SAB	0	0	0	0	0	0	100	100
.75D-04	SA	100	0	0	0	0	0	0	100
	SAB	0	0	0	0	0	0	100	100
.1D-03	SA	98	0	0	0	0	0	2	100
	SAB	0	0	0	0	0	0	100	100
.25D-03	SA	93	0	0	0	0	0	7	100
	SAB	1	0	0	0	0	0	99	100
.5D-03	SA	81	0	0	0	0	0	20	99
	SAB	4	0	0	0	0	0	96	100
.75D-03	SA	77	0	0	0	0	0	25	98
	SAB	8	0	0	0	0	0	92	100
.1D-02	SA	71	0	0	0	0	0	30	99
	SAB	12	0	0	0	0	0	88	100
.25D-02	SA	56	0	0	0	0	0	52	92
	SAB	19	0	0	0	0	0	85	96
.5D-02	SA	55	0	0	0	0	0	57	86
	SAB	38	0	0	0	0	0	71	91
.75D-02	SA	52	0	0	1	0	0	56	90
	SAB	37	0	0	0	0	0	68	95

those spaces, the subset choices will be more and more affected by those perturbations. Using again theorem 3-5 of section 3.3.1, we know that the sensitivity-to-noise of V'_r of A (resp., V_r of [A,b]) is inversely proportional with σ'_r of A (resp., σ_r of [A;b]). Since $\sigma_r \geq \sigma'_r$ from the interlacing property for singular values, V_r is always less noise sensitive than V'_r. This sensitivity plays an important role if data are heavily perturbed (see section 3.4). This is certainly the case for the r-th singular vector, according to corollary 3-6. Indeed, the lowest singular vector v_r of $V_r[A;b]$ can still be accurately computed while the lowest singular vector v'_r of $V'_r(A)$ may already be entirely worthless. This means that SAB is still able to find the right subset A_α which maximizes $\sigma_r[A_\alpha;b]$, while SA fails in its subset selection. This happens as the noise variance σ_v^2 reaches a certain limit. This limit σ^2_{max} depends on the value of the smallest significant singular value σ'_r of A. From (1.117) we know that the r-th singular triplet $\{\sigma'_r, u'_r, v'_r\}$ of A will be very noise sensitive, hence inaccurately defined when $\sigma'^2_r \approx (R_0 + 1)m\,\sigma_v^2$, where R_0 is the required signal-to-noise ratio. The worst case happens if all noise vectors superimposed on the exact data A_0, are mainly oriented in the direction of u'_r, i.e. the noise matrix $\approx \|E\|_F\, u'_r\, v'^T_r$ with $\|E\|_F^2 = m.n\,\sigma_v^2$ the total noise energy. In this case, the r-th singular triplet is already purely noise if $\sigma'^2_r \approx m.n\,\sigma_v^2$. Hence, $\sigma_r^2(A_0)/(m.n) \leq \sigma^2_{max} < \sigma_r^2(A_0)/m$. This has been confirmed in all our experiments showing an increasing variability and instability in the subset selection by SA, as soon as the noise variance σ_v^2 exceeds its limit $\sigma_r^2(A_0) / (m.n)$. This can be seen from table 6.1 where the number of times each of the r variables are deleted by SA and SAB, are presented. Observe that SAB is more stable in its subset choice w.r.t. any perturbation of the data. This higher stability has been confirmed in most of our experiments.

This superiority effect of SAB w.r.t. SA is more pronounced if $\sigma_r[A;b] / \sigma'_r(A)$ increases. This ratio is mainly determined by the orientation and length of the response vector b (see section 3.4).

6.4. CONCLUDING REMARKS

In this chapter, the problem of **subset selection in case of errors in all variables**, is studied. A **new subset selection algorithm SAB-TLS based on TLS**, is presented and compared with the subset selection algorithm SA-LS of Golub, Klema and Stewart. Their respective **selection properties and accuracy in parameter estimation and prediction** are investigated. Hereby, only cases in which one wants to utilize the subset equation to predict the response from exact values of the predictor variables, e.g. in hypothetical and simulation studies, are considered here.

The SA-LS algorithm is recommended if the variables in the data matrix A are assumed to be free of error. As soon as the variables are unobservable and only perturbed data are available, the TLS technique should be used in order to obtain the most accurate estimates for the parameters in any selected subset model. For sufficiently small perturbations of the model Ax = b, subset selection based on the r-th right principal subspace V'^T_r of A (SA) enables a slightly better prediction accuracy, in case of bad subset condition. On the other hand, well-conditioned datasets must always base their selection on the r-th right singular subspace V^T_r of [A;b] (SAB) in order to obtain the best subset and prediction accuracy (SAB-TLS). However, as soon as the variance of the errors present in the model reaches its limit depending on the magnitude of the singular values of A, subset selection based on V^T_r of [A;b]is superior in prediction accuracy for all cases (SAB-TLS). Moreover, SAB takes into account the extra information provided by the response b and guarantees a higher and longer stability and reliability of its subset choice, when errors occur in all variables. Because of its computational efficiency, a variant of the subset selection algorithm SAB-VTS, which enables the same prediction accuracy has been derived and is recommended.

CHAPTER 7

USE OF TLS IN IMPULSE RESPONSE ESTIMATION BY DISCRETE DECONVOLUTION

7.1. INTRODUCTION

Many problems in a variety of fields like signal processing, system theory, automatic control and in general engineering, physics, economics, ... give rise to an overdetermined set of linear equations AX = B which are usually solved with the ordinary LS technique. Very often, the data matrix A as well as the observation matrix B are inaccurate. Hence, the use of TLS may be recommended. In this and the following chapter we explore the **applicability of TLS in the field of time domain system identification and parameter estimation.** These are intended as illustrations of ways how one can proceed in order to obtain better results with TLS. It is expected that TLS can be used successfully in many domains and applications.

A wide variety of different models are used to describe a systems behaviour in the time domain [71-9-32]. Perhaps the simplest model of all is to relate the input sequence u(t) to the output sequence y(t) in a **difference equation.**

$$y(t) + \sum_{k=1}^{na} a_k\, y\,(t-k) = \sum_{k=1}^{nb} b_k\, u\,(t-k) \qquad (7.1)$$

(7.1) is also called a **transfer function model** and can be rewritten as :

$$y(t) = \varphi^T(t)\ \theta \qquad (7.2)$$

with $\theta^T = [a_1, \ldots, a_{na}, b_1, \ldots, b_{nb}]$ and $\varphi^T(t) = [-y(t-1), \ldots, -y(t-na), u(t-1), \ldots, u(t-nb)]$. As an alternative to the difference equation model (7.1), we could use an **impulse response model** :

$$y(t) = \sum_{k=0}^{n} h(k)\ u(t-k) \qquad (7.3)$$

Here h(t) is the impulse response of the system at the time instant t. The model parameters θ in (7.2) and h(t) in (7.3) are to be determined by the identification procedure. The usual way is to observe the input u(t) and output y(t) over a certain interval of time t = 1, ..., m. Writing (7.2) and (7.3) for t = 1, ..., m, one obtains a set of linear equations of the form :

$$Y = UH \qquad (7.4)$$

with $Y^T = [y(1), \ldots, y(m)]$, U the data matrix obtained from (7.2) or (7.3) and H the unknown model parameters θ or h(t). By taking enough data one can increase the accuracy. If m > n, (7.4) is overdetermined. If neither the input nor the output can be observed exactly, then Y as well as U in (7.4) are inaccurate. Keeping the properties of TLS in mind, it is straightforward to use here the TLS technique. In chapter 8 we study the applicability of TLS in transfer function models of the form (7.1).

In this chapter the impulse response model (7.3) is considered. The identification process, computing the impulse response from the discrete input-output data, is called **discrete deconvolution.** The impulse response characterizes entirely the input-output behaviour of a system, i.e. it contains all the information about the system required to compute its response or output on every possible input. This is done by "convolving" every new input with the impulse response. This technique is of paramount importance in the study of mutual interrelations in a system. In fact, it is difficult to analyze the internal relations of many industrial and biological systems (called "black box" systems) and describe them mathematically. Under certain conditions (see section 7.2), the impulse response **allows to study the input-output behaviour of a system and its interaction with other systems.** Since it only requires the knowledge of the response of the system to an impulse, it is called a **black box technique.**

This chapter is organized into 4 sections. Section 7.2 describes the principles of deconvolution and its underlying assumptions. The advan-

tages of TLS w.r.t. LS in discrete deconvolution are illustrated with a simulation example. In section 7.3, the applicability of TLS as a deconvolution technique in renography is discussed and demonstrated. By deconvolving the activity-time curves obtained from the blood and kidney, the renal retention function can be calculated, yielding useful clinical data. However, the current deconvolution techniques in renography, based on back substitution or Gauss elimination using the smoothed curves, are not very reliable and are very sensitive to the inaccuracies present in the data. They are compared with TLS w.r.t. their accuracy. Results from simulation, as well as from clinical data, are presented to show the advantages of the use of TLS w.r.t. noise rejection in the data. It is concluded that TLS needs no smoothing and is a powerful, efficient, reliable and stable deconvolution technique. Finally, section 7.4 summarizes the main conclusions.

7.2. DISCRETE DECONVOLUTION

The response y(t) of a **linear, time invariant, causal system with zero initial state** and impulse response h(t), on any input u(t) can always be computed by [119] :

$$y(t) = \sum_{k=0}^{t} u(k)\, h(t-k) \qquad (7.9)$$

(7.9) is also valid for multivariable systems with n_i inputs, $n_i \geq 1$, and n_o outputs, $n_o \geq 1$. y(t) (resp., u(t)) is an n_o (resp., n_i)-dimensional row vector and the terms h(k) of the impulse response of a multivariable system (also called Markov parameters) are n_o by n_i matrices. The i-th column of every term h(k) represents the response of the system to an impulse at its i-th input and a null sequence at the other inputs. If the impulse response h(t) and input u(t) of a system are known over a certain time interval t = 0, ..., m, we can compute its response y(t), t = 0, ..., m from (7.3). Or, in matrix form :

$$Y = U.H \quad \text{with} \quad Y = \begin{bmatrix} y(0) \\ \vdots \\ y(m) \end{bmatrix}, \quad H = \begin{bmatrix} h(0) \\ \vdots \\ h(m) \end{bmatrix} \quad \text{and} \quad U = \begin{bmatrix} u(0) & 0 & \cdots & 0 \\ u(1) & u(0) & & \vdots \\ \vdots & & \ddots & 0 \\ u(m) & \cdots & u(1) & u(0) \end{bmatrix} \qquad (7.10)$$

Consider now the reverse process, called **deconvolution**, i.e. we assume the input u(t) and corresponding output y(t) of a system to be known and want to compute its impulse response h(t). Then the system will be **identified** and we are able to compute its response on any new input signal. Assuming a linear, time invariant, causal system with zero initial state this can be done by **solving the convolution equations** (7.10). Notice however the fact that (7.10) is a collection of n_o sets of m+1 linear equations each in $n_i(m+1)$ unknowns. This implies that as soon as the number of inputs n_i is larger than one, these sets of equations as such are underdetermined, hence have no unique solution, even for very high m. Only when **the system has a finite state dimension**, i.e. the order of the system is bounded, a fixed, finite number (say n+1) of impulse response samples can be guaranteed to describe completely the exact system behaviour. When m is sufficiently high, the sets of equations become finally overdetermined and thus guarantee a unique and exact solution in the case of exact data.

The requirement of finite dimensionality is a **weak requirement** in the sense that strictly speaking all real systems are of infinite dimensional dynamic behaviour, but for most of them the pertinent behaviour can be described sufficiently accurately by a finite dimensional model behaviour. In the same spirit, the system should be time invariant and linear but slow time variance and slight nonlinearities are admissible if they don't essentially influence the system behaviour and if their effect is within the ranges due to data imprecision and/or allowed model inaccuracy.

If all the above assumptions are valid, then the solution of

$$Y = U.H \quad \text{or} \quad \underset{(m+1)\times 1}{\begin{bmatrix} y(0) \\ \vdots \\ y(m-1) \\ y(m) \end{bmatrix}} = \underset{(m+1)\times n_i}{\begin{bmatrix} u(t-m) & \cdots & u(t-n-m) \\ \vdots & & \vdots \\ u(t-1) & \cdots & u(t-n-1) \\ u(t) & \cdots & u(t-n) \end{bmatrix}} \underset{(n+1)\times 1}{\begin{bmatrix} h(0) \\ \vdots \\ h(n) \end{bmatrix}} \qquad (7.11)$$

computes the impulse response of a system (U is (block) Toeplitz). In this way, the **discrete deconvolution problem is essentially reduced to the problem of solving a finite set of linear equations** (7.11). In order to obtain a matrix U of full rank, the input signal must be persistently exciting (see also section 4.5), e.g. white noise. The right handside Y, as well as the left handside U, of (7.11) are affected by sources of un-

predictable inaccuracies, e.g. due to measurement errors. This suggests solving (7.11) with TLS (multidimensional case) (see section 1.4). In order **to obtain a strongly consistent estimate** of the impulse response of the unperturbed system, the errors in [U;Y] of (7.11) must be row-wise independently and identically distributed with zero mean and common covariance matrix σ_v^2 I. From [26] we know that this is satisfied if **the error sequences** added to the n_i input sequences $u_i(t-m-n)$, ..., $u_i(t)$, $i = 1, \ldots, n_i$ **are stationary, zero mean white noise sequences.** If these conditions are met, all results and conclusions of chapter 3 and 4 are directly applicable.

This is confirmed by the following **simulation example.** Consider a system with exact impulse response as illustrated in fig. 7.1.a. The input is fed with white noise. Solving (7.11) with m = 29, n = 19 and $t \geq n+m$, one obtains the impulse response which is located close to the right singular vector of U corresponding to its minimal singular value. Hence, the differences between TLS and LS will be more pronounced (see section 3.4). Using the same simulation procedure of section 3.4, one obtains fig. 7.1.b comparing the relative errors of TLS and LS in function of the noise variance σ_v^2. Observe that the error of TLS is 13 % smaller than that of LS.

Finally, we stress the fact that TLS computes an estimate of the exact **impulse response of the unperturbed system.** Hence, it is particularly suitable when we are interested in the properties and response behaviour of the unperturbed system but it may not be used for real-time prediction of future responses from new noisy inputs (see section 6.1).

7.3. DECONVOLUTION OF THE RENOGRAM

In this section we want to study the applicability of TLS as deconvolution technique in renography [120-128].

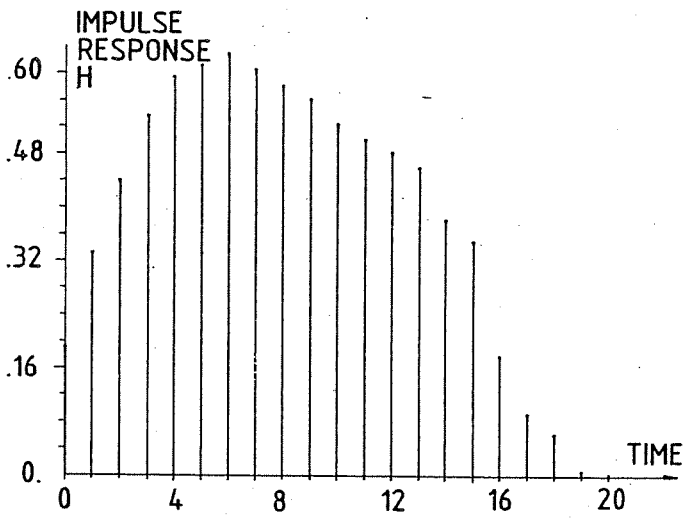


Fig. 7.1.a : The exact impulse response H_o.

7.3.1. THE RENAL RETENTION FUNCTION

In renography an intravenous injection of a radioactive tracer that is cleared by the kidneys is administered and the variation of radioactivity with time over each kidney is visualized in the renogram. This curve is a very simple and elegant aid in the diagnosis in urologic and nephrologic pathology and the study of renal function in general. However, the influence of the tracer input curve on the renogram represents an unwanted element. To eliminate this influence, the renogram is deconvolved to obtain the **renal retention function** yielding more precise and useful clinical data.

Deconvolution analysis of the renogram is based on modelling the kidney as a linear, time invariant, causal system with zero initial state, as assumed in section 7.2. **Causality** is trivial. **Linearity** is generally

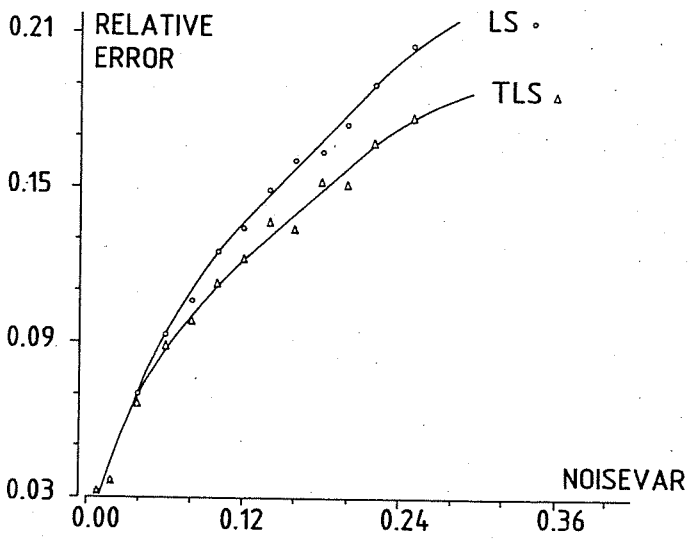


Fig. 7.1.b : Comparison of the average relative error of the LS and TLS impulse response versus the variance of the noise added to the set of convolution equations $(U_o)_{30\times20}\ H_o = Y_o$. U_o is the Toeplitz matrix of a white noise input signal.

satisfied, but **time invariance** may often be violated [76]. The last assumption which is never mentioned although it is very important, implies that the system starts from the **zero initial state**. From a practical point of view this means that there is no radioactive material left in the body of the patient before the beginning of the test [120]. If these conditions are satisfied, the desired retention function can be calculated. The input of the system is the arterial concentration of a radioactive tracer as a function of time. The unit **impulse response is the desired retention function of the kidney** and represents the form of the renogram that would be obtained if an injection had been given directly into the renal artery. The use of deconvolution analysis for the estimation of renal retention of a radioactive tracer has been widely described in the literature [34-63-131-28]. The relative (e.g. left versus right) effective renal plasma and relative glomerular filtration rate may be estimated from the radionuclide renogram using this technique. Unfortunately the current deconvolution techniques are all very sensitive to the inaccuracies inherently present in the data.

In the discrete time case the retention function H(t) is found by solving the deconvolution equations :

$$I.H = R \qquad (7.12)$$

$$\text{with } I = \begin{bmatrix} I_1 & 0 & \cdots & 0 \\ I_2 & I_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & I_1 \\ I_m & I_{m-1} & \cdots & I_{m-n+1} \end{bmatrix}, \quad R = \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_m \end{bmatrix}, \quad H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_n \end{bmatrix} \quad \text{and } m > n$$

I is an m by n Toeplitz matrix, containing the m activities I_i which determine the activity-time curve I(t) obtained from the blood. I_i is the number of counts registered in the vascular region during the i-th sampling interval. The m-dimensional vector R determines the renogram R(t) and contains the activities R_i, measuring the number of counts registered in the kidney region during the i-th sampling interval. The desired retention function H(t), represented by the n-dimensional vector H, is derived by computing the count rate H_i over each sampling interval.

The technique, most often used in renal deconvolution for the solution of (7.12), is the **matrix algorithm**, because it does not assume previous knowledge of the shape of the data [116]. This method always assumes m = n. The samples H_i of the retention function are then evaluated directly from (7.12) by back substitution :

$$H_i = (1/I_1)\left[R_i - \sum_{j=1}^{i-1} I_{i-j+1}\, H_j\right] \qquad (7.13)$$

or preferably, as done in this section, by Gauss elimination with pivot-

ing [48, section 4.4]. If I and R are accurately known and I is nonsingular, the method will work perfectly. However, in reality, data are always noisy because of the lack of resolution and contrast in nuclear images, the blood and tissue background, as well as incorrect delineation of the regions of interest. This means that the errors in R_1 and I_1 will produce a corresponding error in H_1. This error in H_1 will affect the value of H_2 and errors in H_1 and H_2 will affect H_3, and so on. As it is well-known in numerical linear algebra, this method is therefore very **sensitive to errors**, since they propagate through the whole calculation of H. This **loss of accuracy** in the calculated retention function has hindered clinical acceptance of the technique.

The usual way to counteract this problem is to manipulate the original data by **smoothing** [94], data bounding [27], exponential [131] and spline fitting [89]. All those noise filtering techniques have been compared by Fleming and Kenny [35]. Many techniques have been tested. All noise reduction methods manipulate the raw data from the patient in such a way that indeed smooth curves are obtained but the original information is thereby distorted. It is more reasonable to handle the original data in an efficient and reliable way, than to submit the raw data to a technique which is very sensitive to inaccuracies. Since TLS operates directly on the **original** data and is able to reject noise **without error propagation**, its use as deconvolution technique in renography will be explored now.

7.3.2. THE USE OF TLS IN RENOGRAPHY

One obvious way of reducing noise and improving the accuracy of the retention function is to make the set of equations of the discrete deconvolution (7.12) **overdetermined**, i.e. $m > n$. Indeed, as the mean transit time of Hippuran is about 2.3 minutes and that of Tc-99m DTPA about 3 minutes in normal kidneys, the retention function can die out after a few minutes. For those cases, it is unrealistic to calculate the renal impulse response over the whole observation period as the last minutes are pure noise and therefore not relevant. Hence it is obvious to overdetermine the set of equations. The relevant part of the retention function, e.g. the first 10 minutes, can then be obtained by solving (7.12)

with the least squares method. Since all measurements I_i and R_i are noisy, the left handside I as well as the right handside R of the set of equations (7.12) are inaccurate. Hence the use of TLS is appropriate. In addition to the properties of TLS w.r.t. LS discussed in chapter 3 and 4, also other important advantages may confirm the preference of TLS over LS in renography.

- TLS only needs the calculation of the right singular subspace of $[I;R]$, corresponding to its smallest singular values. Therefore the computation time for TLS can be made smaller than for LS (see chapter 9 and 10).

- Moreover, as shown in section 3.4, the accuracy of the TLS solution improves as the number of right handsides d of the set of equations increases, while the accuracy of the LS solution is independent of d. Hence, when more retention functions (e.g. using regional renography) are described with a same blood activity-time curve I(t), TLS can reject more noise and improve the accuracy of the retention functions.

However, we have to be **careful** in order to **apply TLS correctly in practice**. Indeed, an incorrect determination of the parameters like scaling, appropriate length of the retention function, background subtraction, rank determination, ... affects the accuracy of the results obtained with TLS. The same applies for LS except the fact that LS is not influenced by scaling. Therefore, in order to use all the promising advantages of TLS in renography, the following problems have to be studied carefully : noise level determination, scaling, determination of an appropriate length n of the retention function, data handling (e.g. good background subtraction), ... Let us have a look to each of these problems.

7.3.2.1. Noise level determination

It is known that the statistical errors in the count rates registered with the gamma camera have a Poisson distribution. Therefore, in clinical practice it is assumed that the count rates are affected by noise with zero mean and standard deviation equal to their square root. A good

estimate for the **variance of the statistical noise** on each **original** data curve is found by taking the root mean square of the curve or :

$$(\sigma_v)^2_{C_0} = \sqrt{\sum_{i=1}^{m} (C_0)_i^2 / m} \qquad (7.14)$$

with $(C_0)_i$ the activity after i time intervals measured over the region of interest (ROI) C_0.

In order to calculate now the desired retention function we have to calculate the **net curves**. The usual procedure is to **subtract the background** from the original data scaled with a factor equal to the ratio of the ROI surface S_C to the background region surface S_{BG}. Due to this subtraction, noise on the background data is added to the original data relative to the ratio of their surfaces. A good estimate for the **variance of the statistical noise** on each **net** curve is given by :

$$(\sigma_v)^2_C = (\sigma_v)^2_{C_0} + (S_C / S_{BG})^2 \, (\sigma_v)^2_{BG} \qquad (7.15)$$

with C (resp., C_0) the net (resp., original) activity curve.

The above discussion assumed **time invariance** of the retention function. However, this assumption may not be valid due to the occurrence of physiological variations such as pelvo-ureteric contractions. The effect of time variance, known as **physiological noise**, has been studied by Fleming and Kenny [35]. The efficiency of TLS in the rejection of physiological noise has not yet been fully analyzed. However, in much the same way as smoothing and other filtering techniques reduce both statistical and physiological noise [35], it is expected that an appropriate rank determination in the TLS technique will be successful in the rejection of statistical as well as physiological noise.

7.3.2.2. Scaling

Scaling has to be performed carefully in advance in order to apply TLS correctly. According to the scaling strategy recommended in section 1.8.1, we have to scale the entries in [I;R] of (7.12), or equivalently **the activities in I and R, so that all error estimates are** as

closely as possible **equal**. One way of scaling which approximates this situation, consists in taking one curve as reference and scaling the other ones with a factor equal to the ratio of their **maxima**. This way of scaling produces a TLS solution with norm close to one. As the ratio of the maxima of two curves approximates the ratio of the **surfaces** of their regions, the scaled net blood activity-time curve I and renogram R will have comparable count rates. From (7.15), it follows then that all data in (7.12), scaled in this way, will have approximately the same accuracy level. In the experiments performed with clinical data, this scaling method produced most satisfactory results.

7.3.2.3. Determination of an appropriate length of the retention function

Let t_R (resp., t_I) be the time instant at which the net renogram (resp., net blood activity-time curve) drops below the noise level, defined by $(\sigma_v)_C$ in (7.15). This happens when the activities, registered in the kidney (resp., vascular) region per unit surface, become comparable to the activities in the background region per unit surface. Define further Δt as the sampling time interval and t_H as the time instant at which the retention function H has died out. Using the convolution equations (7.12), it is not difficult to prove now that :

$$t_H = t_R - t_I + \Delta t \qquad (7.16)$$

As $t_H = n \, \Delta t$, the appropriate length n follows immediately.

If the net renogram does not drop below the noise level at the end of the observation period, then t_R cannot be defined. In this case a reasonable estimate of n with $(t_R - t_I + \Delta t) / \Delta t \leq n < m$ can be deduced from clinical information about the patient's renal disease. If no information is available , n is set equal to m.

7.3.2.4. Data handling

If the input function is taken from measurements over an input region containing blood arteries near the kidney, it is easily proven [120] that the result obtained by deconvolving the original data will not agree with the desired retention function. Hence background subtraction is required, but subtracting the background (with an appropriate scaling factor) directly from the original data, induces much noise. This problem could be avoided if the heart is taken as input region. However, the gamma camera used in the actual measurement set up (section 7.3.3.2), can usually not display the heart and kidney(s) simultaneously. Therefore methods, which subtract the background in a more reliable way have to be analyzed. The powerful singular value decomposition (SVD) can again be used for the separation of the curves. One promising method has been proposed in [120] but has not yet been fully analyzed.

7.3.3. COMPARING THE MATRIX ALGORITHM AND TLS

7.3.3.1. Simulation results and discussion

If we want to evaluate deconvolution techniques we need an exact reference situation for which the methods give the exact and unique solution. If noise is added to the exact data we are able to compare the exact solution with the solutions obtained with each technique. We therefore simulate the whole process and take, as reference situation, the ideal curves.

Simulation set up :

We start from an exact, possibly overdetermined set of m convolution equations (7.12) in n unknowns $I_o H_o = R_o$. I_o is the m by n Toeplitz matrix generated from m samples I_i of the exact blood activity-time curve. R_o is the m by d matrix whose columns are the m samples of the d exact renograms. H_o contains the n samples of the d exact retention functions. Noise with zero mean and variance σ_v^2 is then added to the exact data I_o and R_o. The exact blood activity-time curve and renograms with added statistical noise can be smoothed, i.e. each point on the curve is replaced by a weighted mean of the point itself and the two neighbouring points, the first and last point being held constant. The degree of smoothing is controlled by the number of smoothings carried out. Call I the m by n Toeplitz matrix generated from the noisy, possibly smoothed blood activity-time curve and call R the m by d matrix generated from the noisy, possibly smoothed renograms.

We then solve the set of convolution equations I.H = R to obtain the d retention functions H. Call H_{TLS} the retention functions computed with TLS, described in section 1.8.1, and H_{MA} the retention functions computed with the matrix algorithm described in section 7.3.1. The relative errors

$$\|H_{MA} - H_o\|_F / \|H_o\|_F \quad \text{and} \quad \|H_{TLS} - H_o\|_F / \|H_o\|_F \tag{7.17}$$

are averaged over 100 sets of m equations affected by noise with equal variance σ_v^2. We repeat the simulation for larger noise variances σ_v^2 and different number of smoothings. The average relative errors are then plotted versus the noise variance (see fig. 7.1 and table 7.1).

Simulation example :

The exact blood activity-time curves $I_o(t)$ are generated from results of three-exponential analysis [131] where the form of the curve is assumed to be

$$I_o(t) = A \exp(-a_1 t) + B \exp(-a_2 t) + C \exp(-a_3 t) \tag{7.18}$$

The values of the rate constants and the contributions of the different components, computed by Fleming [33] for normalized background subtracted Hippuran cardiac clearance curves, were used in our simulations :

$$A = 40.3,\ B = 45.2,\ C = 15.5,\ a_1 = 1.8,\ a_2 = 0.43,\ a_3 = 0.035$$

The exact retention functions remain at a constant value K until the renal minimum transit time t_1 is reached. They then begin to decrease, reach zero at the maximum transit time t_2 and thereafter remain zero. Mathematically, those curves are represented by :

$$H_o(t) \begin{cases} = K & 0 \leq t < t_1 \\ = K.(t - t_2) / (t_1 - t_2) & t_1 \leq t < t_2 \\ = 0 & t \geq t_2 \end{cases} \tag{7.19}$$

Given the exact blood activity-time curve and retention functions, the exact renograms are directly derived from the convolution equations $I_o H_o = R_o$. Data are taken from frames at $t = 20$ sec. intervals over an observation period of 20 min. ($m = 60$). As the retention functions die out at their maximum transit time t_2, we only need to calculate their relevant part. Hence, n is determined by $t_2/\Delta t$.

Taking the blood activity-time curve (7.18) and one ($d = 1$) retention function (7.19) with constant value $K = 1$, $t_1 = 3$ min. and $t_2 = 5$ min. ($n = 15$), one obtains the results shown in table 7.1 and fig. 7.2. The simulations were performed with the noisy curves and the same curves smoothed once, twice and four times. From table 7.1 and fig. 7.2, one observes that :

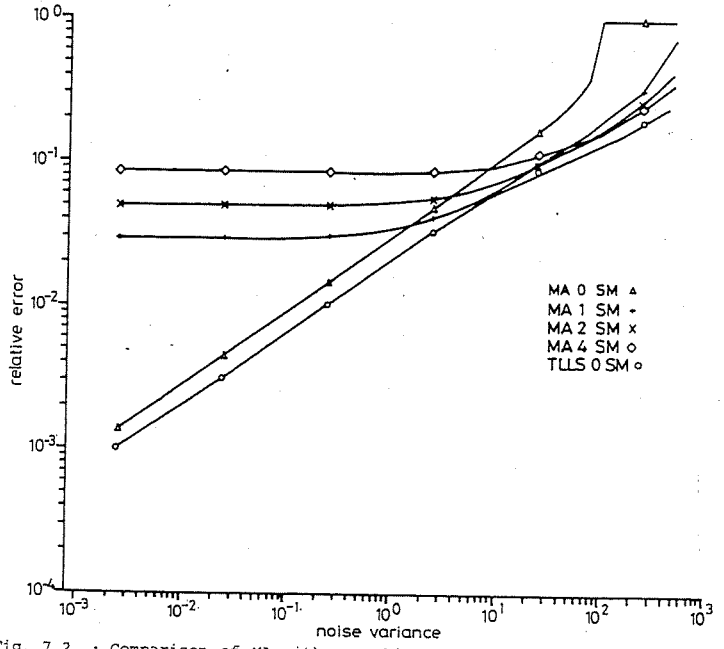


Fig. 7.2. : Comparison of MA with smoothing and TLS without smoothing : relative error versus noise variance of simulation example (log-log diagram). Errors larger than one are set equal to one. 0, 1, 2, and 4 correspond with the number of smoothings carried out on the curves.

Table 7.1 : Average relative errors of MA and TLS in function of the standard deviation σ_v of the added noise. 0, 1, 2 and 4 correspond with the number of smoothings carried out on the curves. The average rank r of the matrix [I;R] involved, computed by TLS, is given too.

σ_v	$\|H_{MA} - H_o\|_F / \|H_o\|_F$				$\|H_{TLS} - H_o\|_F / \|H_o\|_F$				rank r of [I;r]
	0	1	2	4	0	1	2	4	
.050	.0013	.0290	.0491		.0010	.0301	.0541		15
.071	.0019	.0290	.0491		.0015	.0301	.0541		15
.087	.0024	.0290	.0491		.0018	.0301	.0541		15
.100	.0027	.0290	.0491		.0020	.0301	.0540		15
.158	.0044	.0292	.0492		.0031	.0302	.0541		15
.224	.0066	.0295	.0494		.0045	.0304	.0543		15
.274	.0079	.0295	.0494		.0056	.0304	.0543		15
.316	.0093	.0297	.0495		.0063	.0303	.0541		15
.500	.0146	.0304	.0498		.0102	.0303	.0539		15
.707	.0204	.0314	.0500		.0142	.0308	.0539		15
.866	.0255	.0328	.0510		.0178	.0321	.0547		15
1.00	.0297	.0346	.0520	.0841	.0200	.0331	.0554	.0962	15
1.58	.0490	.0414	.0557	.0855	.0331	.0371	.0576	.0970	15
2.24	.0681	.0505	.0617	.0890	.0451	.0426	.0607	.0996	15
2.74	.0859	.0586	.0672	.0905	.0559	.0505	.0666	.0994	15
3.16	.0998	.0654	.0721	.0937	.0652	.0548	.070	.1029	15
5.00	.1667	.0989	.0975	.1143	.0946	.0775	.0858	.1112	13
7.07	.2584	.1342	.1246	.1332	.1025	.0937	.0965	.1149	7
8.66	.3701	.1622	.1457	.1497	.1146	.1091	.1108	.1184	5
10.0	1.04	.1928	.1722	.1630	.1199	.1190	.1200	.1281	4
15.8	99.5	.3194	.2653	.2426	.1925	.1933	.1952	.1992	2
22.4	71015.	.7312	.4440	.3560	.2563	.2513	.2537	.2523	1

- **The matrix algorithm MA is less accurate than TLS for all noise variances, even with smoothing.** Smoothing can reduce significantly errors on deconvolution with MA, as already observed before [35], but the results obtained with TLS and without smoothing are still better. From Table 7.1, one can observe that smoothing only slightly improves the accuracy of the deconvolution with TLS at some specific noise variances. The maximum improvement in accuracy occurs at a standard deviation $\sigma_v = 5$ where 2 % is gained by smoothing the curves once.

- If **more noise** is added to the data, **TLS will lower the rank r of** [I;R] corresponding to (1.112). The TLS solution is then calculated using the rank r approximation of the noise matrix [I;R]. Due to this adaptive rank determination, noise can be rejected and an accurate, stable solution can be obtained. The **MA, however, cannot handle rank-deficient problems.** Hence it is not able to reject the noise and the solution becomes quickly unstable and inaccurate. From a numerical point of view this is due to the error propagation mechanism of the algorithm.

- If an appropriate **number of smoothings** is carried out, the accuracy of deconvolution with MA can improve for high noise levels. As can be seen in fig. 7.2, **TLS is only slightly better than MA used with optimally smoothed curves.** However, the choice of the optimal number of smoothings may be **critical** [35]. If the number of smoothings is too small, noise is not sufficiently removed from the data. But if the number of smoothings is too high, original information is removed and the errors on the deconvolution increase. This is illustrated in fig. 7.2, showing how smoothing affects the accuracy of deconvolution for small noise variances σ_v^2. The estimation by inspection of an experienced observer as to the number of smoothings required is one way of achieving close to optimal use. It is clear that the **adaptive rank determination** used by **TLS** and based on (1.112), does not require any interaction from the user.

- Moreover, if **the matrix** I of the convolution equations $I.H \approx R$ is (nearly) **singular or rank-deficient, MA fails** to compute the retention functions. This can be seen from table 7.1. Generically, singular equations will not occur but nearly singular or rank-deficient ma-

are not uncommon. The high relative errors (> 100 %) obtained with the MA are caused by solving noisy sets whose I matrix is nearly singular. In some cases smoothing can help to remove the singularity but this is not generally the case. For TLS singularity or rank deficiency causes no problem because of its ability to determine the rank adaptively. This means that **MA can solve most but not all deconvolution problems** and hence, depending on the data, retention functions of some patients in clinical practice cannot be obtained. TLS however, is always able to compute an accurate, reliable and numerically stable retention function.

7.3.3.2. Clinical results and discussion

The study is performed using data collected at the division of Nuclear Medicine of the University Hospital of Leuven Gasthuisberg. They confirm the simulation results and proposed methods.

Procedure :

The radiopharmaceutical Tc-99m DTPA or I-123 Hippuran is injected into an antecubital vein and sequential 20 s frames are recorded on disk of a HP 1000 minicomputer during 30 minutes (m = 90). At the end, regions of interest (ROI) are drawn around each kidney, the blood artery in the vicinity of the kidneys, the bladder and also around renal subregions. A representative blood and tissue background area is selected and the desired ROI activity-time curves are generated. Several smoothings are carried out on the curves. After subtracting the background in the usual way (section 7.3.2.1), the corresponding retention functions are calculated from the convolution equations (7.12) with TLS and the matrix algorithm (MA).

The original blood, renal and background activity-time curves obtained in the study of a patient injected with I-123 Hippuran are shown in fig. 7.3. From fig. 7.4, 7.5 and 7.6, the following observations are made.

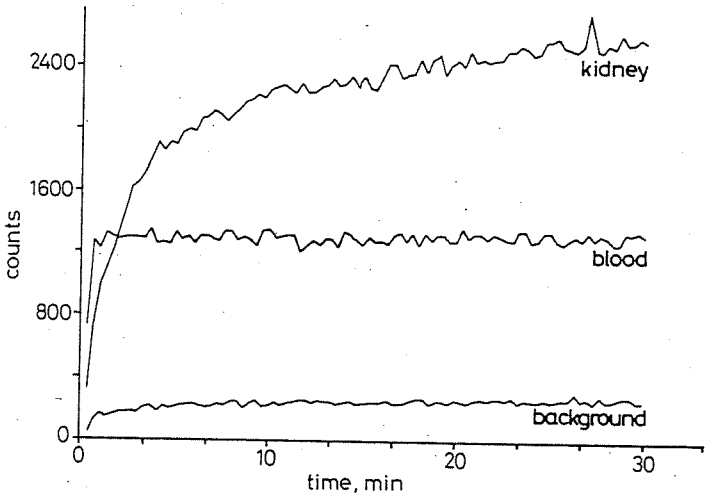


Fig. 7.3 : Original curves of study of a patient injected with I-123 Hippuran.

- By scaling the curves (section 7.3.2.2) and applying the usual background subtraction procedure (section 7.3.2.1), the net curves are obtained. The retention function obtained by deconvolving the net curves with TLS is plotted in fig. 7.4. The computed rank is 13. As the renogram increases steadily, hereby indicating **bad renal clearance**, the retention function does not drop to zero and we must take its length equal to the length of the observation period.

- Calculating the retention function by deconvolution of the net curves with MA and **without smoothing** we obtain a worthless solution diverging and oscillating because of near singularity of the left handside matrix I of the set of equations (7.12) (the minimal singular value is 0.236). The relative error $\|R - I.H\|_F \; / \; \|R\|_F$ obtained with TLS is 3.3 %. With MA one obtains more than 100 % error.

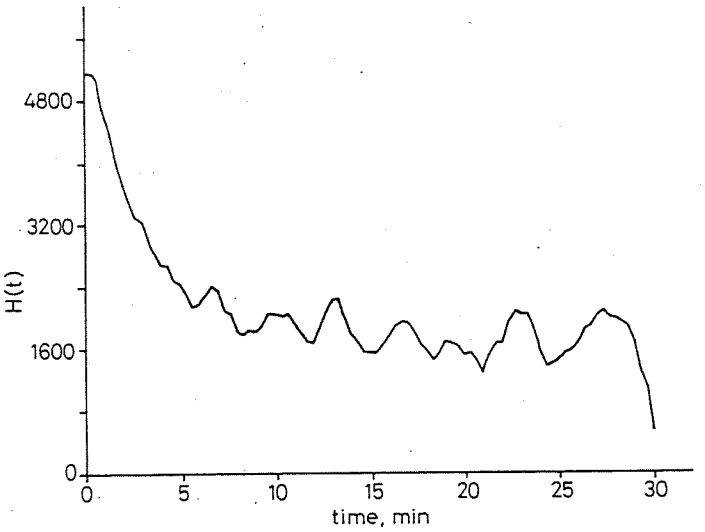


Fig. 7.4 : Retention function of study of a patient injected with I-123 Hippuran, obtained by deconvolution with TLS of the original curves in fig. 7.3 after background subtraction.

- By **smoothing** the original curves in fig. 7.3 once, twice and four times and by recomputing the retention function, we observe that the retention function obtained with TLS only undergoes some minor changes (fig. 7.5 and fig. 7.6). As the Toeplitz matrix I obtained from the smoothed data, is no longer singular (the minimal singular value is now 243), the retention function with MA can be computed. We observe from fig. 7.5 and 7.6 that smoothing reduces significantly the errors. After four smoothings the retention function of MA approaches the result of TLS but still more smoothings are needed to remove the noise with the risk of removing simultaneously the original information.

- In another study of a patient injected with Tc-99m DTPA, the Toeplitz matrix I was highly singular (minimal singular value = 0.122×10^{-12})

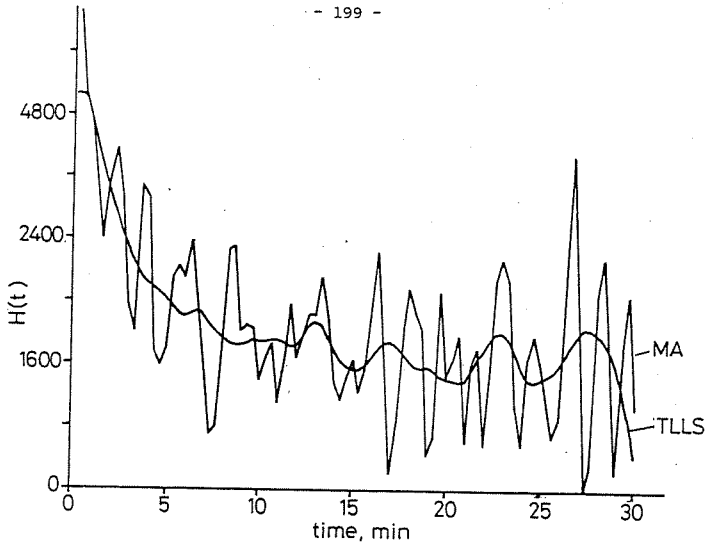


Fig. 7.5 : Retention function of study of a patient injected with I-123 Hippuran, obtained by deconvolution with TLS and MA of the original curves in fig. 7.3 after one smoothing and background subtraction.

[120]. In this case, smoothing could not remove the singularity and MA was unable to compute the retention function. TLS however, reduced the rank correctly and computed an accurate retention function. For brevity reasons these results are not included in this thesis.

7.3.4. EVALUATION OF THE MATRIX ALGORITHM AND TLS

Analyzing the results we can draw the following conclusions and make useful suggestions for future research. It is proven that **TLS is useful as deconvolution technique in the calculation of the renal retention function.** It is more powerful, efficient, reliable and stable than the matrix algorithm MA, even after smoothing. As MA can only handle full rank problems, it is not able to compute the retention function from singular convolution equations (7.12) and fails in those cases. TLS however, can easily reduce the rank of the matrix [I;R] involved and compute a reliable retention function.

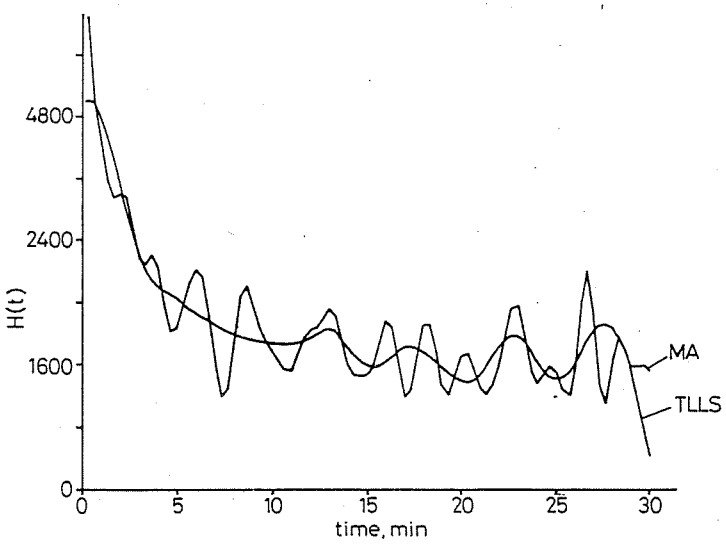


Fig. 7.6 : Retention function of study of a patient injected with I-123 Hippuran obtained by deconvolution with TLS and MA of the original curves obtained in fig. 7.3 after four smoothings and background subtraction.

As smoothing does not improve the accuracy of the retention function computed **with TLS, the original activity-time curves need not be smoothed.** Hence, TLS needs less interaction from the user than MA for which smoothing is necessary in order to reduce significantly the errors of deconvolution. When the optimal number of smoothings is chosen, TLS remains slightly better in accuracy than MA.

When the appropriate length n of the retention function and the rank of [I;R] are chosen, the obtained accuracy and noise rejection effect of TLS is at its best. However, **a more fundamental study of the noise**

(statistical and physiological) **is needed** in order to find a robust strategy for calculating directly the optimal rank and length n of the retention function from the original data and noise level.

The length n chosen for the retention function is not so crucial with respect to the obtained accuracy. If we choose n too long, then the last irrelevant part is pure noise and oscillates around the time axis within the noise margin. However, care should also be taken not to choose n too small, since this would severely affect the accuracy. Most crucial is the rank of [I;R]. Hence it must be calculated carefully.

The **usual background subtraction procedure can induce too much noise** on the data. A more reliable, robust and generally applicable background strategy is therefore needed. For this purpose the SVD can be taken as a useful tool.

The accuracy of the retention function can be further improved in the following way. First one can find an optimal compromise between the length of the sampling interval and the errors on the data w.r.t. the resolution of the computed retention function. A shorter sampling interval allows to compute more samples of the retention function (higher resolution), but at the same time the statistical errors on the data increase. Second one can choose another site for the measurement of the blood activity-time curve, e.g. over the heart. Since this does not require background subtraction, the accuracy will be improved as argued by Reeve and Crawley [86] and Diffey et al. [28].

Further, errors in the data are introduced by **incorrect delineation of the regions of interest** (ROI). The tracing may be particularly difficult in the case of transplant kidneys. Moreover, the choice of a renal ROI encounters the problem of superposition between the renal image and that of the vessels, excretion pathways and even the bladder. This results in an imprecise determination of renal and vascular activities and thus occasionally in a poor determination of renal transit time. These problems have been partially solved with factor analysis [10] but can also be handled with image processing techniques based on the SVD.

7.4. CONCLUDING REMARKS

In this chapter, the **applicability of TLS in the estimation of the impulse response** of a system is studied. The impulse response characterizes entirely the input-output behaviour of a system and is of paramount importance in the study of internal relation in black box systems. If a process can be modelled as a linear, time invariant, causal, finite-dimensional system with zero initial state, then the estimation of its impulse response reduces to solving a finite set of overdetermined equations (section 7.2). If all data are inaccurate, TLS can be used. Moreover, **if the input-output data are perturbed by stationary zero mean white noise, TLS computes a strongly consistent estimate of** the exact impulse response, corresponding to the unperturbed system. The advantages of TLS w.r.t. LS are illustrated with a simulation example.

Next, **the use of TLS** as deconvolution technique is worked out on a practical real life problem, encountered **in renography** (section 7.3). By deconvolving the activity-time curves obtained from the blood and kidney, the renal retention function is calculated, yielding useful clinical data. Results from simulation, as well as from clinical data, show the advantages of calculating the renal retention function with TLS over the current deconvolution techniques based on back substitution or Gauss elimination using the smoothed curves. It is concluded that TLS needs no smoothing and is a powerful, efficient, reliable and stable deconvolution technique.

CHAPTER 8

USE OF TLS IN PARAMETER ESTIMATION OF TRANSFER FUNCTION MODELS

8.1. INTRODUCTION

As announced in the previous chapter, we now consider the estimation of the parameters in the linear model (7.1) and (7.2). These models are called **transfer function** (TF) models. This is seen more clearly by writing (7.1) in polynomial form :

$$A(q^{-1})\ y(t) = B(q^{-1})\ u(t) \tag{8.1}$$

u(t) and y(t) are the input and output of the system at time t respectively. $A(q^{-1})$ and $B(q^{-1})$ are polynomials of order na and nb respectively in the backward shift operator q^{-1} (i.e. $q^{-1}\ y(t) = y(t-1)$) and have the following form :

$$\begin{aligned} A(q^{-1}) &= 1 + a_1\ q^{-1} + \ldots + a_{na}\ q^{-na} \\ B(q^{-1}) &= b_1\ q^{-1} + \ldots + b_{nb}\ q^{-nb} \end{aligned} \tag{8.2}$$

Model (8.1) and (7.2) are also applicable to **multi-input multi-output** systems. For convenience only, no term b_o is included in (8.2). It is straightforward to extend the description, as well as the analysis of this chapter, to models with a b_o term. Furthermore (8.1) is also applicable to inputs with known time delay. If the delay is k (> 1) units, then a new input $\tilde{u}(t) = u(t-k+1)$ can be defined by just shifting the input vector. Also note that there is no restriction ta take $a_o = 1$ in (8.2).

If the output, and possibly the input, can not be obtained noise-free, then measurement errors must be incalculated. In this case, the TF model (8.1) can be considered in its alternative "**errors-in-variables**" or de-

composed form i.e.

$$A(q^{-1})\ y_o(t) = B(q^{-1})\ u_o(t) \tag{8.3.a}$$

$$u(t) = u_o(t) + \Delta u(t) \tag{8.3.b}$$

$$y(t) = y_o(t) + \Delta y(t) \tag{8.3.c}$$

$y_o(t)$ is the unmeasurable, noise-free output of the system, i.e. that part of y(t) which can be explained by the noise-free input $u_o(t)$ through the TF model (8.3.a); u(t) and y(t) are the noisy measured input and output respectively, while $\Delta u(t)$ and $\Delta y(t)$ are the aggregative effect of all stochastic disturbances to the input and output respectively and are assumed to be additive in form. (8.3) reveals overtly the errors-in-variables nature of the estimation problem, for which also the **instrumental variable (IV) method** was originally conceived in case the inputs are noise-free i.e. $\Delta u(t) = 0$.

It is **the aim** of this chapter **to evaluate the TLS approach and the IV techniques w.r.t. parameter estimation in the TF model** (8.3) which includes most time series models. The instrumental variable (IV) techniques constitute a class of popular methods for system identification. These methods are easy to apply and give parameter estimates with good asymptotic properties. They can be seen as generalizations of the ordinary LS method, see e.g. [139]. The IV method was apparently introduced by Reiersöl [87] and has been popular in the statistical field for quite a long period. It has been applied and adapted for dynamic systems both in econometrics and in control engineering. A well written historical background to the use of IV methods in econometrics and control engineering can be found in Young [137], who also discusses some refinements of the basic IV method. For a recent and detailed appraisal of IV methods, see [102-103-112-113]. In particular, proofs of the results stated in section 8.2, as well as many additional references on IV methods can be found in these publications.

In section 8.2 **the IV methods** are introduced. Their extensions, refinements and possibilities **are briefly described**. Section 8.3 then outlines the **relationships between the TLS and IV approach** and clarifies their equivalence under certain conditions. The noise-free, as well as the

noisy input case, is considered. In section 8.4 **simulation results** are presented comparing the finite sample accuracy properties of TLS and several IV techniques. They show significant discrepancies between the finite sample behaviour and that predicted by the asymptotic theory, for short sample lengths encountered in econometric and engineering applications.

Although most of the results are applicable to multi-input multi-output systems, we will only consider here single-output single-input processes for conciseness and clarity of exposition. The extension to the multivariable case can be done straightforwardly by replacing in (8.2) or (7.2) the scalar inputs and outputs by vectors and the parameter and data vector by matrices.

8.2. (EXTENDED) INSTRUMENTAL VARIABLE METHODS

8.2.1. NOISE-FREE INPUT CASE

Throughout this subsection, we assume that the inputs are noise-free, as required for most IV approaches. The extension of IV to the noisy input case is handled in section 8.2.2. Under this assumption, the TF model (8.3) can be written in the form :

$$A(q^{-1})\, y(t) = B(q^{-1})\, u(t) + v(t) \qquad (8.5)$$

with $v(t) = A(q^{-1})\, \Delta y(t)$. **If nb = 0, (8.5) reduces to the well-known autoregressive model** used with time series, e.g. in spectral estimation [37]. Using the notations of (7.2) and assuming m data vectors, (8.5) can be written in matrix form :

$$Y = \Phi\theta + V \qquad (8.6)$$

with $Y = [y(1), \ldots, y(m)]^T$, $V = [v(1), \ldots, v(m)]^T$ and $\Phi = [\varphi(1), \ldots, \varphi(m)]^T$ a matrix of dimension m by $n\theta$ with $n\theta = na + nb$.
When the well-known **least squares method** is used for estimating θ in the model (8.6), the result is in general **inconsistent** [103]. This drawback of the LS method can be seen as a motivation for introducing IV methods.

There are basically two related approaches to this bias problem. **The ordinary IV method** applies directly to model (8.5) in which v(t) is arbitrary noise and is not defined directly in statistical terms. **The refined or optimal IV method** applies to the full stochastic TF model (8.5), where in contrast, v(t) is assumed to be modelled by a stochastic autoregressive-moving average (ARMA) process [139]. As the primary objective of this chapter is the comparison of TLS and IV approaches, we will only consider the ordinary IV approach such that no simultaneous noise model estimation is required. The estimation of noise dynamics as required for the computation of the refined IV, is sometimes troublesome and should be avoided when possible. In fact, one of the attractive features of the original IV concept was its ability to provide consistent estimates of system parameters without estimating noise parameters.

The idea is then to **premultiply the data matrix** $[\Phi;Y]$ in (8.6) by a suitable nz by m matrix $Z^T = [z(1), \ldots, z(m)]$ whose elements z(t) are **as correlated as possible** with the data and **uncorrelated with the disturbances** v(t). This yields :

$$Z^T \Phi\theta = Z^T Y \qquad (8.10)$$

If $nz = n\theta$, then (8.10) is square and we get the so-called **ordinary IV estimator** θ_{IV} of θ :

$$\theta_{IV} = (Z^T \Phi)^{-1} Z^T Y \qquad (8.11)$$

where the inverse is assumed to exist. The elements of the vector z(t) are usually called the instruments or **instrumental variables.**
To obtain a strongly consistent estimate, one requires that :

$$R_\infty = \lim_{m \to \infty} \frac{1}{m} Z^T \Phi \quad \text{exists and is nonsingular} \quad \text{w.p. 1} \qquad (8.13)$$

and

$$\lim_{m \to \infty} \frac{1}{m} Z^T V = 0 \quad \text{w.p. 1} \qquad (8.14)$$

Under fairly general assumptions the sample covariances appearing in (8.13) and (8.14) will tend w.p. 1 to the corresponding expected values as $m \to \infty$, see e.g. [103].
The accuracy of the ordinary IV estimator (8.11) **can be improved** by prefiltering the data or **by extending the IV**, i.e. enlarging the vectors

$z(t)$ to dimension $nz > n\theta$ with additional instruments. The latter extension is also used to satisfy (8.13) more easily, especially when multi-input multi-output systems are treated [103]. The **extended IV estimator** is obtained by solving an **overdetermined** set (8.10) of nz ($> n\theta$) equations in $n\theta$ unknowns in LS sense.

The **instruments** $z(t)$ are usually formed as **delayed** and **filtered** values of the **inputs and the outputs** and have the same structure as the data vector $\varphi(t)$. Typical choices of instruments are :

$$z_1(t) = [-x(t-1), \ldots, -x(t-na), u(t-1), \ldots, u(t-nb)]^T \qquad (8.18)$$

where the signal $x(t)$ is obtained by filtering the input : $C(q^{-1})\, x(t) = D(q^{-1})\, u(t)$. If $C(q^{-1}) = 1$ and $D(q^{-1}) = q^{-nb}$ the IV variant $z_1(t)$ becomes :

$$z_2(t) = [u(t-1), \ldots, u(t-na-nb)]^T \qquad (8.20)$$

By delaying the outputs the following IV variants are obtained :

$$z_3(t) = [-y(t-k-1), \ldots, -y(t-k-na), u(t-1), \ldots, u(t-nb)]^T \qquad (8.21)$$

$$z_4(t) = [-y(t-k-1), \ldots, -y(t-k-na), u(t-k-1), \ldots, u(t-k-nb)]^T \qquad (8.22)$$

The **consistency** condition (8.14) is satisfied for $z_3(t)$ and $z_4(t)$ provided the disturbances $v(t)$ in (8.5) from a moving average process $H(q^{-1})\, e(t)$ where $e(t)$ is white noise and the delay $k \geq$ the degree of polynomial H. Observe that we obtain the well-known **Yule-Walker procedure** used for the estimation of autoregressive parameters of time series [37] when $z_3(t)$ is used with $nb = 0$. A detailed consistency analysis and a discussion of the accuracy aspects of the IV variants presented above is given by Söderstrom and Stoica [102]. The IV estimates are shown to have an asymptotic Gaussian distribution. If the IV are badly correlated with the data, the covariance matrix of their estimates will be large. The input and system parameters have a decisive influence on the accuracy of the different IV approaches. **None of these variants is thus generally superior.** Moreover, if unnecessarily delayed measurements are used as instruments the accuracy can deteriorate. In general, the resulting estimates are not optimum in an asymptotic, minimum variance sense but nevertheless consistent and relatively efficient under a wide variety of circumstances [139].

Finally, we note that a main advantage of all IV methods is that **good on-line estimation schemes** exist. Recursive schemes allow to update the (extended) IV estimates in a **recursive** fashion on the basis of the estimation results, see e.g. [138].

8.2.2. NOISY INPUT CASE

Although IV methods are primarily conceived to handle noise-free inputs, they can also be used for **computing consistent solutions of (8.3) in case the inputs are noisy.** Therefore the conditions (8.13) and (8.14) can be **straightforwardly extended.** Define the noise vector

$$\delta^T(t) = [-\Delta y(t-1), \ldots, -\Delta y(t-na), \Delta u(t-1), \ldots, \Delta u(t-nb)] \qquad (8.24)$$

then we have

$$\Phi = \Phi_o + \Delta \qquad (8.25)$$

where matrix Φ is defined in (8.6) and

$$\Phi_o = [\varphi_o(1), \ldots, \varphi_o(m)]^T$$

$$\text{with } \varphi_o(t) = [-y_o(t-1), \ldots, -y_o(t-na), u_o(t-1), \ldots, u_o(t-nb)] \qquad (8.26)$$

$$\Delta = [\delta(1), \ldots, \delta(m)]^T \quad \text{with} \quad \delta(t) \text{ given by (8.24)} \qquad (8.27)$$

Then a sequence of m by nz matrices Z ($m = n\theta, \ldots, \infty$) and $nz \geq n\theta$, is an IV matrix sequence [13] if

$$\lim_{m \to \infty} \frac{1}{m} Z^T \Phi_o \quad \text{nonsingular} \qquad \text{w.p. 1} \qquad (8.28)$$

$$\lim_{m \to \infty} \frac{1}{m} Z^T \Delta = 0 \qquad \text{w.p. 1} \qquad (8.29)$$

According to IV methods, the parameter estimate θ_{IV} is obtained by solving (8.10).

Instruments $z(t)$ satisfying (8.28) and (8.29) can be obtained using the

IV-4 variant (8.22) with sufficiently large time delay k. Indeed, (8.3) yields :

$$A(q^{-1})\ y(t) = B(q^{-1})\ u(t) + A(q^{-1})\ \Delta y(t) - B(q^{-1})\ \Delta u(t) \qquad (8.30)$$

The consistency condition (8.14) (or equivalently (8.29)) is satisfied for all experimental conditions provided the disturbances

$$v(t) = A(q^{-1})\ \Delta y(t) - B(q^{-1})\ \Delta u(t) = H(q^{-1})\ e(t) \qquad (8.31)$$

form a moving average process with e(t) white noise and $k \geq nh$, the degree of the polynomial H. If the input-output sequences are obscured by additive white noise, v(t) in (8.31) can be modelled as a moving average process with nh = max (na,nb).

Of course, this IV-variant is constructed with **delayed measurements**. Hence, the instruments might be less correlated with the vector $\varphi(t)$. This delay affects the accuracy of the computed estimate θ_{IV}, causing a large covariance matrix of θ_{IV} [102]. Extended IV are needed in order to satisfy (8.14) more easily and improve the accuracy, as confirmed by our experiments (see section 8.4).

8.3. RELATIONSHIPS BETWEEN TLS AND IV METHODS

We now explore the relationships between TLS and the (extended) IV method w.r.t. parameter estimation in the TF model (8.3). First of all, **both methods give strongly consistent estimates** in a statistical framework **if certain conditions are satisfied**. Indeed, consider the data matrix Φ in (8.6) written as :

$$\Phi = [\Phi_y ; \Phi_u] \qquad (8.32)$$

with Φ_u (resp., Φ_y) containing the last nb (resp., first na) columns of Φ, related to the input (resp., the output). Then the TLS estimate θ_{TLS} of (8.3) is the minimum norm solution of :

$$\hat{\Phi}\theta = [\hat{\Phi}_y ; \hat{\Phi}_u]\ \theta = \hat{Y} \qquad (8.33)$$

with $\hat{\Phi} = [\hat{\Phi}_y ; \hat{\Phi}_u]$ and $\hat{Y}$ the TLS approximation of Φ and Y respectively. If the input is noise-free, i.e. $\Delta u(t) = 0$, then the last nb columns

Φ_u of Φ are exactly known and thus, θ_{TLS} must be computed using algorithm 1-2 (see section 1.8.2). Hence, $\Phi_u = \hat{\Phi}_u$.

From (4.16), it follows that **TLS computes a strongly consistent estimate** θ_{TLS} of the true parameters of the undisturbed TF model (8.3.a), if the errors in $[Y ; \Phi]$ of (8.6) are rowwise independently and identically distributed with zero mean and common covariance matrix σ_v^2 I. As $[Y;\Phi]$ is Toeplitz-like, this **requires that Δu(t) and Δy(t) in (8.3) are stationary, zero mean white noise sequences**, according to [26]. The TLS solution is still consistent if the input is noise-free. Indeed, using algorithm 1-2 the initial QR factorization of Φ_u computes two mutual orthogonal subspaces $R(Q_1)$ and $R(Q_2)$ such that the problem is divided into two independent problems which are solved separately. From (1.124) and (8.32), we have :

$$[\Phi_u ; \Phi_y ; Y] = Q_1\ [R_{11} ; R_{12} ; R_{1b}] + Q_2\ [0 ; R_{22} ; R_{2b}] \qquad (8.34)$$

with $[Q_1 ; Q_2]$ orthogonal and Q_1 (resp., Q_2) of dimension m by nb (resp., m by na + 1). $R(Q_1)$ corresponds with the range $R(\Phi_u)$ of the known columns Φ_u, while $R(Q_2)$ is the range of the orthogonal projection of $R([\Phi_y ; Y])$ of the noisy columns Φ_y and Y onto the orthogonal complement of $R(\Phi_u)$. Multiplying (8.34) with $[\theta_b^T ; \theta_a^T ; -1]^T$, $\theta_a = [a_1, \ldots, a_{na}]^T$, $\theta_b = [b_1, \ldots, b_{nb}]^T$, we obtain two independent sets of linear equations :

$$Q_2\ R_{22}\ \theta_b = Q_2\ R_{2b} \qquad (8.35)$$

$$Q_1\ R_{11}\ \theta_a = Q_1\ R_{1b} - Q_1\ R_{12}\ \theta_b \qquad (8.36)$$

By computing a consistent solution $\hat{\theta}_a$ and $\hat{\theta}_b$ in each space $R(Q_1)$ and $R(Q_2)$ respectively, the total solution $\theta_{TLS} = [\hat{\theta}_a^T ; \hat{\theta}_b^T]^T$ remains consistent. This is done as follows. Solving (8.35) with TLS yields a consistent estimate $\hat{\theta}_b$ if the outputs y(t) are obscured by stationary, zero mean white noise. Indeed, the noise covariance matrix remains invariant w.r.t. the orthogonal transformation Q_2. Since the data matrix $Q_1\ R_{11}$ in (8.36) is noise-free, it is well-known that a consistent estimate $\hat{\theta}_a$ is obtained by solving (8.36) with LS. And thus, $\theta_{TLS} = [\theta_a^T ; \theta_b^T]^T$ solved with algorithm 1-2, remains consistent. The proof can be straightforwardly extended for multi-input multi-output systems.

The IV methods compute strongly consistent estimates if the consistency conditions (8.13) and (8.14) (or (8.28) and (8.29) in case of noisy inputs) are satisfied. Hence, **TLS is more restrictive in use than IV** which allows arbitrary noise models.

Comparing the TLS estimate (8.33) with the IV estimate (8.10), we can conclude that **both estimates coincide if the IV matrix Z equals the TLS approximation** $\hat{\Phi}$. Indeed, using the properties of the TLS approximation (see chapter 2) (8.33) can be written in the following form :

$$\hat{\Phi}^T \Phi\theta = \hat{\Phi}^T Y \tag{8.37}$$

The TLS approximation $\hat{\Phi}$ is a more sophisticated IV variant because the IV matrix depends not only on the parameter t but also on the parameter vector θ to be estimated.

Thus, **TLS can be considered as a variant of the IV method where the IV are function of the time instant t and the model parameters** θ. **TLS computes a strongly consistent estimate** θ_{TLS} **of the true parameters** θ_o **if the outputs y(t) and possibly the inputs u(t) are obscured by stationary zero mean white noise.** Observe that the IV need not be generated. Hence, **TLS is much simpler to use than IV.** Unfortunately, **the TLS solution involves off-line computations,** i.e. a set of observations is made and estimates of the system parameters are computed on the basis of the entire set of data. An advantage of **the IV method** is that it exists in recursive form. It thus yields **good on-line estimation** schemes, but instrumental variable generation remains a crucial problem.

The equivalence between both methods can be further elucidated by the following **geometric interpretation.** (8.37) implies that TLS makes its residual vector orthogonal onto the range $R(\hat{\Phi}) = R([\hat{\Phi};\hat{Y}])$ of the TLS approximation, i.e.

$$\hat{\Phi}^T (\Phi\theta_{TLS} - Y) = 0 \tag{8.39}$$

IV makes its residual orthogonal onto the range R(Z) of its instrumental variables, i.e.

$$Z^T (\Phi\theta_{IV} - Y) = 0 \tag{8.40}$$

Note that the LS estimate θ_{LS} makes its residual orthogonal onto the

range $R(\Phi)$ of the data or

$$\Phi^T (\Phi\theta_{LS} - Y) = 0 \tag{8.41}$$

Since the residuals are the errors between the real output y(t) and the output $\varphi^T(t)\theta$ predicted by the model, (8.40) requires the **prediction error** $y(t) - \varphi^T(t)\theta$ to be **uncorrelated with the instruments** z(t). Therefore the IV methods are called **correlation** techniques.

By **enlarging** the instrumental vector z(t), the IV matrix Z will have more columns. If Z estimates the true data space $R(\Phi_o)$, this implies that $R(\Phi_o)$ can be better estimated by extending the IV since more data vectors are involved in Z. Since Z must satisfy the consistency conditions (8.13) and (8.14), the extended IV estimate is consistently estimated by solving (8.10) with LS. No asymptotic bias is introduced. Since the asymptotic variance of the TLS estimate is larger than the asymptotic variance of the LS estimate, it makes no sense to compute (8.10) with TLS. This has been confirmed in our experiments, showing comparable accuracy of the LS and TLS estimates of (8.10) in a lot of experimental conditions.

Finally, by **scaling the data appropriately,** TLS also computes **consistent estimates** of the parameter vector θ in case the **noise** $\Delta y(t)$ and $\Delta u(t)$ on the outputs and inputs in (8.3) are **stationary, zero mean processes with known rational, nonsingular spectral density matrix.** Indeed, it is a consequence of the spectral factorization theorem [8] that the disturbances can be described as

$$\Delta u(t) = G(q^{-1})\, \varepsilon(t) \tag{8.42}$$

$$\Delta y(t) = H(q^{-1})\, e(t) \tag{8.43}$$

where $H(q^{-1})$, $G(q^{-1})$ and their inverse are asymptotically stable finite order filters and e(t), $\varepsilon(t)$ are zero mean white noise sequences. If G and H are known, model (8.3) can be transformed appropriately as follows :

$$\tilde{A}(q^{-1})\, \tilde{y}_o(t) = \tilde{B}(q^{-1})\, \tilde{u}_o(t) \tag{8.44.a}$$

$$\tilde{u}(t) = \tilde{u}_o(t) + \varepsilon(t) \tag{8.44.b}$$

$$\tilde{y}(t) = \tilde{y}_o(t) + e(t) \qquad (8.44.c)$$

$$\text{with } \tilde{u}_o(t) = G(q^{-1})^{-1}\, u_o(t) \quad \text{and} \quad \tilde{y}_o(t) = H(q^{-1})^{-1}\, y_o(t) \qquad (8.45.a)$$

$$\tilde{u}(t) = G(q^{-1})^{-1}\, u(t) \quad \text{and} \quad \tilde{y}(t) = H(q^{-1})^{-1}\, y(t) \qquad (8.45.b)$$

$$\tilde{B}(q^{-1}) = B(q^{-1})\, G(q^{-1}) \quad \text{and} \quad \tilde{A}(q^{-1}) = A(q^{-1})\, H(q^{-1}) \qquad (8.45.c)$$

Since the transformed data (8.44.b) and (8.44.c) are obscured by white noise, the coefficients of $\tilde{A}(q^{-1})$ and $\tilde{B}(q^{-1})$ can be estimated consistently by solving (8.44) with TLS. Using these estimates and the transformations (8.45.c), the coefficients of the polynomials $A(q^{-1})$ and $B(q^{-1})$ of the original model (8.3) can be computed.
Hence, TLS can always be used as an alternative consistent estimation technique of the parameters in the TF model (8.3), if the moving average processes G and H in (8.42) and (8.43) are known. Of course, if the common covariance matrix R of the noise on the data vectors $[y(t); \varphi^T(t)]$ is given, then the data given in matrix form (8.6) can also be appropriately transformed using the scaling method discussed in section 4.5.

8.4. SIMULATION RESULTS AND DISCUSSION

Many methods for estimating the parameters of a transfer function model (8.3) have been suggested in the literature. However, in the presentation of these methods little or no attempt has been made **to evaluate their performance when used with noisy data or short sample lengths**, and to compare the performance of the method presented with other existing methods. To provide this evaluation and comparison, the results of an experimental study are now described. We compare the TLS technique with several IV methods for estimating the parameters in the TF model (8.3), given only the normal operating input and output of the system over a **limited** period of time. We moreover assume that the noise on the data is **discrete and white.**

Simulation set up

We start from an exact TF model (8.3.a) in which the true model parameters θ, namely the na coefficients a_i of $A(q^{-1})$ and the nb coefficients b_i of $B(q^{-1})$, are to be estimated. The real coefficients a_i of $A(q^{-1})$ are such that the polynomial A(z) has all its zeros outside the unit circle. The input sequence $u_o(t)$ is persistently exciting of order $n\theta$ = na + nb. We **obscure** the inputs $u_o(t)$ and outputs $y_o(t)$ **by stationary, zero mean white noise** $\Delta u(t)$ and $\Delta y(t)$ respectively, to obtain u(t) in (8.3.b) and y(t) in (8.3.c). The model parameters in (8.3.a) are then estimated using m data observations $y(k_o+t)$, $\varphi^T(k_o+t)$ given by (7.2), for t = 1, ..., m with k_o an initial offset. This requires m + na - 1 output samples and m + nb - 1 input samples. **The sample lengths m are** kept **rather short** compared to most experiments described in the literature, see e.g. [58-111]. The TLS estimate is obtained by solving a finite set of m linear equations (8.6) in $n\theta$ = na + nb unknown model parameters θ, given by (7.2). In case of multi-input multi-output systems, u(t) and y(t) are vectors of dimension na and ny respectively and the coefficients b_i and a_i are matrices of dimension ny by nu and ny by ny respectively. **Three IV methods are examined.** The (extended) IV estimate is obtained by solving a squared or overdetermined set of nz equations (8.10), $nz \geq n\theta$, for different values of nz. If the inputs are noise-free, the IV-2 variant (8.20), consisting of nz delayed inputs, and the IV-3 variant (8.21), consisting of na delayed outputs and nz-na delayed inputs with $k \geq na$ (for consistency) are used to generate the IV. In case of noisy inputs, the IV are given by the IV-4 variant (8.22), having na delayed outputs and nz - na delayed inputs with $k \geq \max(na,nb)$ to maintain consistency.
The **accuracy** of the estimates are measured by computing the **squared bias, total variance and MSE** given by (4.37) and (4.38). The expected quantities are estimated by taking the average over the estimates obtained from 100 different realizations of input-output sequences obscured by noise with equal variance σ_v^2. These estimated values are then divided by $\|\theta\|_F^2$ with θ the true model parameters and plotted versus the noise variance σ_v^2 in a log-log diagram.

Several TF models taken out of literature [103-111-112] but used with different in- and outputs, are considered, see e.g. fig. 8.1 and 8.2. The following observations can be made, regarding the results.

In general, TLS outperforms the IV methods. It gives estimates with

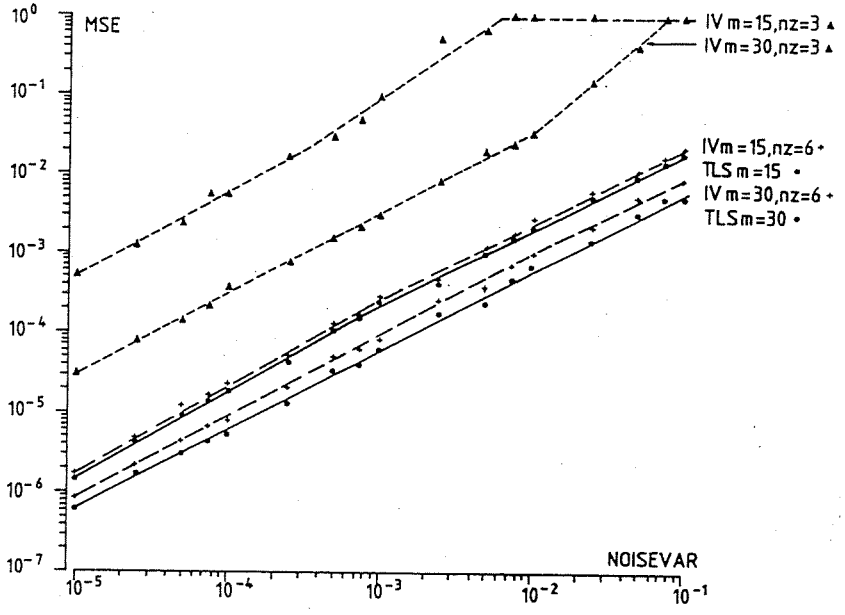


Fig. 8.1 : Comparison of the relative mean squared error (MSE) of the TLS estimator and (a) the IV-2 estimator (8.20), (b) the IV-3 estimator (8.21), (c) the IV-4 estimator (8.22) versus the variance of the noise added to the outputs (and inputs in 8.1.c) of the exact TF model :

$$y_o(t) - 1.75\, y_o(t-1) + 0.76\, y_o(t-2) = u_o(t-1)$$

The zeros of A(z) are 1.25 and 1.05. The input sequence $u_o(t)$ is white noise with zero mean and unit variance. The sample lengths m = 15 and 30 and the IV extension degrees nz = 5 and 10 are chosen (log-log diagram). Errors larger than one are set equal to one.

smaller MSE than the ordinary IV methods. The MSE becomes comparable by extending the IV sufficiently. Because of their consistency, the TLS and IV estimates have bias magnitudes which are consistently less than their

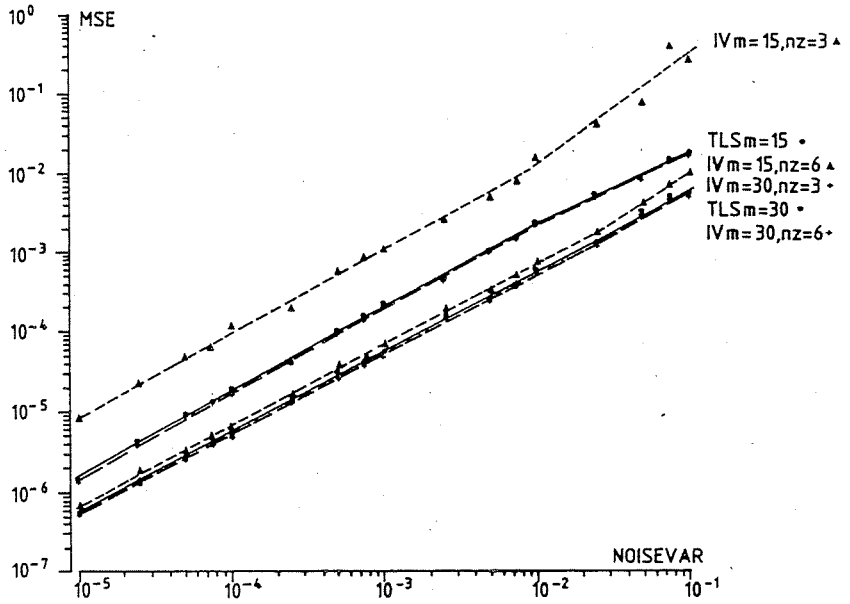


Fig. 8.1.b : The IV-3 estimator.

standard deviations. Mostly the TLS method is superior in terms of both bias and variance. The **superiority of TLS w.r.t. IV is particularly clear in case the zeros of the polynomial A(z) approach the unit circle** (fig. 8.1) **or both the inputs and the outputs are noisy** (fig. 8.1.c and fig. 8.2.c). **In other cases, the TLS method and IV methods give quite similar results** (see fig. 8.2.a-b).

The relative performance of both methods **w.r.t. their accuracy in parameter estimation is mainly influenced** by the following factors.

- First of all, **the extension degree nz of the IV is crucial.** It is difficult to give precise rules for choosing nz. As the zeros of A(z) approach the unit circle, the differences in accuracy between the ordinary and extended IV estimates become more pronounced and nz should

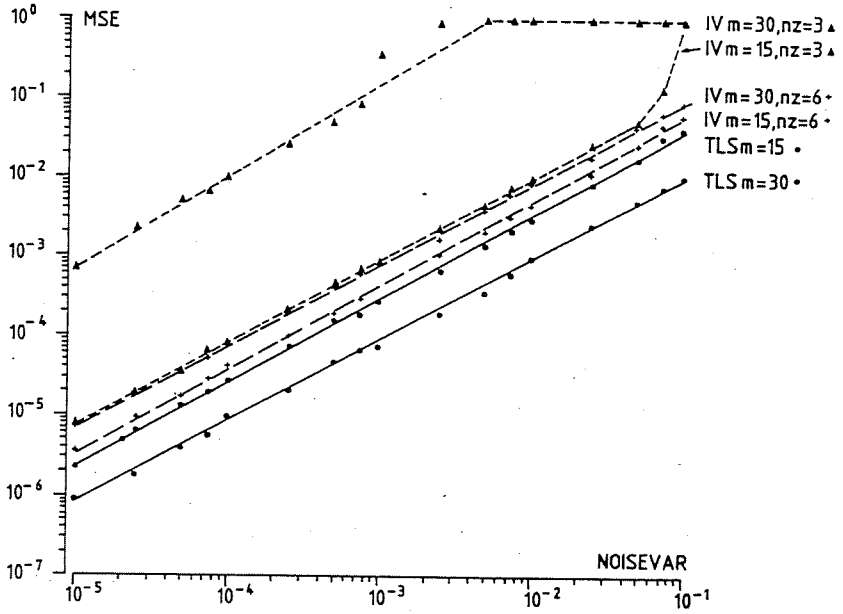


Fig. 8.1.c : The IV-4 estimator.

be enlarged to obtain the best accuracy. This was also the conclusion of a large number of empirical studies reported in the signal processing literature [36-37-111] and can be explained as follows. Since the IV mainly consist of delayed inputs and outputs, the elements of the matrices $Z^T\Phi$ in (8.10) are essentially correlation coefficients. When the zeros of A(z) are near the unit circle, there is still valuable information in the higher order correlation coefficients, as the correlation function is highly oscillatory and slowly decaying. This is the case, for example, if the process y(t) contains narrow band signals. Hence, choosing nz > nθ will improve the statistical efficiency of the estimator. For short sample lengths, it was observed that the accuracy of the IV estimates increases with growing nz until it saturates at a certain value nz_o. This value depends on the zeros of A(z). As long as $nz < nz_o$, TLS is clearly superior to IV. The differences

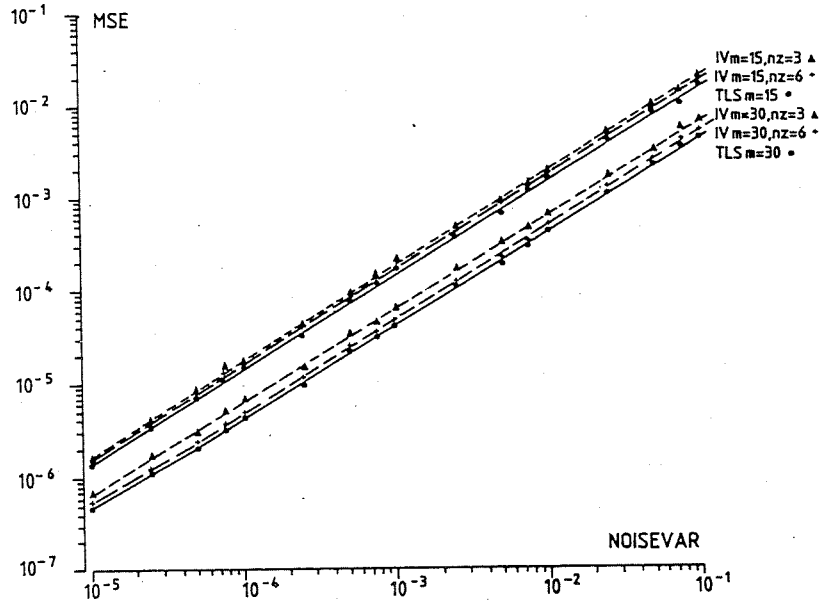


Fig. 8.2 : Comparison of the relative mean squared error (MSE) of the TLS estimator and (a) the IV-2 estimator (8.20), (b) the IV-3 estimator (8.21), (c) the IV-4 estimator (8.22) versus the variance of the noise added to the outputs (and inputs in 8.2.c) of the exact TF model :

$$y_o(t) - .3\, y_o(t-1) + .5\, y_o(t-2) = 1.5\, u_o(t-1)$$

The zeros of A(z) are $.3 \pm 1.382\, j$. The input sequence $u_o(t)$ is white noise with zero mean and unit variance. The sample lengths m = 15 and 30 and the IV extension degrees nz = 5 and 10 are chosen (log-log diagram). Errors larger than one are set equal to one.

may be quite large : the TLS estimates can be 100 times better than the ordinary IV estimates in case of noise-free inputs (fig. 8.1.a) and even 1000 times better in case of noisy inputs (fig. 8.1.c). Both

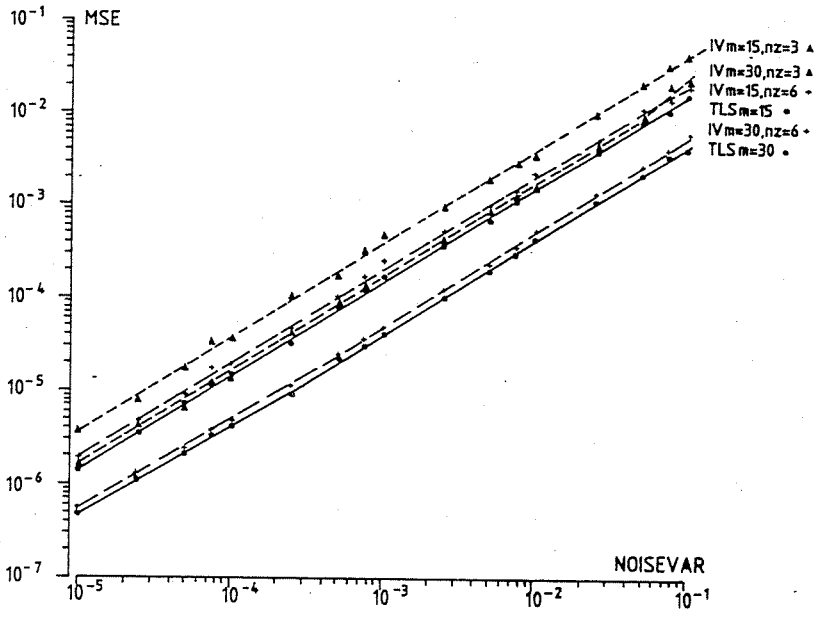


Fig. 8.2.b : The IV-3 estimator.

methods become comparable in accuracy once the bound nz_o is reached and this remains so for larger nz or larger m. If the zeros A(z) are far from the unit circle, the ordinary IV estimates (8.11) with nz = nθ, are already sufficiently accurate in case of noise-free inputs so that IV and TLS give comparable results (fig. 8.2.a).

- Second, **the sample length m** influences the relative accuracy of both methods. For TLS, asymptotic theory (for m going to infinity) holds quite well for short sample lengths. However, for IV methods considerable differences between asymptotic theory and practical finite sample behaviour occur for short sample lengths especially when the zeros of A(z) approach the unit circle. Agreement is only found at large extension degrees nz. That is why the differences in accuracy may be quite large. They can even increase with growing m, if the extension degree

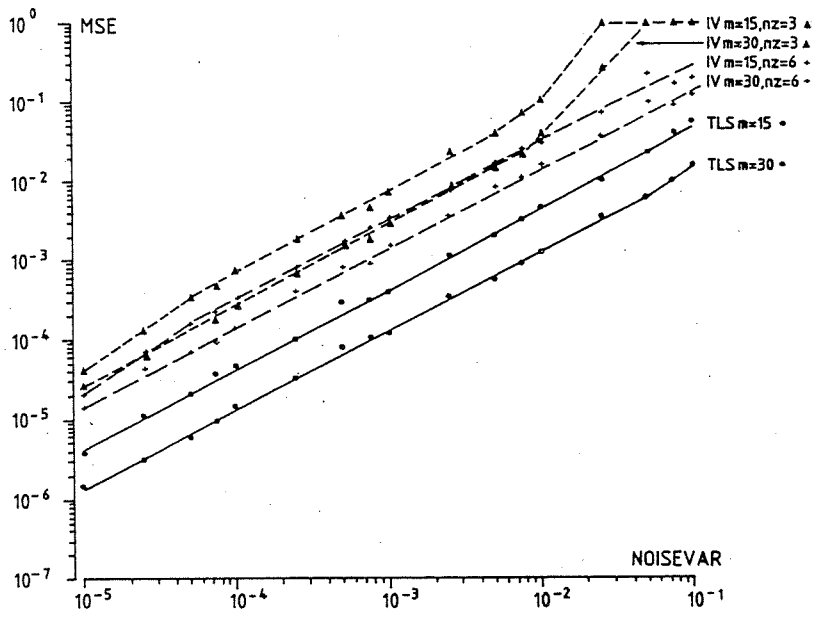


Fig. 8.2.c : The IV-4 estimator.

nz is far below the value nz_o at which the IV accuracy saturates (fig. 8.1). If nz approaches nz_o, the IV accuracy always improves with growing m but the improvement is larger for TLS. However, once the extension degree nz exceeds nz_o, both methods improve equally in accuracy and hence, the sample length m no longer influences the relative accuracy of TLS w.r.t. IV (fig. 8.2).

- Third, **the chosen IV variant determines the accuracy** of the IV estimate and thus influences the relative accuracy w.r.t. TLS. Let us first consider the **noise-free input** case for which the IV-2 and IV-3 variants are used to generate the IV. As said in section 8.2, among those examined, there is no IV variant which is systematically superior. The input and the system parameters have a crucial importance on the performance of the IV method. It was observed that IV-2 (fig.

8.2.a) is better in accuracy than IV-3 (fig. 8.2.b) when the ordinary IV estimates are used and the zeros of A(z) do not approach the unit circle. This result agrees with [102]. For larger extension degrees nz or when the zeros of A(z) approach the unit circle, IV-3 was found to be more accurate (fig. 8.1.a-b).

If **both the inputs and outputs are noisy**, IV-4 is used to generate the IV. The accuracy of the IV-4 estimates and also the IV-3 estimate is maximized if the IV are as correlated as possible with the data vector $\varphi(t)$. Therefore the delay k should be minimized in order to keep the covariance matrix of the IV estimates as small as possible, while maintaining consistency. For discrete, white noise disturbances this minimal value of k is equal to na for IV-3 in (8.21) and equal to max (na, nb) for IV-4 in (8.22). It was observed that **a larger delay deteriorates the accuracy.** Since IV-4 must be constructed from sufficiently delayed inputs and outputs to maintain consistency, the IV-4 are less correlated with the data $\varphi(t)$ than IV-3 or IV-2. This causes more ill-conditioned matrices $Z^T\Phi$ and larger covariance matrices of the IV-4 estimates. That is why the accuracy of the IV method deteriorates considerably in the presence of noisy inputs and remains quite inferior w.r.t. the TLS estimate (fig. 8.1.c and 8.2.c). **Highly extended IV** with $nz >> n\theta$, **are required** in order to satisfy (8.14) more easily and improve the accuracy of the IV estimate. Nevertheless, TLS remains more accurate. This superiority of the TLS estimate over the IV estimate is more pronounced with decreasing extension degree nz and increasing sample length m. In particular, if the zeros of A(z) approach the unit circle, the differences are quite large and TLS may exceed IV over several orders of magnitude in accuracy (fig. 8.1.c).

Finally note that we only compared TLS with **some** IV methods w.r.t. their accuracy in the presence of **white noise** disturbances on the data and **short sample lengths**. Very often, IV is used with large sample lengths $m > 1000$. In these cases, not only the differences in accuracy w.r.t. TLS are less pronounced but mainly the **computational efficiency** becomes now an important factor. Because of their ability to yield good on-line estimation schemes, IV methods outperform TLS involving off-line computations.

Summarizing, we may conclude that **TLS is especially useful** in the estimation of the parameters of a TF model **for short sample lengths**, when the outputs and possibly the inputs are obscured by **white noise.** TLS is particularly recommended **when the zeros of the polynomial A(z) approach the unit circle or when both the inputs and outputs are disturbed.**

8.5. CONCLUDING REMARKS

This chapter evaluates the TLS approach and the IV techniques w.r.t. parameter estimation in the transfer function (TF) model. **Both methods** are able **to compute strongly consistent estimates.** In the **Instrumental Variable (IV) method**, the estimate is obtained by first premultiplying the matrix of observed data by a suitable rectangular matrix whose elements, called instrumental variables, are as correlated as possible with the observations and uncorrelated with the disturbances. This yields a square, invertible matrix which is then solved. **By taking additional IV** or by prefiltering the data, **the accuracy** of the estimate **can be improved.** This yields an overdetermined set of linear equations from which the extended IV estimate is obtained. An advantage of the IV methods is that it exists in recursive form allowing **good on-line estimation** schemes, but **IV generation remains** a **crucial** problem. Usually, the IV are formed as delayed and filtered values of the inputs and outputs. IV is essentially conceived for noise-free inputs. However, IV can also be used to compute consistent estimates when one has to deal with noisy inputs.

When the outputs and possibly the inputs are obscured by discrete stationary, zero mean white noise, TLS computes a **strongly consistent** estimate of the true model parameters θ_o. TLS can be considered as a **variant of the IV method** where the IV matrix given by the TLS approximation of the data, is function of the time instant t and the model parameters. Contrary to the IV methods, the IV need to be generated. Hence TLS is **much simpler to use but more restrictive** (IV allows arbitrary noise models). Unfortunately the TLS solution involves **off-line computations** and is thus computationally not so attractive when large sample lengths are used.

Finally, the accuracy of both methods w.r.t. parameter estimation in a TF model is compared by simulations. It is concluded that **TLS outperforms the IV methods** when **short sample lengths** are used and when the outputs and possibly the inputs are obscured by stationary, zero mean **white noise**. Mostly, the TLS method is superior in terms of both bias and variance and gives estimates with smaller MSE than the ordinary IV methods. The MSE becomes comparable by extending the IV sufficiently. The **superiority** of TLS is **particularly clear in case the zeros of the polynomial A(z) approach the unit circle or both the inputs and outputs are noisy**. In **other cases**, TLS and IV methods give **quite similar results**.

CHAPTER 9

DIRECT SPEED IMPROVEMENT OF THE TLS COMPUTATIONS

9.1. INTRODUCTION

In the previous chapters, we mainly studied the **accuracy** of the TLS technique in the presence of perturbations on all data or multicollinearities. A thorough comparison with many other currently used estimation techniques such as LS, the IV methods, ... enlightens the superiority of TLS in solution accuracy. Its practical use is moreover proven in diverse applications, e.g. subset selection (chapter 6), impulse response estimation by discrete deconvolution (chapter 7) and parameter estimation in a transfer function model (chapter 8). This analysis of the TLS problem must convince the reader of the importance of TLS for his own applications.

Nevertheless, the reader can still have objections. Indeed, until now our study is **mainly concentrated on the accuracy** of the TLS problem. Although TLS can improve the accuracy, its **computational cost** can be **too high** in applications where the computational speed is an important factor, e.g. in real-time estimation. That is the main reason why less reliable methods are still used. Therefore, we now try to **improve the efficiency of the TLS computations** as presented in the classical TLS algorithm 1-1 (section 1.8.1). Since the TLS solution of (0.1) is deduced from **only one** right singular vector or in general, **a basis of the right singular subspace** associated with the smallest singular values of the data [A;B], a considerable **saving in computation** time is possible by only calculating those desired base vectors. This can be done in a **direct way** by modifying the SVD algorithm, as described in this chapter, or in an **iterative fashion** by making use of start vectors, as discussed in the following chapter.

As only a basis of the desired singular subspace is needed, the classi-

cal Singular Value Decomposition (SVD) algorithm can be modified in three ways. First, the Householder transformations of the bidiagonalization need only to be applied on the base vectors of the desired singular subspace. Second, the bidiagonal must only be partially diagonalized and third, the convergence rate of the iterative diagonalization can be improved by an appropriate choice between QR and QL iteration steps. **This improved algorithm,** called "**partial SVD** (PSVD)", **reduces the computation time considerably while the reliability of the classical SVD algorithm is maintained.** It calculates in an efficient and reliable way a basis for the left and/or right singular subspace of a matrix associated with its smallest singular values. The dimension of the desired subspace may be fixed or depend on a given parameter θ [129]. It can be successfully used for the TLS computations [2] but also for solving homogeneous equations AX = 0. Indeed, only a basis of the right singular subspace of A associated with its smallest singular values is required to compute the solution X. PSVD also applies to the calculation of the null space of a perturbed matrix whose rank is defined by the number of its singular values larger than a certain perturbation level θ. Even the solution of nonhomogeneous linear equations AX = B only requires the computation of the null space of the corresponding matrix [A;B].

This chapter is organized into six sections. In section 9.2, the classical SVD algorithm is briefly described. Section 9.3 shows the modifications in the classical SVD algorithm which improve the computational efficiency in the cases mentioned above. In section 9.4 both algorithms are compared. An analysis of the operation counts, as well as computational results, show the relative efficiency of PSVD with respect to the classical SVD algorithm. Based on this new algorithm PSVD, the **TLS computations are speeded up** and summarized **into an improved "partial TLS" algorithm** PTLS which is outlined and demonstrated in section 9.5. Finally, section 9.6 gives the conclusions.

9.2. THE CLASSICAL SVD ALGORITHM

Let A be an m by n matrix, with $m \geq n$. The Singular Value Decomposition (SVD) of A is defined by (for clarity of exposition we omit the primes used in (0.2)) :

$$A_{m\times n} = [U_1;U_2] \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} V^T \qquad (9.1)$$

with

$U_1 = [u_1, \ldots, u_n]$ the left singular vectors, $U_2 = [u_{n+1}, \ldots, u_m]$,

$u_i \in \mathbb{R}^m$, $U = [U_1;U_2]$ and $U^T U = I_m$,

$V = [v_1, \ldots, v_n]$, $v_i \in \mathbb{R}^n$, $V^T V = I_n$ the right singular vectors and

$\Sigma_{n\times n} = \mathrm{diag}\,(\sigma_1, \ldots, \sigma_n)$, $\sigma_1 \geq \ldots \geq \sigma_n \geq 0$ the singular values.

If $\sigma_n \neq 0$, the columns of U_1 (resp., V) constitute an orthonormal basis for the range (resp., row space) of A. The columns of U_2 span the complement of the range of A.
The **rank** of A is immediately deduced from the number of singular values strictly greater than zero, i.e. rank (A) = r if $\sigma_{r+1} = \ldots = \sigma_n = 0$.
There is no loss of generality in assuming that $m \geq n$. Indeed, if $m < n$ then we can instead compute the SVD of A^T.

The **classical SVD** algorithm **consists of three phases.** In the bidiagonalization phase, **matrix A is made bidiagonal.** One constructs two finite sequences of Householder transformations [48, section 3.3] :

$P^{(k)}$, $k = 1, 2, \ldots, p$ and $p = \min(m-1, n)$

and

$Q^{(k)}$, $k = 1, 2, \ldots, q$ and $q = \min(n-2, m)$

such that

$$J^{(0)} = P^{(p)} \ldots P^{(1)} A Q^{(1)} \ldots Q^{(q)} = \begin{bmatrix} q_1 & e_2 & & & 0 \\ & q_2 & e_3 & & 0 \\ & & \ddots & \ddots & \\ & & & q_n & \\ & & & & e_{n+1} \\ \hline & & 0 & & \end{bmatrix} \begin{matrix} \\ \\ \\ \\ n \\ \hline m-n \end{matrix} \qquad (9.2)$$

is an upper bidiagonal matrix. Specially, $P^{(i)}$ zeroes out the subdiagonal elements in column i and $Q^{(j)}$ zeroes out the appropriate elements in row j. Because all the transformations introduced are orthogonal, the singular values of $J^{(0)}$ are the same as those of A. Thus, if

$$J^{(0)} = G \Sigma H^T \qquad (9.3)$$

is the SVD of $J^{(0)}$ then

$$A = PG \Sigma H^T Q^T \qquad (9.4)$$

so that

$$U = PG \ , \quad V = QH \qquad (9.5)$$

with $P = P^{(1)} \ldots P^{(p)}$ (9.6)

and $Q = Q^{(1)} \ldots Q^{(q)}$ (9.7)

In the process, the matrix A can be destroyed and the $P^{(i)}$ (resp., $Q^{(i)}$) are stored in the lower (resp., upper triangular) part of A.

After the bidiagonalization, **the orthogonal matrices** P in (9.6) and/or Q in (9.7) **must be computed explicitly** during the initialization phase if U and/or V in (9.1) are required. This is done by accumulating the Householder transformations $P^{(i)}$ in (9.6) (resp., $Q^{(i)}$ in (9.7)) in the space provided for U (resp., V).

After this initialization phase, **the bidiagonal $J^{(0)}$ can be diagonalized iteratively** by the QR method [48, section 8.3] during the diagonalization phase. This diagonalization implies the construction of a sequence

$$J^{(0)} \longrightarrow J^{(1)} \longrightarrow \ldots \longrightarrow \Sigma$$

$$\text{where } J^{(i+1)} = (S^{(i)})^T J^{(i)} T^{(i)} \tag{9.8}$$

$S^{(i)}$ and $T^{(i)}$ are products of Givens transformations [48, section 3.4] and are hence orthogonal. The matrices $T^{(i)}$ are chosen so that the sequence $J^{(i)T} J^{(i)}$ converges to a diagonal matrix while the matrices $S^{(i)}$ are chosen so that all $J^{(i)}$ have a bidiagonal form. The products of the $T^{(i)}$ and the $S^{(i)}$ are exactly the matrices H and G^T, respectively, in (9.5). It has been reported in [45] that the average number of iterations on $J^{(i)}$ in (9.8) is approximately 2n i.e. $J^{(2n)}$ is usually a good approximation to a diagonal matrix. In order to accumulate the transformations and obtain the final matrices U and/or V, the $S^{(i)}$ are applied to P from the right and $(T^{(i)})^T$ are applied to Q^T from the left, for each i.

If m is larger than 5n/3, the computations are made more efficient by **first transforming the matrix A into upper triangular form**, by Householder transformations on the left [19]:

$$L^{(n)} \ldots L^{(1)} A = L^T A = \begin{bmatrix} \text{▽} \\ 0 \end{bmatrix} = \begin{bmatrix} R \\ --- \\ 0 \end{bmatrix} \tag{9.9}$$

R is n×n upper triangular and L is orthogonal. We then proceed to bidiagonalize R. The important difference here is that the matrix R is much smaller than A (if $n^2 << m.n$). Hence, the work required to bidiagonalize R is much less than that originally done by the right transformations when $m >> n$. It can be easily verified that if the SVD of R is equal to $X\Sigma Y^T$, then the SVD of A is given by

$$A = L \begin{bmatrix} X \\ --- \\ 0 \end{bmatrix} \Sigma Y^T \tag{9.10}$$

We can identify U with $L[X^T;0]^T$ and V with Y. Observe that one requires an extra product $L[X^T;0]^T$. This can be obtained from

$$L \begin{bmatrix} X \\ --- \\ 0 \end{bmatrix} = L \begin{bmatrix} I \\ --- \\ 0 \end{bmatrix} X \tag{9.11}$$

by first accumulating $L[I;0]^T$ in the space provided for U, and then multiplying by X. If U is not needed explicitly, e.g. in TLS problems, then we do not have to accumulate any left transformations. In that case, for $m > n$ we will make a substantial saving. This algorithm which triangularizes A whenever $m > 5n/3$, is called the **hybrid SVD** algorithm in [19].

9.3. THE PARTIAL SVD ALGORITHM PSVD

The original motivation is to improve the speed of the classical SVD algorithm when one requires only a basis of the singular subspace of a matrix associated with its smallest singular values. In that case, three improvements are possible.

9.3.1. DELAY INITIALIZATION PHASE

During the second phase of the classical SVD algorithm, the matrices U and V are initialized by forming the products of applied Householder transformations (9.6) and (9.7) explicitly. Now, since only a few vectors in U and/or V are requested, we can save a great amount of work by **delaying the initialization of U and V until the end of the diagonalization** phase. If we then apply the Householder transformations $P^{(i)}$ (resp., $Q^{(i)}$) only on those vectors in G (resp., H) in (9.5) which are associated with the desired smallest singular values of the bidiagonal $J^{(0)}$, we obtain the desired base vectors in U (resp., V) with a reduced computational effort.

9.3.2. PARTIAL DIAGONALIZATION

During the diagonalization phase, the singular values and associated vectors are computed one after the other. If convergence has occurred to all desired singular values, then the associated singular vectors re-

main unchanged during the following QR iterations when the QR method is converging to the remaining singular values. Hence, we can stop the QR iterations immediately after all desired singular values have been computed. In case of one singular vector, this means that the QR method only needs to converge to the associated singular value. No further QR iterations are needed for the computation of the required singular vector. Thus, the **iteration process may be stopped as soon as convergence has occurred to all desired singular values.** Moreover, if only a basis for a singular subspace is required, we need not zero out all associated superdiagonal elements of the bidiagonal $J^{(i)}$. We only have to partition the bidiagonal into unreduced subbidiagonals, such that all singular values of each subbidiagonal are either all greater than a given bound θ or all smaller than or equal to θ. The required basis then consists of all vectors associated with those subbidiagonals, whose singular values are all smaller than or equal to the bound θ.

To **partition a bidiagonal** of the form (9.2) **into unreduced subbidiagonals**, we must search the bidiagonal band for zeros. If $e_i = 0$ for some i, then the bidiagonal is immediately decoupled. If a diagonal element $q_i = 0$ for some i, then premultiplication by a sequence of Givens rotations $S^{(i)}$ can zero e_{i+1} and postmultiplication by a sequence of Givens rotations $T^{(i)}$ zeroes out e_i. This cancelling operation decouples the bidiagonal by isolating a 1 by 1 unreduced submatrix with zero singular value. Hence, its corresponding base vector belongs to the desired singular subspace. Thus, we can partition the bidiagonal whenever an e_i, $i = 2, \ldots, n$ or a q_j, $j = 1, \ldots, n$ is zero. See also [29, section 11] and [2]. By choosing an appropriate shift for the QR iterations, the partial diagonalization proceeds as follows.
One constructs a sequence of bidiagonals $J^{(i)}$, similar to (9.8) in the classical SVD algorithm. After each QR iteration step with appropriate shift (i-th step), one obtains :

$$J^{(i)} = \begin{bmatrix} J_1^{(i)} & & & 0 \\ & J_2^{(i)} & & \\ & & \ddots & \\ 0 & & & J_k^{(i)} \end{bmatrix} \qquad (9.12)$$

containing k unreduced subbidiagonals $J_j^{(i)} = \begin{bmatrix} x & x & & 0 \\ & x & \ddots & \\ & & \ddots & x \\ 0 & & & x \end{bmatrix}$, $j=1, \ldots, k$

The subbidiagonals $J_j^{(i)}$ are partitioned into three classes

$C_1 = \{J_j \mid$ all singular values of J_j are $> \theta\}$

$C_2 = \{J_j \mid$ all singular values of J_j are $\leq \theta\}$

$C_3 = \{J_j \mid J_j$ has at least one singular value $\leq \theta$ and at least one singular value $> \theta\}$

If C_3 is empty then stop,
else apply one QR iteration step onto each element of C_3 and classify again.

In order to obtain a basis for the singular subspace of a matrix associated with its smallest singular values, **the shift** is set equal to the **smallest diagonal element of the** considered **subbidiagonal.** This shift must be smaller than the given bound θ, otherwise it is set equal to zero. With this choice the iteration process is forced to converge first to the smallest singular values of the matrix and its associated vectors. This shift was experimentally found to give the best convergence results.

If no bound on the singular values can be given but instead, **the dimension d of the desired singular subspace is known,** then **one can compute a bound** θ with the **bisection method** [48, section 8.5] such that the ma-

trix has exactly d singular values smaller than or equal to θ. This is done at the end of the bidiagonalization phase with the following algorithm :

Algorithm 9-1 : **estimation of the bound θ**

Given : - an n by n bidiagonal J, given by (9.2).
- the dimension d of the desired singular subspace of J.
- an initial estimate of θ ($\theta \leq \delta$ if no initial estimate is available).
- a stop criterion δ, defining the minimum subinterval length in the bisection method.

Step 1 : **initialization of θ**

if $\theta \leq \delta$ then if d = 1 then $\theta \longleftarrow \min_{i=1,n} |q_i|$

else $\theta \longleftarrow |q_{n-d+1}|$

Step 2 : **check quality initial estimate of θ**

num $\longleftarrow \#\{\sigma_i(J) \mid \sigma_i \leq \theta\}$

if num = d return

Step 3 : **initialization starting values for bisection method**

if num < d then $y \longleftarrow \theta$; $z \longleftarrow \sqrt{\max_i (\alpha_i + |\beta_i| + |\beta_{i+1}|)}$

else $z \longleftarrow \theta$; $y \longleftarrow \sqrt{\max(\delta, \min_i (\alpha_i - |\beta_i| - \beta_{i+1}|))}$

with α_i, β_i given by (9.15)

Step 4 : **bisection method**

```
Do while num ≠ d
        θ ⟵ (y + z) / 2
        num ⟵ #{σ_i(J) | σ_i ≤ θ}
        if num < d  then y ⟵ θ
                    else z ⟵ θ
        if z - y < δ  then begin
                           θ ⟵ z
                           d ⟵ #{σ_i(J) | σ_i ≤ θ}
                           return
                           end
        end
END
```


Some comments are in order :

- In step 1, using the properties of triangular matrices [67] we know that min $|q_i|$ is always an upper bound for the minimal singular value of J and approaches it very closely, when the gap is large, i.e. $\sigma_{n-1}(J) >> \sigma_n(J)$. The initial estimate of θ in case d > 1 was experimentally found to be suitable if the gap between $\sigma_{n-d}(J)$ and $\sigma_{n-d+1}(J)$ is large (this is the case for most TLS problems).

- In step 2 and 4, the number of singular values which are ≤ θ is the sum of the number of singular values ≤ θ over all unreduced bidiagonals J_j, given in (9.12). The computation of the number of singular values ≤ θ of each unreduced bidiagonal J_j is based on the **Sturm sequence** property of the unreduced tridiagonal matrices $T_j = J_j^T J_j$ (see below).

- In step 3, the starting values for the bisection method are derived from the **Gerschgorin circle** theorem [48, theorem 7.2-1].

- In step 4, the bisection method produces a sequence of subintervals [y,z] that are repeatedly halved in length but such that

$$\#\{\sigma_i(J) \mid \sigma_i \leq y\} < d < \#\{\sigma_i(J) \mid \sigma_i \leq z\}$$

We have to find a bound θ such that $\sigma_{n-d}(J) > \theta \geq \sigma_{n-d+1}(J)$. Hence, if at a certain moment z - y < δ then at least two singular values of J lie in the interval [y,z] within a distance < δ from each other. If δ is defined by the standard deviation of the noise on the data [A;B] (like the bound (1.117)), this means that σ_{n-d+1} may be considered as a singular value of multiplicity > 1. Hence, it makes no sense to continue the bisection further, trying to separate the singular values σ_{n-d} and σ_{n-d+1} of J. Instead, we consider the singular values of J in the interval [y,z] as coinciding with σ_{n-d+1} and **determine its multiplicity** by the number of singular values of J in the interval [y,z]. We therefore better enlarge the dimension of the desired singular subspace such that it contains all base vectors of the ρ-dimensional singular subspace associated with $\sigma_{n+d-1}(J)$ of multiplicity ρ. Observe however that this method only allows to compute the multiplicity of a singular value rather arbitrarily. Of course, a warning is given to the user.

The classification of the subbidiagonals is based on the Sturm sequence property of the unreduced tridiagonal matrices $T_j^{(i)} = J_j^{(i)T} J_j^{(i)}$ [48, section 8.5]. One can easily compute the number of singular values of $T_j^{(i)}$ smaller than or equal to the bound θ, by counting the number of sign changes in the sequence

$$p_0(\theta),\ p_1(\theta),\ \ldots,\ p_s(\theta) \tag{9.13}$$

where $p_0(\theta) = 1$ and $p_r(\theta) = \det(T_j^{(i)} - \theta.I)$ for $r = 1, \ldots, s$ with s the dimension of $T_j^{(i)}$. (Convention : $p_r(\theta)$ has the opposite sign from $p_{r-1}(\theta)$ if $p_r(\theta) = 0$). A simple determinantal expansion can be used to show that

$$p_r(\theta) = (\alpha_r - \theta)\, p_{r-1}(\theta) - \beta_r^2\, p_{r-2}(\theta) \qquad r = 1, \ldots, s \tag{9.14}$$

with α_r the diagonal and β_r the superdiagonal elements of the tridiagonal $T_j^{(i)}$ ($\beta_1 = 0$). These coefficients are immediately obtained from their relation with the elements of the bidiagonal $J_j^{(i)}$. Let

$$T_j^{(i)} = J_j^{(i)T} J_j^{(i)} = \begin{bmatrix} \alpha_1 & \beta_2 & & 0 \\ \beta_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_s \\ 0 & & \beta_s & \alpha_s \end{bmatrix} \quad \text{and} \quad J_j^{(i)} = \begin{bmatrix} q_1 & e_2 & & 0 \\ & q_2 & \ddots & \\ & & \ddots & e_s \\ 0 & & & q_s \end{bmatrix} \tag{9.15}$$

then $\alpha_i = q_i^2 + e_i^2 \qquad i = 1, \ldots, s \qquad$ and $\quad e_1 = 0$

$\beta_{i+1} = q_i\, e_{i+1} \qquad i = 1, \ldots, s-1 \qquad$ and $\quad \beta_1 = \beta_{s+1} = 0$

9.3.3. QL ITERATIONS

Usually, the smallest diagonal element of the bidiagonal $J^{(0)}$ in (9.2) is the element in the lower right corner, i.e. q_n. This is surely the case if the gap between the singular values is large. Hence, QR iterations are appropriate. However, in some cases, e.g. when the gap is not so large, **the smallest diagonal element of $J^{(0)}$ may be in the upper left corner**, i.e. q_1 in (9.2). In this case **the number of iteration steps can be greatly reduced by using QL iterations.** QL iterations diagonalize the bidiagonal by zeroing the superdiagonal elements from top to bottom while QR iterations start the zeroing process at the bottom superdiagonal element and go on to the top superdiagonal element [14]. In other words, the QL method is identical to the QR method except that at each step the matrix $J^{(i)}$-shift.I is factored into a product of an orthogonal matrix and a lower triangular matrix (with QR, we obtain an upper triangular matrix). For example, consider the 5 x 5 bidiagonal with singular values 6, 5, 4, 3 and 0.9.

$$\begin{bmatrix} 1.18 & 2.47 & 0 & 0 & 0 \\ 0 & -3.05 & 3.05 & 0 & 0 \\ 0 & 0 & -4.73 & -2.25 & 0 \\ 0 & 0 & 0 & -4.00 & -1.98 \\ 0 & 0 & 0 & 0 & -4.75 \end{bmatrix}$$

To compute its minimal singular value to a precision 10^{-15}, 6 QR iteration steps are needed to zero out the bottom superdiagonal element but only 3 QL iteration steps to zero out the top superdiagonal element. Observe that the smallest diagonal element is in the upper left corner.

9.3.4. OUTLINE OF THE PSVD ALGORITHM

In essence, a delay of the initialization phase (section 9.3.1) improves the efficiency of PSVD optimally if the number of desired vectors $\ll n$. The partial diagonalization (section 9.3.2) works best if the gap between the singular values associated with the desired and undesired vectors is large. This is usually the case when solving TLS problems and (non)homogeneous linear equations. In case of a smaller gap, the convergence rate can be improved by using QL iterations (section 9.3.3). All these improvements are incorporated in the following **"partial SVD" algorithm PSVD.**

Algorithm 9-2 : **Partial SVD (PSVD)**

Given : - an m by n matrix A, $m \geq n$
- a bound θ on the singular values σ_i of A associated with the desired singular subspace or its dimension d with an initial estimate of θ,

- logical variables wantu and wantv indicating the need for a basis of a left or right singular subspace respectively,
- and a stop criterion δ defining the minimum subinterval length in the bisection method (step 4 of algorithm 9-1).

Step 1 : **Bidiagonalization phase**

1.a If $m \geq 5n/3$, transform A into upper triangular form R (see (9.9)) by Householder transformations $L^{(1)}, \ldots, L^{(n)}$.

1.b Transform A (or R) into bidiagonal form $J^{(0)}$ (see (9.2)), using Householder transformations $P^{(1)}, \ldots, P^{(p)}$ and $Q^{(1)}, \ldots, Q^{(q)}$ with p = min (m-1, n) and q = min (n-2, m).

1.c If (wantu), initialize G with the m by m identity matrix I_m.
If (wantv), initialize H with the n by n identity matrix I_n.

Step 2 : **Partial diagonalization phase**

If θ is not known, compute θ such that $\#\{\sigma_i(J^{(0)}) \mid \sigma_i \leq \theta\} = d$, using algorithm 9-1.

If $q_n \leq q_1$ in (9.2) then use QR iterations
else use QL iterations.

For i = 0, 1, ...
classify the k unreduced subbidiagonals $J_j^{(i)}$, j = 1, ..., k of the bidiagonal $J^{(i)}$ in (9.12), obtained in the i-th iteration step, into three classes :

$C_1 = \{J_j \mid \sigma_i(J_j) > \theta, \forall i\}$

$C_2 = \{J_j \mid \sigma_i(J_j) \leq \theta, \forall i\}$

$C_3 = \{J_j \mid \exists i : \sigma_i(J_j) \leq \theta \text{ and } \exists t. : \sigma_t(J_j) > \theta\}$

if C_3 empty then go to step 3
else begin
perform one QR/QL iteration step onto each element $J_j^{(i)}$ of C_3 with shift = max (0, min $\{q_j \mid q_j \in \text{diag}(J_j^{(i)})$ and $|q_j| \leq \theta\}$) using Givens rotations $S^{(i)}$ and $T^{(i)}$: $J_j^{(i+1)} = (S^{(i)})^T J^{(i)} T^{(i)}$

if (wantu), apply the Givens rotations $S^{(i)}$ onto G
if (wantv), apply the Givens rotations $T^{(i)}$ onto H.
end
end

Step 3 : **Back transformation phase**

3.a Apply the Householder transformations $P^{(i)}$ (resp., $Q^{(i)}$) onto each vector g_i of G (resp., h_i of H), associated with the bidiagonals in C_2 :
if (wantu) $u_i \leftarrow P^{(1)} \ldots P^{(p)} g_i$ and p = min (m-1,n)
if (wantv) $v_i \leftarrow Q^{(1)} \ldots Q^{(q)} h_i$ and q = min (n-2,m)

3.b If $m \geq 5n/3$ and (wantu), then apply the Householder transformations $L^{(i)}$ onto each computed vector u_i in step 3.a : $u_i \leftarrow L^{(1)} \ldots L^{(n)} u_i$.

END

This algorithm has been programmed in Fortran, fully documented and tested. A program listing is given in [121].

Observe that only minor modifications in the algorithm are required to compute a basis of the singular subspace of a matrix, corresponding to its largest singular values ($> \theta$). Indeed, one must compute another shift in order to converge to the largest singular values, and the base vectors associated with the subbidiagonals of C_1 are now considered.

The same modifications discussed in section 9.3.1, 9.3.2 and 9.3.3, can also be applied to the **generalized SVD** algorithm [80] in so far as it is based on the QR algorithm. They improve the efficiency when only a basis of the singular subspace associated with the largest or smallest generalized singular values is required. In the same way, the **(generalized) symmetric eigenvalue decomposition** algorithm which applies the symmetric QR algorithm after tridiagonalizing the symmetric matrix by Householder transformations [48, section 8.2 - 8.6] can be modified. Its efficiency is improved when only a basis of the eigen subspace associated with its largest or smallest (generalized) eigenvalues is required.

9.3.5. THE CASE m < n

If m < n, we can apply the PSVD algorithm 9-2 on A^T. Sometimes, it may be preferable to work on A. For those cases, the PSVD algorithm in section 9.3.4 can also be used but before the diagonalization phase **one preprocessing step must be included in order to bring the bidiagonal into its right form.** In fact bidiagonalizing an m by n matrix A, m < n, produces an m by m+1 bidiagonal of the form :

$$J^{(0)} = \begin{bmatrix} q_1 & e_2 & & & 0 \\ & q_2 & \ddots & & \\ & & \ddots & e_m & \\ 0 & & & q_m & e_{m+1} \end{bmatrix} \qquad (9.16)$$

Just like in [29, chapter 11], the element e_{m+1} in the lower right corner is cancelled by Givens rotations, which must be performed on the matrix V. If $J^{(0)}$ is unreduced, m Givens rotations are needed. In order to apply the algorithm to all possible m and n, an extra step is needed :

Step 1.d : if m < n then begin
cancel e_{m+1} of $J^{(0)}$ in (9.16) by Givens rotations $F^{(i)}$:

$$J^{(0)} \longleftarrow F^{(1)} \ \ldots \ F^{(r)} \ J^{(0)}, \quad r < m$$

if (wantv), apply the Givens rotations $F^{(i)}$ onto H :

$$H \longleftarrow F^{(1)} \ \ldots \ F^{(r)} \ I_n \ , \quad r < m$$

end.

9.3.6. COMPUTATIONAL CONSIDERATIONS

If m > n, the Givens rotations $S^{(i)}$ associated with each QR/QL iteration step in the partial diagonalization phase, **only have to be applied on the first n rows of G.** This is immediately clear from the fact that the Givens rotations $S^{(i)}$ only affect the first n columns of G. As the last

m-n components of the first n columns of G are set to zero during the initialization of G, they remain unaffected by the Givens rotations $S^{(i)}$. The same applies, if m < n, for the first m columns of H. However, in the preprocessing step, the (n+1)-th row of H is affected by the Givens rotations $F^{(i)}$ and made nonzero. Hence, if m < n, the Givens rotations $T^{(i)}$ have to be applied on the first m+1 rows of H.
The classical SVD algorithm accumulates the Householder transformations $P^{(i)}$ and $Q^{(i)}$ of the bidiagonal phase, already before the diagonalization, in the space provided for U and V respectively. As all elements in U and V are nonzero, the Givens rotations $S^{(i)}$ and $T^{(i)}$ of the diagonalization phase must be applied on all components in the first n columns of U and V (if m > n). Thus, if m > n and vectors from the left singular subspace U of A are wanted, then each Givens rotation in the diagonalization phase costs O(m) operations when using the classical SVD algorithm, while PSVD only requires O(n) operations (see section 9.4.2).

Moreover, **if m > n and only the basis U_2 in (9.1) is required,** i.e. a basis for the orthogonal complement of the range of A, the computations with PSVD can be made very efficient. First, **the partial diagonalization phase can be entirely omitted.** This is clear from the fact that the Givens rotations $S^{(i)}$ of each QR/QL iteration step only operate on the first n columns of U. The last m-n columns, i.e. U_2, are unaffected. Secondly, if we first triangularize A to upper triangular form R, we can omit the bidiagonalization phase of R entirely, if only the basis U_2 in (9.1) is required. This is clear from (9.10) and (9.11) by showing that the matrix U is obtained by multiplying the m by m matrix L (containing the Householder transformations of the triangularization of A) with the m by n matrix $[X^T;0]^T$ (with X containing the Householder transformations of the bidiagonalization of R). As the last m-n columns of $[X^T;0]$ are zero, it follows that the last m-n columns of U, i.e. U_2, are unaffected by the bidiagonalization of R. Thus, if m > n and only the basis U_2 in (9.1) is desired, **the PSVD algorithm reduces to the execution of the following steps :**

Algorithm 9-3 : **U_2 computation**

Step 1 : Transform A into upper triangular form R (see (9.9)), by Householder transformations $L^{(1)} \ \ldots \ L^{(n)}$ from the left.

Step 2 : Apply the Householder transformations $L^{(i)}$ onto the last m-n columns of the m by m identity matrix I_m.

$$U_2 = [u_{n+1}, \ldots, u_m] \longleftarrow L^{(1)} \ldots L^{(n)} \begin{array}{c} \left[\begin{array}{c} 0 \\ \hline 1 \;\ddots\; 0 \\ 0 \quad\; 1 \end{array}\right] \end{array} \begin{array}{l} n \\ \\ m-n \end{array}$$

END

The same applies for the case m < n when only the base vectors $v_{m+1}, \ldots, v_n$ of the kernel of A are required. In that case, A must be triangularized by Householder transformations from the right.

9.3.7. SOLUTION OF A SET OF NONHOMOGENEOUS EQUATIONS

Using the algorithm 9-3, **the computation time for solving nonhomogeneous linear equations AX = B with SVD reduces significantly.** Indeed, we can bring AX = B with A n by n, X and B n by d matrices, into the following form :

$$[X^T;-I]\ [A;B]^T = 0 \tag{9.17}$$

with $[A;B]^T$ an n+d by n matrix of rank n. Taking the SVD (9.1) of $[A;B]^T$, n+d left singular vectors u_i and n right singular vectors v_i can be computed. If $[A;B]^T$ has rank n, then there are exactly d vectors $u_{n+1}, \ldots, u_{n+d}$ in the orthogonal complement of the range of $[A;B]^T$. As the d solution vectors in (9.17) must be orthogonal with respect to the range of $[A;B]^T$, i.e. U_1 in (9.1), they must be parallel with the vectors $u_{n+1}, \ldots, u_{n+d}$, i.e. U_2 in (9.1) :

$$\left[\begin{array}{c} X \\ \hline -I \end{array}\right] /\!/ [u_{n+1}, \ldots, u_{n+d}] = \left[\begin{array}{c} U_{21} \\ \hline U_{22} \end{array}\right] \begin{array}{l} n \\ \\ d \end{array} \tag{9.18}$$

Hence the solution matrix X is obtained from (9.18) by solving

$$XU_{22} = -\ U_{21} \tag{9.19}$$

If the set of equations has only one right handside b, i.e. d = 1 in (9.17), (9.19) reduces to :

$$X = -\ u_{n+1} \ /\ u_{n+1,n+1} \tag{9.20}$$

with $u_{n+1,n+1}$ the (n+1)-th component of the (n+1)-th left singular vector u_{n+1}.

Thus, **if a set of nonhomogeneous equations AX = B must be solved and rank (A) = n, the solution is obtained by the following algorithm :**

Algorithm 9-4 : **Solution of AX = B with n by n matrix A of full rank**

Step 1 : Compute the basis $U_2 = \left[\begin{array}{c} U_{21} \\ \hline U_{22} \end{array}\right] \begin{array}{l} n \\ \\ d \end{array}$ of $[A;B]^T$, using algorithm 9-3.

Step 2 : Solve $XU_{22} = -\ U_{21}$.

END

Even if the rank r of $[A;B]^T$ < n and the minimum norm solution must be computed, a large reduction in computation time is obtained using the PSVD algorithm as only n-r vectors $u_{r+1}, \ldots, u_n$ of $[A;B]^T$ must be processed in the partial diagonalization phase (see section 9.4.2). Indeed, if $[A;B]^T$ has rank r, then the minimum norm solution X is obtained by computing an orthonormal matrix Q such that :

$$[u_{r+1}, \ldots, u_{n+d}]\ Q = \begin{array}{c} \left[\begin{array}{c|c} \diagdown\!\!\diagup & Z \\ \hline 0 \ldots 0 & \Gamma \end{array}\right] \begin{array}{l} n \\ \\ d \end{array} \\ \begin{array}{cc} n-r & d \end{array} \end{array} \tag{9.21}$$

X is obtained by solving

$$\Gamma X = -\ Z \tag{9.22}$$

The solution X in (9.22) is the **minimum norm total least squares solution** of the set AX = B (see section 1.5). If the n-r smallest singular values of [A;B] are exactly zero, X also equals the minimum norm solution obtained by the least squares method.

9.4. COMPARISON OF THE CLASSICAL SVD VERSUS THE PARTIAL SVD ALGORITHM

9.4.1. STORAGE REQUIREMENTS

If $m < 5n/3$, the Householder transformations $P^{(i)}$ (resp., $Q^{(i)}$) are stored in the lower (resp., upper triangular) part of A. If $m \geq 5n/3$ and U is needed, then the Householder transformations $L^{(i)}$ are stored in the lower part of A. The Householder transformations $Q^{(i)}$ onto the rows of the triangular form R of A can be stored in the upper triangular part of A but the Householder transformations $P^{(i)}$ onto the columns of R must be stored in $n(n+1)/2$ extra storage locations. The classical SVD algorithm accumulates the products of the Householder transformations $P^{(i)}$ and $Q^{(i)}$ immediately in U and V respectively, during the initialization phase. Hence, they need not be stored and matrix A can be used to store U or V.

During the partial diagonalization phase of PSVD, the matrices G and H can be stored into the locations provided for the left and right singular vectors of A, U and V respectively. No extra storage locations are needed. Thus by **delaying the initalization phase, PSVD needs m.n extra storage locations if U (or U and V) are needed and n^2 extra storage locations, if only V is needed.** However, the computations are made more efficient. **In the partial diagonalization phase, no extra storage** is required. Moreover, the computation time is greatly reduced (see section 9.4.2).

9.4.2. OPERATION COUNTS

In section 9.3, we indicated that PSVD should be more efficient than the classical SVD algorithm when only a few base vectors of the left and/or right singular subspace of a matrix associated with its smallest singular values, are needed. In this subsection we study the **relative efficiency between the classical SVD and PSVD as a function of m and n.** We do this by computing the number of multiplications (and divisions) for each algorithm. For the classical algorithm, the counts are based on the SVD algorithm DSVDC and the triangularization algorithm DQRDC of the LINPACK Library [29, chapter 9-11] (extra calculations, which deal with

under- and overflow, are not taken into account). In the following operation counts, we assume $m \geq n$ and only consider the third and second order terms in m and n. So the results are correct for moderate m and n. The average number of QR iteration steps for each singular value during the diagonalization in the classical SVD algorithm is denoted by s. The average number of QR/QL iteration steps needed for convergence to one base vector in PSVD is denoted by z and d is the dimension of the desired singular subspace in PSVD.

1. Bidiagonalization (classical SVD and PSVD)

$m < 5n/3$: $2\,mn^2 - \frac{2}{3}n^3 + 2\,n^2$ mult.

$m \geq 5n/3$: $mn^2 - \frac{n^3}{3} + \frac{1}{2}n^2 + mn$ mult. (triangularization of A)

$\frac{4}{3}n^3 + 2\,n^2$ mult. (bidiagonalization of R)

2. Initialization (in classical SVD only)

V : $\frac{2}{3}n^3 - \frac{n^2}{2}$ mult.

if $m < 5n/3$

U_1 : $mn^2 - \frac{1}{3}n^3 + \frac{3}{2}n^2 - nm$ mult.

U : $\frac{2}{3}m^3 - \frac{2}{3}(m-n)^3 + mn - \frac{1}{2}n^2$ mult.

if $m \geq 5n/3$

U_1 : $2\,mn^2 + \frac{1}{3}n^3 + 2\,n^2 - nm$ mult.

U : $\frac{4}{3}n^2 + 2\,m^2n - mn^2 + mn$ mult.

3. QR/QL iterations (classical SVD and PSVD)

classical SVD (U only) : $2\,sn^2m + 10\,sn^2 + (4-2s)nm$ mult.

classical SVD (V only) : $2\,sn^3 + 8\,sn^2$ mult.

classical SVD (U and V required) : $2\,sn^2m + 2\,sn^3 + 8\,sn^2 + (4-2s)nm$ mult.

PSVD (d vectors u_i only) : $4\,zdn^2 + 2\,zdn\,(9-d)$ mult.

PSVD (d vectors v_i only) : $4\ zdn^2 + 2\ zdn\ (9-d)$ mult.

PSVD (d vectors u_i and v_i required) : $8\ zdn^2 + 4\ zdn\ (4-d)$ mult.

4. Back transformation (in PSVD only)

for each vector $v_i \in V$: n^2 mult.

if $m < 5n/3$, for each vector $u_i \in U$: $2\ mn - n^2$ mult.

if $m \geq 5n/3$, for each vector $u_i \in U_1$: $2\ mn$ mult.

for each vector $u_i \in U_2$: $2\ mn - n^2$ mult.

Some comments are in order :

- Usually, the average number s of QR iteration steps for each singular value during the diagonalization phase in the classical SVD is assumed to be 2 [45] and is experimentally found to be quite accurate.

- The average number z of QR/QL iteration steps needed for **convergence** to one base vector in PSVD is usually slightly larger than s, **depending on the desired numerical accuracy ε, the dimension d of the desired singular subspace and the gap**, i.e. the distance between $\{\sigma_i(A) \mid \sigma_i(A) > \theta\}$ and $\{\sigma_i(A) \mid \sigma_i(A) < \theta\}$. The influence of those parameters on z is illustrated in [2].
First, if the elements of the matrix A are affected by noise, we need not reduce the superdiagonal elements of the bidiagonals $J^{(i)}$ in the partial diagonalization phase of PSVD to machine precision. Instead, we could **stop the iteration process** as soon as the superdiagonal elements are **smaller than a given error parameter** ε, depending on the allowed approximation level. A value of ε which is substantially larger than the machine precision, reduces the number of required QR/QL iterations and thus z. Second, it has been observed experimentally that the **computation time** does **not increase linearly with the dimension d** of the desired singular subspace. On the contrary, the computation time is rather independent of d. This follows from the fact that for large d, the bidiagonal $J^{(i)}$ will preferably break down in subbidiagonals belonging to the classes C_1 or C_2 (see section 9.3.4). Hence, only a few QR/QL iterations are needed to process the remaining subbidiagonals of C_3. Thus, for large d the average number z of QR/QL

iterations will be smaller than for small d. Third, z depends on the gap. If **the gap is large**, i.e. larger than one order of magnitude, z **is only slightly larger than s**. Experimentally z is found to be approximately 3 in those cases with d = 1. But if the gap is small, z will be larger. However, due to the efficient choice between QR and QL iterations (see section 9.3.3), z can be kept small.

- From these operation counts, one can compute **in which cases the delay of the initialization phase** in the classical SVD until the end, i.e. the back transformation phase in PSVD, is **computationally more efficient**. Therefore, the dimension d of the desired singular subspace must be smaller than the ratio of the operation counts needed for the initialization phase to the operation counts of the back transformation in PSVD e.g. :

If only vectors v_i are required, then $d < \frac{2}{3} n$

If $m < 5n/3$ and only vectors $u_i \in U_1$ are required,

then $d < \frac{n}{2} \left(\frac{m-n/3}{m-n/2}\right)$ approximately

If $m < 5n/3$ and vectors $u_i \in U_1$ and v_i are required,

then $d < \frac{n}{2} \left(1 + \frac{n}{3m}\right)$ approximately

- We can **distinguish between four cases in the comparison** :
case a : only d vectors v_i are required explicitly.
case b : only d vectors $u_i \in U_1$ are required explicitly.
case c : d vectors $u_i \in U_1$ and d vectors v_i are required explicitly.
case d : d vectors $u_i \in U_1$ and the basis U_2 are required explicitly.
Case a arises in total least squares applications and in the solution of homogeneous linear equations (see section 9.1).
Case b is case c for A^T.
Case c arises in the computation of singular subspaces and case d (and also case a) arises in the computation of the null space of a perturbed matrix whose rank depends on a given perturbation bound θ. The total number of multiplications (and divisions) for each case is given in table 9.1.

TABLE 9.1. : Total number of operations of the classical SVD and PSVD for each of the cases a, b, c, d (see section 9.4.2).

Case	Classical SVD	PSVD
a. $m < 5n/3$:	$2mn^2 + 2sn^3 + (\frac{3}{2} + 8s)n^2$	$2mn^2 - \frac{2}{3}n^3 + [(4z + 1)\,d + 2]n^2$
$m \geq 5n/3$:	$mn^2 + (\frac{5}{3} + 2s)n^3 + (2 + 8s)n^2 + mn$	$mn^2 + n^3 + [(4z + 1)\,d + 5/2]n^2 + mn$
b. $m < 5n/3$:	$(3 + 2s)mn^2 - n^3 + (\frac{7}{2} + 10s)n^2 + (3 - 2s)nm$	$2mn^2 - \frac{2}{3}n^3 + [(4z - 1)\,d + 2]n^2 + 2dmn$
$m \geq 5n/3$:	$3mn^2 + (\frac{4}{3} + 2s)n^3 + (\frac{9}{2} + 8s)n^2$	$mn^2 + n^3 + (4zd + 5/2)n^2 + (2d + 1)mn$
c. $m < 5n/3$:	$(3 + 2s)mn^2 + (2s - \frac{1}{3})n^3 + (3 + 8s)n^2 + (3 - 2s)mn$	$2\,mn^2 - \frac{2}{3}n^3 + (8zd + 2)n^2 + 2dmn$
$m \geq 5n/3$:	$3mn^2 + (2 + 4s)n^3 + (4 + 8s)n^2$	$mn^2 + n^3 + [(8z + 1)\,d + 5/2]n^2 + (2d + 1)mn$
d. $m < 5n/3$:	$2m^2n + 2sn^2m + (\frac{3}{2} + 10s)n^2 + (5 - 2s)mn$	$2m^2n - mn^2 + \frac{1}{3}n^3 + (4zd-d-1)n^2 - m^2 + (2d + 4)mn$
$m \geq 5n/3$:	$2m^2n + (\frac{7}{3} + 2s)n^3 + (\frac{5}{2} + 8s)n^2 + 2mn$	$2m^2n - 2mn^2 + 2n^3 + (4zd - \frac{1}{2})n^2 - m^2 + (2d + 5)mn$

Using table 9.1, we can compute approximations of the ratios of the operation counts of the classical SVD to that of PSVD in each of the four cases. The results are given in table 9.2 where the ratio is expressed as a function of C = m/n. These ratios show the relative efficiency of PSVD in each of the four cases, mentioned above.

It can be seen that **the relative efficiency of PSVD increases when the dimension d of the desired singular subspace is relatively smaller w.r.t. the dimension n**, i.e. the ratio d/n decreases. If the average number z of QR/QL iteration steps in PSVD is comparable with the average number s of QR iteration steps in the classical SVD, PSVD is about 2.5 times faster than classical SVD in the cases a and d, three times faster in case b and 3.5 times faster in case c. The relative efficiency of PSVD with respect to classical SVD decreases with growing ratio m/n in the cases a and d, and increases in case b. In case c, the relative efficiency of PSVD is quasi independent of the ratio m/n. Some numerical results for case a are given in [2].

- With respect to **solving nonhomogeneous equations** we can also compare the operation counts of algorithm 9-4 with those of other currently used methods. For solving AX = B with X and B n by d matrices and A of full rank, algorithm 9-4 requires

$$\frac{2}{3}n^3 + (\frac{3}{2} + 2d)\,n^2 \text{ multiplications.}$$

These counts can be compared with (only third order terms are taken) :

Gauss elimination method [48, section 4.2] $\sim \frac{1}{3}n^3$

Householder triangularization [48, section 6.2] $\sim \frac{2}{3}n^3$

SVD method (SVD of A) [48, section 6.5], [100] $\sim 6\,n^3$

Classical TLS algorithm (see section 1.8.1) $\sim 6\,n^3$

Moreover if the rank of [A;B] < n, the singular vector u_n corresponding to the smallest singular value of $[A;B]^T$, must be computed to obtain the minimum norm solution (see section 9.3.7). This is easily done using the PSVD algorithm. The operation counts are $\sim \frac{4}{3}n^3$. This result is much better than the efficiency of the ordinary least

TABLE 9.2. : Ratio of Operation Counts of the classical SVD to that of PSVD for each of the cases a, b, c, d (see section 9.4.2).

Case	Ratio (C = m/n)	s = 2
a. $m < 5n/3$	$(C + s)/(C - \frac{1}{3} + 2z\frac{d}{n})$	$(C + 2)/(C - \frac{1}{3} + 2z\frac{d}{n})$
$m \geq 5n/3$	$(C + 2s + \frac{5}{3})/(C + 1 + 4z\frac{d}{n})$	$(C + \frac{17}{3})/(C + 1 + 4z\frac{d}{n})$
b. $m < 5n/3$	$((3 + 2s)C - 1)/(2C - \frac{2}{3} + 4z\frac{d}{n})$	$(7C - 1)/(2C - \frac{2}{3} + 4z\frac{d}{n})$
$m \geq 5n/3$	$(3C + \frac{4}{3} + 2s)/(C + 1 + 4z\frac{d}{n})$	$(3C + \frac{16}{3})/(C + 1 + 4z\frac{d}{n})$
c. $m < 5n/3$	$((3 + 2s)C + 2s - \frac{1}{3})/(2C - \frac{2}{3} + 8z\frac{d}{n})$	$(7C + \frac{11}{3})/(2C - \frac{2}{3} + 8z\frac{d}{n})$
$m \geq 5n/3$	$(3C + 2 + 4s)/(C + 1 + 8z\frac{d}{n})$	$(3C + 10)/(C + 1 + 8z\frac{d}{n})$
d. $m < 5n/3$	$(2C^2 + 2sC)/(2C^2 - C + \frac{1}{3} + 4z\frac{d}{n})$	$(2C^2 + 4C)/(2C^2 - C + \frac{1}{3} + 4z\frac{d}{n})$
$m \geq 5n/3$	$(C^2 + s + \frac{7}{6})/(C^2 - C + 1 + 2z\frac{d}{n})$	$(C^2 + \frac{19}{6})/(C^2 - C + 1 + 2z\frac{d}{n})$

squares method based on the SVD of A ($\sim 6\ n^3$) while the same numerical accuracy is maintained.

9.4.3. COMPUTATIONAL RESULTS

The analysis based on the operation counts in section 9.4.2 **agrees well with our computational experiments** for moderate n. All tests were run on the IBM 3030 of the Central Computer Center at the Katholieke Universiteit Leuven. Double precision was used throughout the calculation. The SVD routine of the LINPACK Library [29, chapter 11] was used as classical SVD routine. The matrices were randomly generated with specified singular value spectrum. In our examples given below, ε denotes the numerical accuracy. θ denotes the bound separating the singular values associated with the desired and undesired singular subspace. The CPU times are expressed in milliseconds.

Example 9-1 :

A is a 22 by 20 matrix with singular values 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, .9, .8, .7, .6, .5, .4, .3, .2, .1 and .0001; $\varepsilon = 10^{-13}$ and $\theta = .01$. Hence, only the singular vector corresponding to $\sigma_{20} = .0001$ is desired. 2 QR iteration steps were required for convergence. The following CPU times were obtained if cases a, b, c and d (defined in section 9.4.2) are considered :

	a		b		c		d	
	PSVD	DSVDC	PSVD	DSVDC	PSVD	DSVDC	PSVD	DSVDC
bidiagonalization	43	43	43	43	43	43	43	43
initialization U or V		17		19		38		24
diagonalization	7	80	7	84	12	129	7	85
back transformation	2		2		4		6	
Total	52	140	52	146	59	210	56	152

Example 9-2 :

A is a 20 by 8 matrix with singular values 1000, 500, 100, 50, 10, 5, 1 and .01; $\varepsilon = 10^{-13}$ and $\theta = .5$. Hence, only one singular vector must be

computed. 3 QR iterations were required for convergence. If the case a, b, c and d in section 9.4.2 are considered, the following CPU times were obtained :

	a		b		c		d	
	PSVD	DSVDC	PSVD	DSVDC	PSVD	DSVDC	PSVD	DSVDC
triangularization	4	4	4	4	4	4	4	4
bidiagonalization of R	4	4	4	4	4	4	4	4
initialization U or V of R		2		2		3		2
diagonalization of R	3	9	3	9	4	12	3	9
back transformation	0	0	2	13	2	13	11	22
Total	11	19	13	32	14	36	22	41

Example 9-3 :

A is a 15 by 12 matrix with singular values 1000, 800, 400, 100, 80, 40, 20, 10, 8, 1, .1, .01. $\varepsilon = 10^{-13}$ and $\theta = 5$. Hence, 3 base vectors are required. The total number of QR iteration steps was 5. As the diagonal element q_1 of the bidiagonal $J^{(0)}$ in (9.2) was larger than q_{12}, it is expected that QR iterations are most efficient. Indeed, the total number of QL iteration steps needed for convergence to three base vectors is 7. The following CPU times were obtained if cases a, b, c and d (defined in section 9.4.2) are considered :

	a		b		c		d	
	PSVD	DSVDC	PSVD	DSVDC	PSVD	DSVDC	PSVD	DSVDC
bidiagonalization	13	13	13	13	13	13	13	13
initialization U or V		4		6		10		9
diagonalization	7	20	7	20	12	29	8	21
back transformation	3		3		5		5	
Total	23	37	23	39	30	52	26	43

Example 9-4 :

A is a 18 by 11 matrix with singular values 10, 8, 4, 2, 1, .8, .4, .2, .1, .01, .001. $\varepsilon = 10^{-13}$ and $\theta = .05$. Hence a singular subspace of dimension $d = 2$ is requested. The total number of QR iterations was 4. The

following CPU times have been obtained for the cases a, b, c and d of section 9.4.2 :

	a		b		c		d	
	PSVD	DSVDC	PSVD	DSVDC	PSVD	DSVDC	PSVD	DSVDC
triangularization	6	6	6	6	6	6	6	6
bidiagonalization of R	9	9	9	9	9	9	9	9
initialization U or V of R		3		3		6		3
diagonalization of R	6	17	6	17	9	23	6	17
back transformation	1	0	3	20	4	21	10	27
Total	22	35	24	55	28	65	31	62

The following examples illustrate the efficiency of solving a set of nonhomogeneous equations using algorithm 9-4. They are compared with the CPU times needed to solve the set with the ordinary least squares method based on SVD, e.g. the procedure MINFIT in the EISPACK library [100].

Example 9-5 :

[A;b] is a 10 by 11 matrix with singular values 10, 9, 8, 7, 6, 5, 4, 3, 2, 1. $\varepsilon = 10^{-13}$.

	Algorithm 9-4	MINFIT
triangularization of [A;b]	4	
bidiagonalization of A		7
initialization V of A		3
diagonalization of A		16
back transformation	1	
solution $x = V\Sigma^{-1} U^T b$		1
Total	5	27

Example 9-6 :

[A;b] is a 15 by 16 matrix with singular values 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, .9, .8, .7, .6, .5. $\varepsilon = 10^{-13}$.

	Algorithm 9-4	MINFIT
triangularization of [A;b]	10	
bidiagonalization of A		20
initialization V of A		2
diagonalization of A		36
back transformation	2	
solution $x = V\Sigma^{-1} U^T b$		2
Total	12	66

We also checked the accuracy of some of the computed results. The desired vectors of the matrices U and V in (9.1) agree to within a few units of the machine precision in almost all the cases but the signs of the corresponding vectors may be reversed. In fact, the SVD is only unique to within such a sign change.

9.5. THE PARTIAL TLS ALGORITHM PTLS

9.5.1. OUTLINE OF THE PTLS ALGORITHM

The computational efficiency of the classical TLS algorithm 1-1, presented in section 1.8.1, can be improved using the "partial SVD" instead of the classical SVD algorithm in step 1. Indeed, as shown in section 1.8.1, the TLS solution of (0.1) can be obtained from any orthogonal basis of the right singular subspace associated with the smallest n+d-r singular values of the m by n+d matrix [A;B] of rank r (see (1.116)). This improvement is incorporated in the following **"partial TLS" algorithm PTLS.**

Algorithm 9-5 : **Partial TLS (PTLS) computation of the TLS solution $\hat{X}$ of AX = B.**

Given : - an m by n data matrix A and an m by d observation matrix B.
- a rank determinator θ on the singular values σ_i of [A;B] (such that $\sigma_r[A;B] > \theta \geq \sigma_{r+1}[A;B]$) or the desired rank r of [A;B] with an initial estimate of θ (the dimension of the desired singular subspace is then n+d-r).
- and a stop criterion δ defining the minimum subinterval length in the bisection method (step 4 in algorithm 9-1).

wantu = .false.
wantv = .true.

Step 1 : **Bidiagonalization phase**

perform step 1 of the PSVD algorithm 9-2

Step 2 : **Partial diagonalization phase**

perform step 2 of the PSVD algorithm 9-2.

Step 3 : **Back transformation phase**

perform step 3 of the PSVD algorithm 9-2

Step 4 : Compute a Householder matrix Q such that

$$[v_{r+1}, \ldots, v_{n+d}]\, Q = \begin{bmatrix} \times & Z \\ 0 & \Gamma \end{bmatrix} \begin{matrix} n \\ d \end{matrix} \qquad (9.23)$$

(n-r columns, d columns)

with Γ a d by d upper triangular matrix.

Step 5 : If Γ singular then begin
lower the rank r with one
go back to step 2
end
else solve $\hat{X}\Gamma = -Z$ by forward elimination
END

Some comments are in order :

- This algorithm has been programmed in Fortran, fully documented and tested. A program listing is given in [121].

- In case the rank must be lowered in step 5, some additional base vectors must be computed in step 2. The additional computations can be kept to a minimum. Indeed, we only have to further diagonalize the

partially reduced bidiagonal by zeroing out some more superdiagonal elements (step 2) and back transforming these newly computed base vectors (step 3). The base vectors which were already computed, remain unaffected.

- If the rank r is lowered in step 5 but σ_r is of multiplicity > 1, the rank r will be further reduced with the multiplicity of σ_r in the beginning of step 2 by the θ-estimation algorithm 9-1 (see step 4). Here, the multiplicity of σ_r is determined by the number of singular values lying within an interval of length δ around σ_r. Observe however that this definition is rather arbitrary and does not allow to check the multiplicity of singular values as efficiently and reliably as the classical TLS algorithm (see section 1.8.1).

9.5.2. COMPARISON OF THE CLASSICAL TLS VERSUS THE PARTIAL TLS ALGORITHM

Since the TLS computations are entirely dominated by the SVD, it is evident that **a comparison between classical TLS and PTLS reduces mainly to a comparison between classical SVD and PSVD**, case a, discussed in section 9.4.1. However, some **additional comments** are in order.

Concerning the storage requirements, **PTLS needs more memory space**, namely $(n+d)(n+d-1)/2$ (resp., $m(n+d-(m+1)/2)$) extra storage locations in case $m \geq n+d$ (resp., $m < n+d$). Indeed, the Householder transformations $Q^{(i)}$ performed onto the rows of $[A;B]$ during its bidiagonalization (see step 1) must be stored until the end of the computations (step 5). The classical TLS algorithm accumulates the products of the Householder transformations $Q^{(i)}$ immediately into the right singular base matrix V after the bidiagonalization. Hence, they need not be stored and matrix $[A;B]$ can be used to store V.

W.r.t. **the operation counts**, the following comments are worth mentioning.
Once the desired base vectors of $[A;B]$ are computed by classical SVD or PSVD, the TLS solution of (0.1) must be calculated. Hereto we first reduce those base vectors with Householder transformations to the form (1.113) (step 3 of algorithm 1-1 or step 4 of algorithm 9-5) and then

solve

$$\hat{X}\Gamma = -Z \tag{9.24}$$

by forward elimination (step 4 of algorithm 1-1 or step 5 of algorithm 9-5). If the rank of $[A;B]$ equals n, d base vectors have been calculated. In this case, the computation of the TLS solution from (9.24) requires

$$[(9n + 4d + 12)\, d^2 + (15n + 26)\, d] / 6 - 3n - 7 \quad \text{multiplications} \tag{9.25}$$

Observe that (9.25) reduces to n if d = 1. This is indeed the number of multiplications required to scale the (n+1)-th right singular vector of $[A;b]$ and obtain $\hat{x}$ (see (1.16)).
The number of operations of the classical TLS algorithm and the partial TLS algorithm are immediately obtained by adding (9.25) to the operation counts, given in table 9.1, case a, and replacing n by the number of columns in $[A;B]$, namely n+d.

Finally, we give some **computational results**. Observe that the comparison between classical TLS and PTLS largely agrees with the computational results of the classical SVD and PSVD. Indeed, when $d \ll n$, the additional computations (9.25) to obtain the TLS solution, are negligible w.r.t. the computations of the desired base vectors by classical SVD or PSVD.
The matrices $[A;B]$ were randomly generated with specified singular value spectrum. All tests were run on the IBM 3030 of the Central Computer Center at the Katholieke Universiteit Leuven. ε denotes the numerical accuracy. Double precision was used throughout the calculations. The CPU times are expressed in milliseconds.

Example 9-7 :

Consider the set equations (0.1) with A a 30 by 16 matrix and one observation vector b(d=1). The singular values of $[A;B]$ whose rank is 16, are 1000, 800, 400, 200, 100, 80, 40, 20, 10, 8, 4, 2, 1, .8, .4, .2 and .01. Hence, only one solution vector must be computed. $\varepsilon = 10^{-13}$. 3 QR iterations were required for convergence. The following CPU times were obtained :

	PTLS	classical TLS
triangularization of [A;b]	21	21
bidiagonalization of R	28	28
initialization V of R		11
diagonalization	8	47
back transformation	1	
solve $\hat{x}\gamma = -z$	0	0
Total	58	107

Example 9-8 :

Consider the set equations (0.1) with A a 15 by 10 matrix and 3 observation vectors (d=3). The singular values of [A;B] whose rank is 10, are 100, 50, 28, 26, 24, 22, 20, 18, 15, 12, .3, .2 and .1. Hence, 3 solution vectors must be computed. $\varepsilon = 10^{-13}$. 5 QR iterations were required for convergence. The following CPU times were obtained.

	PTLS	classical TLS
bidiagonalization of [A;B]	15	15
initialization V		5
diagonalization	9	32
back transformation	2	
solve $\hat{x}\Gamma = -z$	1	1
Total	27	53

9.6. CONCLUDING REMARKS

Since the TLS solution of (0.1) is deduced from only one right singular vector or in general, a basis of the right singular subspace associated with the smallest singular values of the data [A;B], its **computational cost can be considerably reduced by computing its SVD only partially.**

Hereto, **an improved algorithm PSVD** is presented in this chapter comput-

ing the singular subspace of a matrix, associated with its smallest singular values. There are **three reasons for its high efficiency with respect to the classical SVD algorithm.** First, the Householder transformations of the bidiagonalization are only applied on the base vectors of the desired singular subspace. Second, the bidiagonal is only partially diagonalized and third, a choice is made between QR and QL iteration steps (section 9.3). By analyzing the operation counts, the relative efficiency of PSVD with respect to classical SVD is demonstrated and confirmed with computational results (section 9.4). Depending on the gap, the desired numerical accuracy and the dimension of the desired subspace, **PSVD can be three times faster than the classical SVD algorithm while the same accuracy can be maintained.** Using the new algorithm, **a large reduction in computation time can be obtained in total least squares applications, in the computation of the null space of a matrix and in solving (non)homogeneous linear equations.** Based on PSVD a very efficient and reliable algorithm is also derived for solving nonhomogeneous linear equations (section 9.3.7). From a practical point of view, one may observe that in general the classical SVD algorithm is not likely to be improved in speed but is used extensively in many applications where the above mentioned problems occur. Hence the improved speed of PSVD should prove to be quite valuable for all these applications.

Incorporating the "partial SVD" algorithm into the TLS computations, results in an **improved "partial TLS" algorithm PTLS.** Its higher efficiency w.r.t. the classical TLS algorithm (see section 1.8.1) is demonstrated and confirmed with computational results (section 9.5). Typically, PTLS reduces the computation time with a factor 2.

CHAPTER 10

ITERATIVE SPEED IMPROVEMENTS OF THE TLS COMPUTATIONS

10.1. INTRODUCTION

In the previous chapter, a speed improvement of the TLS computations is obtained in a direct way by computing only that part of the complete SVD that is needed. However, **no adaptive version is developed such that a priori information**, e.g. the TLS solution of a previous time step, **could be incorporated** and reduce further the computation time. This happens for instance if slowly varying sets of equations have to be solved at each time instant. Two types of varying sequences of data matrices are possible : adding a new column or row and omitting one (for instance, when new data become available and old data are to be deleted). Another possibility is time variance elementwise when all elements change slowly in each time step. For those cases, the TLS solution on two subsequent time steps are related. Hence, the TLS solution on step t could be used as initial guess for the solution on t+1 in an **iterative algorithm.** All methods described in this chapter, are iterative and can be more time efficient for these applications than the classical TLS and PTLS algorithm which make no use of an a priori available initial guess. There are still other reasons for preferring iterations in high speed computing. Each step of an iteration routine gives a new and better estimate of the solution. This allows to control the level of convergence depending on the perturbations of the data. Furthermore, iteration is easy to code. Finally, in linear algebra an iteration uses the given matrix over and over again and does not modify the matrix during the computation. This is most important if the matrix itself is highly structured e.g. sparse, Hankel, Toeplitz, ...

In this chapter, **four iteration methods are described and their efficiency evaluated w.r.t. the TLS problem.** First of all in section 10.2, a common theoretical basis for better understanding and relating the apparently distinct iteration techniques is given. Based on it, we could deduce some new general and very useful convergence formulas. These expressions allow to compare the convergence properties of the iteration techniques under study straightforwardly and clarify their differences in relative efficiency w.r.t. the TLS problem. Section 10.3, 10.4 and 10.5 describe the **Chebyshev iteration**, the **inverse iteration** technique and **Rayleigh quotient iteration** method respectively. The algorithms, their convergence properties and applicability w.r.t. adaptive TLS problems are discussed. Rayleigh quotient iteration is considered as a variant of inverse iteration with variable shifts. Its rapid convergence when it does converge to an eigenvalue/eigenvector pair, is well-known. In contrast to these methods which compute the desired eigenvectors and eigenvalues of the matrix involved, the **Lanczos method** involves a tridiagonalization (or bidiagonalization) of the given matrix. Unlike the Householder approach, no intermediate (and full) submatrices are generated. Equally important information about the extremal eigenvalues tends to emerge long before the tridiagonalization is complete. This makes the Lanczos algorithm particularly useful in situations where a few of the largest or smallest eigenvalues of a matrix are desired. The derivation and exact arithmetic properties of the method are briefly presented in section 10.6. Unfortunately, roundoff errors make the method somewhat difficult to use in practice. This shortcoming can be overcome in a number of ways.

Based on the convergence rate and the number of operations in each iteration step all iterative methods described here are evaluated w.r.t. their efficiency in TLS problems and **compared with the direct computation methods** : classical SVD and PSVD (see chapter 9) in section 10.7. It is shown how **inverse iteration is the most promising technique for solving one-dimensional generic TLS problems with unique solution.** In particular in section 10.8, the efficiency of the inverse iteration method, the PTLS and classical TLS algorithm is compared in a practical **real life problem** concerning the estimation of frequency response functions.

10.2. MATRIX FUNCTIONS

The theory of matrix functions allows to derive new general interesting convergence formulas, making a unified approach of the iteration techniques described in this chapter possible w.r.t. their convergence properties and efficiency. It is well-known [38] that one can define functions on matrices, called **matrix functions**, certainly if those functions are analytic. From the point of view of abstract algebraic structure, the computation with a given matrix S and its powers is isomorphic to the computation in the ring of polynomials in a variable λ. Orthogonality of polynomials is important because orthogonal polynomials are distinguished by extremum properties [90-106]. Based on this isomorphism one can prove the following theorem [38] for any analytic function f_k and hence for every polynomial. Consider a real n by n symmetric matrix S with eigenvalue decomposition of the form

$$S = V \wedge V^T \tag{10.1}$$

where V are the n eigenvectors and $\wedge = \text{diag}\ (\lambda_1, \ldots, \lambda_n)$. Then the matrix function f_k satisfies :

$$f_k\ (S) = V\ f_k\ (\wedge)\ V^T \quad \text{and} \quad f_k\ (\wedge) = \text{diag}\ (f_k\ (\lambda_1), \ldots, f_k\ (\lambda_n)) \tag{10.2}$$

In the case where S is positive definite, the eigenvalue decomposition equals the SVD. Of course, expression (10.2) also applies to an arbitrary m by n rectangular matrix A. Indeed, since the matrices A^TA and AA^T are real symmetric, we obtain from the relation between their eigenvalues and the SVD (9.1) of A :

$$f_k\ (A^TA) = V\ f_k\ (\Sigma^2)\ V^T \tag{10.3}$$

$$f_k\ (AA^T) = U_1\ f_k\ (\Sigma^2)\ U_1^T \tag{10.4}$$

where $\lambda_i\ (AA^T) = \lambda_i\ (A^TA) = \sigma_i^2\ (A)$ for $i = 1, \ldots, \min\ (m,n)$.

Consider now an arbitrary n-dimensional unit vector q_1 which has the following coordinates in the eigenvector basis V, called **internal coordinates** :

$$V^T q_1 = [g_1, \ldots, g_n]^T \quad \text{and} \quad g_i = v_i^T q_1 \quad i = 1, \ldots, n \tag{10.5}$$

It is not difficult to see that the internal coordinates of the vector $f_k\ (S)\ q_1$ in the eigenvector basis V are given by :

$$V^T\ f_k\ (S)\ q_1 = [f_k\ (\lambda_1)\ g_1, \ldots, f_k\ (\lambda_n)\ g_n] \tag{10.6}$$

Suppose now that one wants to compute the i-th eigenvector v_i, $i \leq n$, with internal coordinates :

$$V^T v_i = [0, \ldots, \underset{\uparrow}{1}, 0, \ldots, 0]^T \tag{10.7}$$
$$\qquad\qquad\quad i$$

to a precision ε. For example, in one-dimensional generic TLS problems (0.1), the (n+1)-th eigenvector of the matrix $[A;b]^T\,[A;b]$ has to be computed. Then **the sequence of functions f_k has to be chosen in such a way that $f_k\ (S)\ q_1$ converges to the desired eigenvector up to a precision ε.** This implies that there must exist a finite K such that the internal coordinates of $V^T\ f_K\ (S)\ q_1$ satisfy :

$$|f_K\ (\lambda_i)\ g_i| \ /\ |f_K\ (\lambda_j)\ g_j| > 1/\varepsilon \quad \forall j \neq i \tag{10.8}$$

or

$$(\max_{j \neq i} |f_K\ (\lambda_j)\ v_j^T q_1|) \ /\ |f_K\ (\lambda_i)\ v_i^T q_1| < \varepsilon \tag{10.9}$$

Taking φ_i as the angle between v_i and q_1, the convergence requirement is more elegantly expressed by :

$$(\max_{j \neq i} |f_K\ (\lambda_j)|)\,|\tan \varphi_i| \ /\ |f_K\ (\lambda_i)| < \varepsilon \tag{10.10}$$

Since $\cos \varphi_i = |v_i^T q_1|$ and $\sin \varphi_i = 1 - \cos^2 \varphi_i \geq \max_{j \neq i} |v_j^T q_1|$, (10.10) is more severe than (10.9).

For example, the **power method** [82] applies the functions $f_k\ (\lambda) = \lambda^k$ onto the matrix S to converge to its greatest eigenvector/eigenvalue pair. **Inverse iteration** uses the functions $f_k\ (\lambda) = (\lambda - \lambda_o)^{-k}$ with λ_o an appropriately chosen shift, in order to converge to the eigenvector associated with the eigenvalue which approaches λ_o most closely. If λ_o varies in each iteration step, we obtain the **Rayleigh quotient iteration** technique. Finally, by applying the orthogonal Chebyshev polynomials $C_k^{yz}(\lambda)$ defined over an appropriate interval [y,z] containing all undesired eigenvalues of S, the **Chebyshev iteration** method allows to converge to an eigenvector of S associated with an eigenvalue outside [y,z].

For all those iteration methods the ratio $f_k(\lambda_j)/f_k(\lambda_i)$ can be written as $(f_1(\lambda_j)/f_1(\lambda_i))^k$. Hence, by taking the logarithm (with base 10) of (10.8), we can derive **the number of iteration steps K required for convergence** to the desired eigenvector up to a precision ε :

$$K > \frac{\log \varepsilon^{-1} + \log |v_j^T q_1 / v_i^T q_1|}{\log |f_1(\lambda_i) / f_1(\lambda_j)|}$$

$$\text{with } |f_K(\lambda_j)\, v_j^T q_1| = \max_{k \neq i} |f_K(\lambda_k)\, v_k^T q_1| \qquad (10.11)$$

or analogously, from (10.10) :

$$K > \frac{\log \varepsilon^{-1} + \log |\tan \varphi_i|}{\log (|f_1(\lambda_i)| / \max_{j \neq i} |f_1(\lambda_j)|)} \qquad 1 \leq i \leq n,\ 1 \leq j \leq n \qquad (10.12)$$

The derived formula is of course very interesting since it shows clearly **the 3 different effects that influence** the number of iteration steps :

1. $\log |f_1(\lambda_i) / f_1(\lambda_j)|$: the larger **the gap between the function value $f_1(\lambda)$ of the desired eigenvalue λ_i w.r.t. the remaining eigenvalues λ_j**, the faster the convergence. If $f_1(\lambda_i) / f_1(\lambda_j)$ approaches one, convergence will be very slow.
2. $\log |v_j^T q_1 / v_i^T q_1|$: the better **the quality of the initial guess q_1**, the smaller the number of iterations. The internal coordinates (10.6) of a very good initial vector q_1 satisfy $q_i >> q_j$, $\forall j \neq i$.
3. $\log \varepsilon^{-1}$: this quantity measures **the number of desired correct digits** in the iteration vector converging to the desired eigenvector v_i. If, for instance, t correct digits are necessary, one sets $\varepsilon = 10^{-t}$.

Hence the numerator of (10.11) measures the number of digits of the initial guess vector q_1 that has to be modified in order to obtain the desired eigenvector v_i up to a precision ε. Observe that **the numerator of (10.11) describes the initial state information, while the denumerator determines the convergence rate**, i.e. the number of correct decimal digits that can be gained in each iteration step.

To **control the convergence of the iterative algorithm** in practice one could use expression (10.11) or (10.12). However this would imply estimates of the quality $\log |v_j^T q_1 / v_i^T q_1|$ of the initial guess vector q_1 and the convergence rate $\log |f_1(\lambda_i) / f_1(\lambda_j)|$. This of course complicates the software implementation. The most practical test is the classical convergence test :

$$\|q_{k+1} - q_k\|_2 \overset{?}{\gtrless} TOL \qquad \text{with } \|q_{k+1}\|_2 = \|q_k\|_2 = 1 \qquad (10.13)$$

q_{k+1} and q_k are two consecutive iteration vectors and TOL is a user defined accuracy tolerance depending upon the precision ε that is asked for.

Still other convergence tests exist but are computationally complexer, e.g. a test based on the backward error matrix [24].

10.3. CHEBYSHEV ITERATION

10.3.1. CHEBYSHEV POLYNOMIALS

Chebyshev polynomials $C_k(x)$ are orthogonal over an interval $[-1,1]$ w.r.t. the density function $1/\sqrt{1-x^2}$. They can be defined in several ways [90] but we only need here some important, well-known properties. As all orthogonal polynomials, the Chebyshev polynomials satisfy a three term recurrence relation :

$$C_o(x) = 1\ ; \quad C_1(x) = x$$

$$C_{k+1}(x) = 2x\, C_k(x) - C_{k-1}(x) \qquad (10.14)$$

They are defined by :

$$C_k(x) = \cos(k \arccos x) = \cos k\varphi \qquad |x| \leq 1 \qquad (10.15)$$

$$= \cosh(k \operatorname{arccosh} x) \qquad |x| > 1 \qquad (10.16)$$

$$= 0.5\, [(x + \sqrt{x^2 - 1})^k + (x + \sqrt{x^2 - 1})^{-k}] \qquad |x| > 1 \qquad (10.17)$$

$$\approx 0.5\, (2x)^k \quad \text{as} \quad x \to \infty \quad k \text{ fixed} \qquad (10.18)$$

Hence,

$$|C_k(x)| \leq 1 \quad , |x| \leq 1, \ \forall k \qquad (10.19)$$

$$C_k(1) = 1 \qquad \forall k \qquad (10.20)$$

The behaviour of x outside [-1,1] is characterized by

$$|C_k(x)| \geq 1 \qquad \forall k \tag{10.21}$$

$$|C_{k+1}(x)| > |C_k(x)| \tag{10.22}$$

$$|C_k(x + \Delta x)| > |C_k(x)| \quad (\Delta x > 0) \tag{10.23}$$

$$C_k(1 + 2\varepsilon) \approx 0.5\,(1 + 2\sqrt{\varepsilon} + 2\varepsilon)^k \quad \text{for } 0 \leq \varepsilon < 0.1 \tag{10.24}$$

$$\approx 0.5 \exp(2k\sqrt{\varepsilon}) \quad \text{for } k\sqrt{\varepsilon} > 1 \tag{10.25}$$

It is this rapid growth of $C_k(1 + 2\varepsilon)$ for small ε and large k that makes the Chebyshev polynomials attractive for computation of the partial SVD. The optimal characteristics of C_k stem from its famous equioscillation property : on [-1,1], C_k switches between its maximal absolute value 1 with alternating signs exactly k times. No other polynomial of degree k or less can equioscillate more, apart from multiples of C_k. Observe from (10.15) that the extrema and zeros of C_k are symmetrically located around x = 0. The following theorem, proven in [90], is extremely important for the Chebyshev iterative algorithm :

Theorem 10-1 :

The Chebyshev polynomial $2^{1-k} C_k(x)$ is always the largest possible monic (with leading coefficient 1) polynomial of degree k outside [-1,1] : for every monic polynomial $p_k(x)$ of degree k, we have with $M = \max |p_k(x)|$, $x \in [-1,1]$ and for $|y| > 1$ that $|p_k(y)| \leq M\, 2^{1-k} |C_k(y)|$. (10.26)

This theorem shows that the **monic Chebyshev polynomial $2^{1-k} C_k(x)$ has the steepest slope outside [-1,1]** w.r.t. all monic polynomials of degree k. That is also the reason why **kernel polynomials** [106] defined on [-1,1] will **not converge so fast** as Chebyshev polynomials outside [-1,1]. Hence an iterative scheme based on kernel polynomials will converge more slowly than the Chebyshev iteration method. This is easily proven.

In order to get the analogous results for an arbitrary interval [y,z] we can use the **adapted Chebyshev polynomials** with similar properties over [y,z] :

$$C_k^{yz}(x) = C_k((2x - y - z) / (z - y)) = C_k(1 + 2(x - z) / (z - y)) \tag{10.27}$$

Observe that now every $x \in [y,z]$ is projected onto a $t \in [-1,1]$ by the linear transformation function f, given by :

$$f(x) = (2x - y - z) / (z - y) = t \tag{10.28}$$

satisfying

$$f(y) = -1; \quad f(z) = 1 \quad \text{and} \quad C_k^{yz}(x) = C_k(f(x)) = C_k(t) \tag{10.29}$$

10.3.2. THE CHEBYSHEV ITERATION ALGORITHM

The application of Chebyshev polynomials to accelerate iterative eigenvalue algorithms is certainly not new, see e.g. [82-93-134-24]. **By choosing [y,z] so that it contains all eigenvalues of a matrix S except one, the Chebyshev iteration method will converge to the eigenvector associated with the eigenvalue outside [y,z].** W.r.t. the computation of the smallest singular triplet of a matrix A, e.g. in one-dimensional generic TLS problems, this means that [y,z] must contain all eigenvalues of the squared matrix $S = A^TA$ except one. If (9.1) is the SVD of A, this implies that :

$$z \geq \sigma_1^2 \quad \text{and} \quad \sigma_n^2 < y \leq \sigma_{n-1}^2 \tag{10.30}$$

By applying the adapted Chebyshev polynomials (10.27) satisfying (10.30) onto the matrix $S = A^TA$, one obtains the sequence of matrices :

$$C_k^{yz}(S) = C_k(f(S)) = C_k((2A^TA - (y+z) I) / (z-y)) \tag{10.31}$$

$$= V[(2\Sigma^T\Sigma - (y+z) I) / (z-y)]V^T = VDV^T \tag{10.32}$$

satisfying the conditions (10.19) and (10.21). Indeed, the eigenvalues are :

$$|d_i| = |(2\sigma_i^2 - y - z) / (z - y)| \leq 1 \quad \text{for } i = 1, \ldots, n-1$$

and

$$|d_n| = |(2\sigma_n^2 - y - z) / (z - y)| > 1 \tag{10.33}$$

Only the singular values (eigenvalues of S) are changed but the singular vectors are conserved. This means that the following **Chebyshev iteration algorithm** will **converge to the smallest singular triplet** of an m by n matrix A, hence allow to compute the unique solution of a one-dimensional generic TLS problem (0.1) if A is replaced by the m by n+1 matrix [A;b] :

Algorithm 10-1 : Chebyshev iteration to compute the right singular vector of the smallest singular triplet

Given : - an m by n matrix A, $m \geq n$
- bounds y, z satisfying (10.30)
- the desired numerical accuracy ε
- a start vector q_1

$q_2 \leftarrow [2\, A^T (A\, q_1) - (y + z)\, q_1] \,/\, (z - y)$

For k = 2, 3, ...

$$q_{k+1} \leftarrow [4\, A^T (A\, q_k) - (y + z)\, q_k] \,/\, (z - y) - q_{k-1}$$

$$\text{if } \| q_{k+1} \,/\, \|q_{k+1}\|_2 - q_k \,/\, \|q_k\|_2 \|_2 < \varepsilon, \text{ stop}$$

END

Some comments are in order :

- The number of operations (multiplications and divisions) is :

$$n\,(2\,m + 5)\,K_c \quad \text{with} \quad K_c \text{ the number of required iteration steps} \tag{10.33}$$

- Observe that the singular vectors are only defined up to a scaling factor. In order to perform the convergence test correctly we have to take care that the subsequent iteration vectors are equally signed, e.g. by scaling each iteration vector such that its first component has positive sign.
- The matrix-vector product $A^T A\, q_k$ is computed as $A^T (A\, q_k)$ which is numerically more accurate than $(A^T A) q_k$.
- The matrix A is not changed during the computations of algorithm 10-1.
- **By computing the (n+1)-th right singular vector v_{n+1} of [A;b] with algorithm 10-1 and scaling it as in (1.16), the solution of the generic one-dimensional TLS problem (0.1) is obtained.**

10.3.3. CONVERGENCE PROPERTIES AND APPLICABILITY

Under the assumption (10.30), numerical convergence of $t = -\log \varepsilon$ digits will be reached if the last internal coordinate in (10.6) dominates the other by t decimal digits. Hence from (10.8) numerical convergence occurs in K iteration steps if :

$$\log \left(|C_K^{yz}(\sigma_n^2)\, v_n^T q_1| \,/\, \max_{j=1,n-1} |C_K^{yz}(\sigma_j^2)\, v_j^T q_1| \right) = \log \varepsilon^{-1} = t \tag{10.34}$$

Since $|C_K^{yz}(\sigma_j^2)| \leq 1 \;\forall j = 1, n-1$ from (10.29) and the assumptions (10.30), this becomes for the worst case :

$$\log \left(C_K^{yz}(\sigma_n^2) |\, v_n^T q_1| \,/\, \max_{j=1,n-1} |v_j^T q_1| \right) = -\log \varepsilon = t \tag{10.35}$$

This equation determines the number K of iteration steps necessary for numerical convergence of $t = -\log \varepsilon$ decimal digits. For sufficiently large K, $C_K^{yz}(\sigma_n^2) = C_K(d_n)$ with $d_n = f(\sigma_n^2)$ (see (10.28)) can be approximated from (10.17) by :

$$C_K(d_n) \approx \frac{1}{2} \left(d_n + \sqrt{d_n^2 - 1} \right)^K \tag{10.36}$$

Substituting this into (10.35) or from (10.11), one can estimate **the number K of required iteration steps for convergence of the Chebyshev iteration algorithm** 10-1 :

$$K \approx \frac{\log (2\, \varepsilon^{-1}) + \log \left(\max_{j=1,n-1} |v_j^T q_1| \,/\, |v_n^T q_1| \right)}{\log (d_n + \sqrt{d_n^2 - 1})} \tag{10.37}$$

or from (10.12) :

$$K \approx \left(\log (2\, \varepsilon^{-1}) + \log |\tan \varphi_n| \right) / \log (d_n + \sqrt{d_n^2 - 1}) \tag{10.38}$$

with $d_n = 1 + 2\rho_n = 1 + 2\,(y - \sigma_n^2)/(z - y)$ and y,z satisfying (10.30) and

$$\varphi_n = \cos^{-1} |v_n^T q_1|, \; q_1 \text{ start vector}, \; \|q_1\|_2 = 1 \tag{10.39}$$

From the properties of the Chebyshev polynomials, convergence is optimal if $|d_n - 1|$ is as large as possible. It is not difficult to prove that this corresponds to optimal choices of y and z as :

$$y = \sigma_{n-1}^2 \quad \text{and} \quad z = \sigma_1^2 \tag{10.40}$$

As discussed in section 10.2, the number K of required iteration steps decreases if the desired accuracy ε is low, the quality of the start vector q_1 good and the convergence rate $\log (d_n + \sqrt{d_n^2 - 1})$ high. However, observe from (10.39) that **the convergence rate not only depends on the gap between σ_{n-1}^2 and σ_n^2 but also on the relative spread of the squared singular value spectrum** $(\sigma_1^2 - \sigma_{n-1}^2)/\sigma_{n-1}^2$. It is mainly **this last parameter** which **slows down the convergence of the Chebyshev iteration algorithm in case of badly conditioned matrices A**, even if the gap between σ_{n-1} and σ_n is quite large. For example, if $\tan \varphi_n = 10^{-2}$ and $\varepsilon = 10^{-7}$ is required, then the number K of iteration steps given by (10.38) and using the optimal bounds (10.40) for y and z is :

K = 11 if $\sigma_1 = 2$, $\sigma_{n-1} = 1$, $\sigma_n = .1$

K = 61 if $\sigma_1 = 10$, $\sigma_{n-1} = 1$, $\sigma_n = .1$

K = 613 if $\sigma_1 = 100$, $\sigma_{n-1} = 1$, $\sigma_n = .1$

K = 6132 if $\sigma_1 = 1000$, $\sigma_{n-1} = 1$, $\sigma_n = .1$

Moreover, if one wants to compute subsequent singular triplets, e.g. the singular vectors associated to singular values $< \theta$, **the same Chebyshev iteration algorithm 10-1 can be used for computing the singular triplets consecutively**, if we prevent the algorithm to re-compute the results already found. This is done by using deflation techniques or by orthogonalizing the iteration vector to the singular vectors already found in each step [82]. Also, extension of the Chebyshev iteration algorithm to the subspace iteration case is straightforward. Indeed, by using an n by r start matrix Q_1 with orthogonal columns, algorithm 10-1 can also converge to a basis of the right singular subspace associated with the r smallest singular values of a matrix A. Therefore, we must keep the columns of the iteration matrices Q_k orthogonal in each iteration step, e.g. by using Householder transformations [48, section 6.2]. Of course, all these extensions increase the computational complexity of the Chebyshev iteration algorithm 10-1 and make it less efficient.

W.r.t. the TLS problem (0.1) with SVD given in (0.3), one can conclude that the Chebyshev iteration method will be efficient if :

- the initial guess vector is very good
- the desired accuracy rather low
- the rank of [A;B] known
- the bounds y,z can be very well estimated i.e. $y \approx \sigma_1^2$ and $z \approx \sigma_n^2$
- the problem is highly structured or sparse
- $m \approx n$
- m, n large (> 100)
- the gap between desired and undesired singular values is large i.e. $(\sigma_n^2 >> \sigma_{n+1}^2$
- **the relative spread of the squared singular value spectrum** given by $(\sigma_1^2 - \sigma_n^2)/\sigma_n^2$ **must be small.**

Especially **the last requirement is for most TLS problems not satisfied and makes the Chebyshev iteration algorithm less efficient than the classical TLS algorithm as soon as the relative spread exceeds 100 for moderate m,n.** This has been experimentally confirmed and is illustrated in the following example.

Example 10-1 :

Consider the set equations (0.1) with A a 10 by 5 matrix and one observation vector b (d = 1). The singular values of [A;b] are (1, .9, .8, .721, .7, 10^{-15}). We perturb the set with noise of zero mean and variance .0006 and take the solution of the previous set as initial guess for the next perturbed set. The upper bound z is set to $\|[A;b]\|_F^2$ and is approximately 3.5. Different values for the lower bound y are tried. The following average CPU times on the IBM 3030 of the computer center at the K.U.Leuven are obtained.

Chebyshev iteration algorithm :

31.2 ms (49 iterations) if y = .5 and $\varepsilon = 10^{-12}$

6.25 ms (7 iterations) if y = .5 and $\varepsilon = 1.7 \times 10^{-3}$

6.25 ms (7 iterations) if y = .4 and $\varepsilon = 1.7 \times 10^{-3}$

6.25 ms (7 iterations) if y = .3 and $\varepsilon = 1.7 \times 10^{-3}$

9 ms (11 iterations) if y = .1 and $\varepsilon = 1.7 \times 10^{-3}$

15.7 ms (19 iterations) if y = .012 and $\varepsilon = 1.7 \times 10^{-3}$

classical TLS algorithm :

11 ms if $\varepsilon = 10^{-3}$ or 10^{-12}

partial TLS algorithm :

6 ms if $\varepsilon = 10^{-3}$ or 10^{-12}

If we take $\sigma_1 = 10$ instead of 1 in the example, then we obtain :

Chebyshev iteration algorithm :

20 ms (24 iterations) if $y = .5$, $z = 110$ and $\varepsilon = 1.7 \times 10^{-3}$

classical TLS algorithm : 10 ms.

10.4. INVERSE ITERATION

10.4.1. THE INVERSE ITERATION ALGORITHM

The idea of inverse iteration is already quite old. Many authors studied this iterative method in (non)symmetric eigenvalue problems. See [82] and [48, section 7.6] for references. According to Wilkinson [134], the inverse iteration method is the most powerful and accurate method for computing eigenvectors.

By **applying the inverse power functions** $f_k(\lambda) = (\lambda - \lambda_o)^{-k}$ onto the n by n matrix $S = A^TA$ with A given by (9.1), one obtains the inverse of $S - \lambda_o I$ and its powers :

$$f_k(S) = (S - \lambda_o I)^{-k} = V(\Sigma^T\Sigma - \lambda_o I)^{-k} V^T = VDV^T \qquad (10.41)$$

with eigenvalues $d_j = f_k(\sigma_j^2) = (\sigma_j^2 - \lambda_o)^{-k}$ for $j = 1, \ldots, n$. Clearly, as the functions $(\lambda - \lambda_o)^{-k}$ are hyperbolic with a vertical asymptote in λ_o, **the function value $f_k(\lambda)$ of the eigenvalue λ_i which minimizes $|\lambda_i - \lambda_o|$ will dominate the other.** Hence, the iteration vector q_k :

$$q_k = f_k(S)\, q_1 = (S - \lambda_o I)^{-1} q_{k-1} = (S - \lambda_o I)^{-k} q_1 \qquad (10.42)$$

with q_1 a start vector, becomes increasingly strong in the direction of the eigenvector v_i and its internal coordinates (10.6) will converge to (10.7) for $k \to \infty$, as illustrated geometrically in fig. 10.1. Observe from (10.41) that again the singular values of the matrix functions $f_k(S)$ are changed but the singular vectors are conserved.

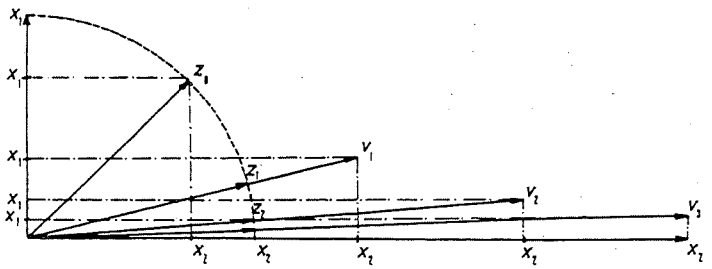


Fig. 10.1 : Geometric illustration of the inverse iteration process, applied to a 2 by 2 matrix with eigenvalues $\lambda_1 = 2$ and $\lambda_2 = 0.5$ and associated eigenvectors x_1 and x_2. The shift approaching λ_2, is set to zero and the start vector z_o is given by $(x_1 + x_2)/\sqrt{2}$. The figure shows the three first iteration vectors v_1, v_2 and v_3 and their corresponding unit vectors z_1, z_2 and z_3, converging to the eigenvector x_2.

Of course, **we will not compute** the subsequent iteration vectors q_k by computing **the inverse** in (10.42) **but instead solve a set of linear equations** of the form :

$$(S - \lambda_o I)\, q_k = q_{k-1} \qquad (10.43)$$

in each iteration step. The usual way to solve (10.43) is by first factorizing the matrix $S - \lambda_o I$. This factorization is then used to solve the simultaneous linear equations. We can take advantage of the symmetry of $S - \lambda_o I$ to choose one of the well-known **factorizations of symmetric matrices.** The eigenvalue decomposition (10.1) is too costly and has no sense in an inverse iteration algorithm. Since $S - \lambda_o I$ can be semi-definite or negative definite depending on the chosen shift λ_o, the symmetric triangular factorization $A = UDU^T$ with U unit upper triangular and D diagonal [134] and also the Cholesky factorization $A = R^TR$ with R upper triangular [48, section 5.2] may not exist or be unstable [29, chapter 5]. The triangular factorization $A = UDU^T$ with U a product of elementary unit upper triangular and permutation matrices and T symmetric and tridiagonal [1] is stable but more costly than the stable **diagonal pivoting factorization** $A = UDU^T$, where U is a product of ele-

mentary unit upper triangular and permutation matrices and D is symmetric block diagonal with blocks of order 1 or 2 [16-17-18]. This factorization is a generalization of Lagrange's method for reducing quadratic forms to diagonal form and corresponds to block symmetric Gaussian elimination with partial pivoting. It **is the best choice for solving the set of equations (10.43) in the inverse iteration algorithm.** A comparison of various algorithms for solving symmetric indefinite systems of equations can be found in [11]. The diagonal pivoting factorization of $S - \lambda_o I$ requires at most $n^3/6 + 3n^2/4 + 7n/3$ multiplications (and divisions). Since the matrix $S - \lambda_o I$ is left unchanged by inverse iteration, this factorization must be calculated once. Given the factorization $S - \lambda_o I = UDU^T$ we then solve (10.43) in each iteration step by computing $DU^T q_k = U^{-1} q_{k-1}$. This requires $n^2 + 2n$ multiplications (and divisions). Using this method, the following **inverse iteration algorithm** will **converge to the singular vector** of a matrix A, **associated with its singular value which minimizes** $|\sigma_i^2 - \lambda_o|$:

Algorithm 10-2 : Inverse iteration based on the diagonal pivoting factorization

Given : - an m by n matrix A, $m \geq n$
- a shift λ_o
- the desired numerical accuracy ε
- a start vector q_1 with $\|q_1\|_2 = 1$

$S \leftarrow A^T A - \lambda_o I$

$UDU^T = S$ (diagonal pivoting factorization)

for k = 1, 2, ...

solve $DU^T q_{k+1} = U^{-1} q_k$

$q_{k+1} \leftarrow q_{k+1} / \|q_{k+1}\|_2$

if $\|q_{k+1} - q_k\|_2 < \varepsilon$, stop

END

Some comments are in order :

- A **geometric** representation of the inverse iteration process is given in fig. 10.1.

- In order to perform the convergence test correctly, the iteration vectors q_k are scaled such that their first component has positive sign (see also algorithm 10-1).

- The total number of operations of the inverse iteration algorithm 10-2 is

$$mn\ (n+1)\ /\ 2 + n^3/6 + 3n^2/4 + 7n/3 + K_I\ (n^2 + 5n) \qquad (10.44)$$

with K_I the number of required iteration steps for convergence.

- For the computation of the smallest singular triplet of a matrix A whose SVD is given by (9.1), e.g. in one-dimensional generic TLS problems, we take **shift** λ_o zero or a better estimate, if available, satisfying :

$$|\sigma_n^2 - \lambda_o| < \sigma_n^2 \quad \text{and} \quad |\sigma_n^2 - \lambda_o| < |\sigma_{n-1}^2 - \lambda_o| \qquad (10.45)$$

- Observe that (10.43) is very **ill-conditioned** if the shift λ_o is very close to some eigenvalue σ_i^2 of S. However, this fact **does not affect the accuracy** of the solution, as proven by Peters and Wilkinson [84] and Parlett [82] and as experimentally verified in [2].

- Notice also that inverse iteration destroys the structure of the matrix A by factorizing $A^TA - \lambda_o I$.

- **Instead of factorizing** $S - \lambda_o I$, **we could first triangularize it using Householder transformations** Q, i.e.

$$T = Q^T\ (S - \lambda_o I)\ Q \qquad (10.46)$$

Since T and $S - \lambda_o I$ have the same eigenvalues we could compute the eigenvector s_i of the tridiagonal T by inverse iteration and finally obtain the eigenvector v_i of the original matrix $S - \lambda_o I$ by back transformation, i.e.

$$v_i = Q\ s_i \qquad (10.47)$$

In order to compute the eigenvector of T, a tridiagonal set of equations must be solved in each step. Gaussian elimination with partial pivoting [48, section 5.3] is the best choice w.r.t. stability and efficiency [11]. This inverse iteration algorithm is used in the subroutine TINVIT of the EISPACK library [100]. Although the computation of the eigenvector s_i of the tridiagonal requires minimal cost, the necessary transformations are costly and make this algorithm less efficient for solving

TLS problems. Indeed, the EISPACK algorithm based on tridiagonalizing $S - \lambda_o I$ requires $mn(n+1)/2 + 2n^3/3 + 5n^2/2 + 17n/6 - 16 + K(7n-4)$ operation counts. **It is less efficient than the inverse iteration algorithm 10-2, if the number K of iteration steps** :

$$K < (n^3/2 + 7n^2/4 + n/2 - 16) / (n-2)^2 \qquad (10.48)$$

For example, $K < 10$ for $n = 10$, $K < 14.5$ for $n = 20$ and $K < 19.2$ for $n = 30$. For large n, we have $K < n/2$. Hence, algorithm 10-2 will always be more efficient **if n is small or the convergence rate is good so that $K < n/2$**. This is generally the case for TLS problems. Therefore, algorithm 10-2 based on a diagonal pivoting factorization is the best choice for solving TLS problems w.r.t. efficiency and stability.

10.4.2. CONVERGENCE PROPERTIES AND APPLICABILITY

Substituting (10.41) into (10.8), it follows that the inverse iteration algorithm 10-2 converges up to a precision ε in K iteration steps to the n-th right singular vector of A, given by (9.1), if :

$$\max_{j=1,n-1} |(\sigma_j^2 - \lambda_o)^{-K} v_j^T q_1| \; / \; |(\sigma_n^2 - \lambda_o)^{-K} v_n^T q_1| < \varepsilon \qquad (10.49)$$

$(\sigma_j^2 - \lambda_o)^{-1}$ is maximal for $j = n-1$ if $j \neq n$. From (10.11), we can estimate the number K of iteration steps necessary for numerical convergence of the inverse iteration algorithm 10-2, up to $t = -\log \varepsilon$ decimal digits :

$$K \approx \frac{\log \varepsilon^{-1} + \log \left(\max_{j=1,n-1} |v_j^T q_1| \; / \; |v_n^T q_1|\right)}{\log \left(|\sigma_{n-1}^2 - \lambda_o| \; / \; |\sigma_n^2 - \lambda_o|\right)} \qquad (10.50)$$

or from (10.12) :

$$K \approx (\log \varepsilon^{-1} + \log |\tan \varphi_n|) \; / \; \log \left(|\sigma_{n-1}^2 - \lambda_o| \; / \; |\sigma_n^2 - \lambda_o|\right) \qquad (10.51)$$

with $\varphi_n = \cos^{-1} |v_n^T q_1|$, q_1 start vector and $\|q_1\|_2 = 1$.

As discussed in section 10.2, the number K of required iteration steps decreases if the desired accuracy ε is low, the start vector is a good estimate of v_n and the convergence rate depending on the ratio :

$$|\sigma_{n-1}^2 - \lambda_o| \; / \; |\sigma_n^2 - \lambda_o| \qquad (10.52)$$

is high. The effect of those parameters is illustrated in [2]. Even if the start vector q_1 is orthogonal onto the desired singular vector v_n of A, convergence is slown down but still occurs because roundoff errors always induce a component of q_1 in the direction of v_n. From (10.49) it follows that **fast convergence occurs if the shift λ_o approaches σ_n^2 and the gap between σ_{n-1} and σ_n is large, as in the case for most TLS problems**. From (10.41) and (10.6), it is clear that the reduction of the component in the direction of v_{n-1} w.r.t. the component in the direction of v_n is proportional with (10.52). The higher this ratio, the faster the convergence. Contrary to Chebyshev iteration **the inverse iteration convergence does not depend on the relative spread of the squared singular value spectrum !** This is clear by considering the same examples as in section 10.3.2. If $\tan \varphi_n = 10^{-2}$ and $\varepsilon = 10^{-7}$ is required, then the number K of iteration steps in the inverse iteration algorithm 10-2 given by (10.51) is :

$$K = 2.5 \quad \text{for} \quad \sigma_{n-1} = 1 \quad \text{and} \quad \sigma_n = .1$$

(independently from the value of σ_1).

The ratio (10.52) equals one if $|\sigma_{n-1}^2 - \lambda_o| = |\sigma_n^2 - \lambda_o|$. This happens if the singular values σ_{n-1} and σ_n coincide : convergence then occurs to a singular vector of the associated right singular subspace. The ratio (10.52) also equals one if the shift is precisely the average $(\sigma_{n-1}^2 + \sigma_n^2)/2$ of two distinct singular values. In this case, the iteration process does not converge to one singular vector but to a vector having components in the direction of v_{n-1} and v_n. If (10.52) is slightly larger than one, convergence occurs but is slow. Observe however that a shift λ_o making (10.52) smaller than one, forces the iteration process to converge to the singular vector v_{n-1} instead of v_n, showing clearly the effects of a bad choice of the shift λ_o.

Just like the Chebyshev iteration algorithm, **algorithm 10-2 can be used to compute subsequent singular triplets** if we prevent the algorithm to recompute the results already found. This is done by using deflation techniques or orthogonalizing the iteration vector in each step to the singular vectors already found. Also, extensions to the subspace iter-

ation case are straightforward. In this case, the method is called **simultaneous (or orthogonal) inverse iteration** [92-48, section 8.5] and a basis of the singular subspace associated with the d smallest eigenvalues (in absolute value) of $S - \lambda_o I$, is computed by the following algorithm :

Algorithm 10-3 : Simultaneous inverse iteration based on a diagonal pivoting factorization

Given : - an m by n matrix A, $m \geq n$
- the dimension d of the desired right singular subspace of A
- a shift λ_o
- the desired numerical accuracy ε
- an n by d orthonormal start matrix Q_1

$S \leftarrow A^TA - \lambda_o I$
$UDU^T = S$ (diagonal pivoting factorization)
For k = 1, 2, ...
solve $DU^T Z = U^{-1} Q_k$
$Q_{k+1} R = Z$ (QR factorization)
if $\|Q_{k+1} - Q_k\|_2 < \varepsilon$, stop
END

This algorithm requires

$$mn(n+1)/2 + n^3/6 + 3n^2/4 + 7n/3$$
$$+ K_{SI}[n^2d + (2d^2 + 3d)n - 2d^3/3 + 2d^2 - d/3 - 4] \qquad (10.53)$$

operations. K_{SI} is the number of required iteration steps and d the dimension of the subspace.
Simultaneous inverse iteration is mainly used to converge to a subspace associated with clustered eigenvalues, e.g. multidimensional TLS problems. To estimate the number of required iteration steps, formula (10.51) can be used

$$K_{SI} \approx (\log \varepsilon^{-1} + \log |\tan \varphi_n|) / \log (|\sigma^2_{n-d} - \lambda_o| / |\sigma^2_{n-d+1} - \lambda_o|) \qquad (10.54)$$

φ_n is the maximal angle between Q_1 and $V_n = [v_{n-d+1}, \ldots, v_n]$, defined by $\sin \varphi_n = \text{dist}(Q_1, V_n)$ (see section 2.5).

W.r.t. the TLS problem (0.1) and SVD given in (0.3), one can conclude that the inverse iteration method will be efficient if :
- the initial guess vector is good
- the desired accuracy low
- the rank of [A;B] known
- moderate m,n
- the shift λ_o approaches σ^2_{n+1}
- the ratio $|\sigma^2_n - \lambda_o| \; / \; |\sigma^2_{n+1} - \lambda_o|$ is large.

Especially the last requirement is important and is satisfied for almost all TLS problems. For example, if σ_n/σ_{n+1} has order $O(10^k)$, the convergence rate is of order $O(10^{2k})$ and may be very fast even if the other conditions mentioned above are not satisfied. Hence, inverse iteration will be very efficient in solving generic one-dimensional problems with unique solution, as illustrated in section 10.7 and 10.8.

10.5. RAYLEIGH QUOTIENT ITERATION

10.5.1. THE RAYLEIGH QUOTIENT ITERATION ALGORITHM

By varying the shift $\lambda_o(k)$ in each step of the inverse iteration in a well-defined way, **we obtain Rayleigh quotient (RQ) iteration** [48, section 8.5]. The RQ iteration vectors q_k given by (10.41), are obtained by applying onto the matrix $S = A^TA$ the inverse power functions $f_k(\lambda) = (\lambda - \lambda_o(k))^{-k}$ with variable shift $\lambda_o(k)$, causing a well-defined motion of the vertical asymptote in each iteration step. The shift $\lambda_o(k)$ in each iteration step is the Rayleigh quotient (RQ) $r(q_k)$ of the vector q_k, defined by :

$$r(q_k) = q_k^T S q_k \; / \; q_k^T q_k \qquad (10.55)$$

minimizing $f(\lambda) = \|(S - \lambda I) q_k\|_2 \qquad (10.56)$

Clearly, as q_k is an approximate eigenvector, $\lambda_o(k)$ is a reasonable choice for the corresponding eigenvalue. On the other hand, if $\lambda_o(k)$ is an approximate eigenvalue, then inverse iteration tells us that the solution to :

$$(S - \lambda_o(k))\, q_k = q_{k-1} \qquad (10.57)$$

is almost always a good approximate eigenvector. Combining these two ideas in the natural way give rise to the RQ iteration.

In order to solve the set of linear equations (10.57), observe that the matrix $S - \lambda_o(k)\, I$ changes in every step and hence, its factorization has to be updated (contrary to inverse iteration with fixed shift). If S is tridiagonal, this updating operation requires O(n) multiplications. However, if S is not tridiagonal the updating is more difficult or even impossible since the transformation matrices in the factorization are not orthogonal, and require $O(n^2)$ to $O(n^3)$ multiplications. Therefore in case of RQ iteration, **it is strongly recommended to reduce $S = A^T A$ first to its tridiagonal form T via orthogonal (Householder) transformations Q** (an operation which preserves the eigenvalues). We then compute the eigenvector s_i of the tridiagonal T associated to the desired eigenvalue λ_i and finally obtain the eigenvector v_i of S, or equivalently the singular vector of A, by the back transformation (10.47). The set of equations in each step of the RQ iteration with T tridiagonal is solved by **Gauss elimination with partial pivoting**, a stable technique with minimal computational cost [11]. As said in section 10.4.1, this strategy is also used in the inverse iteration algorithm of the EISPACK library [100].

The **RQ iteration** is summarized in the following **algorithm** :

Algorithm 10-4 : Rayleigh quotient iteration with preceding orthogonal tridiagonalization

Given : - an m by n matrix A, $m \geq n$
- the desired numerical accuracy ε
- a start vector q_1, $\|q_1\|_2 = 1$

$S \leftarrow A^T A$

$T = Q^T S Q$ (Householder tridiagonalization)

For k = 1, 2, ...

$\lambda_o(k) = q_k^T\, T\, q_k$

$LU = T - \lambda_o(k)\, I$

Solve $LU\, q_{k+1} = q_k$ } Gauss elimination with partial pivoting

$q_{k+1} \leftarrow q_{k+1} / \|q_{k+1}\|_2$

if $\|q_{k+1} - q_k\|_2 < \varepsilon$, then begin

$v \leftarrow Q\, q_{k+1}$ (back transformation)

stop

end

END

Some comments are in order :

- The total number of operations is given by :

$$mn(n+1)/2 + 2n^3/3 + 5n^2/2 - n/6 - 13 + K_{RQ}\,(14n-9) \qquad (10.58)$$

with K_{RQ} the number of required iteration steps.

- In order to perform the convergence test correctly, the iteration vectors q_k are scaled such that their first component has positive sign (see also algorithm 10-1).

- Observe that the **start vector** must be an **initial guess of the eigenvector of the tridiagonal T** associated with the desired eigenvalue. However, we can also start from an initial guess q_A of the singular vector of the original matrix A if we transform q_A to $q_1 = Q^T q_A$ after the tridiagonalization before starting the iteration loop.

- Moreover, **the start vector must be correctly chosen in order to converge to the right singular vector of A.** Therefore the component of q_1 in the direction of the desired singular vector must be largest.

- **Performing the RQ iteration by computing the diagonal pivoting factorization** of $S - \lambda_o(k) I$ for solving (10.57) in each iteration step requires $mn(n+1)/2 + K(n^3/6 + 11n^2/4 + 22n/3)$ operation counts. From this and (10.58), one easily computes that algorithm 10-4 will be more efficient if the number K of RQ iteration steps satisfies :

$$K > (4n^3 + 15n^2 - n - 78) \,/\, (n^3 + 16.5n^2 - 40n + 54) \qquad (10.59)$$

For example, $K > 2.35$ if $n = 10$ and $K > 2.74$ if $n = 20$. For large n, $K > (4n + 15) / (n + 16.5)$. Hence, the algorithm 10-4 is nearly always the most efficient choice except in cases of very good initial guesses or very fast convergence.

10.5.2. CONVERGENCE PROPERTIES AND APPLICABILITY

Substituting (10.41) into (10.9) it follows that the RQ iteration algorithm 10-4 converges up to a precision ε in K iteration steps to the n-th right singular vector of A given by (9.1), if :

$$\max_{j=1,n-1} |(\sigma_j^2 - \lambda_o(K))^{-K} v_j^T q_1| \; / \; |(\sigma_n^2 - \lambda_o(K))^{-K} v_n^T q_1| < \varepsilon \qquad (10.60)$$

with q_1 an initial guess of the desired singular vector of A. However, since the shift $\lambda_o(k)$ varies from step to step, we can not use the convergence formula derived in section 10.2, to estimate the number K of required iteration steps. Parlett [81] has shown that the RQ iteration algorithm converges globally and that **the convergence is ultimately cubic**, i.e. after a few iteration steps. This is demonstrated for the case n = 2 by Golub and Van Loan [48, section 8.5].

Since RQ is a variant of inverse iteration, the same parameters influence the convergence properties. Also, similar extensions to subspace iteration and computing various singular vectors are possible. Therefore, we refer the reader to section 10.4.2 for a discussion of the convergence properties and the extensions and restrict ourselves here to some additional comments, proper to RQ iteration.

As **RQ iteration** chooses the best shift in each step, it **converges faster than inverse iteration but at a higher computation cost for each iteration step** (see also section 10.7 for a detailed comparison in operation counts). However, as said in section 10.5.1, we have **to take care that the start vector is always correctly chosen in order to converge to the desired singular vector of A !** This is very important for the TLS problem in order to ensure a correct solution. For example in adaptive TLS problems, varying sets of equations have to be solved at each time instant. Most often, the solution of the set equations at a previous time instant is a **good initial guess vector** for solving the set at the next time instant, but this **can not be guaranteed.** Indeed, most adaptive problems are characterized by abrupt changes at certain time instants which affect seriously the quality of the initial guess and cause RQ iteration to converge to an undesired singular triplet instead of the smallest one. Hence, the TLS solution may be **incorrectly** computed. On

the other hand, if we use inverse iteration with zero shift, convergence to the smallest singular triplet is always ensured ! Of course, we can **combine both algorithms** 10-2 and 10-4 to ensure a correct solution at an **optimal convergence** rate. No general rules can be set up but we must work out an optimal strategy for each particular application.

W.r.t. TLS problems (0.1) and SVD given in (0.3), we can conclude that **RQ iteration will be more efficient than inverse iteration if :**

- a good quality of the initial guess vector is guaranteed in order to converge to the desired (n+1)-th right singular vector of [A;b]
- the ratio $|\sigma_n^2 - \lambda_o| \; / \; |\sigma_{n+1}^2 - \lambda_o|$ is not high, slowing down the convergence rate of the inverse iteration process.

The latter condition happens if the gap between the desired (σ_{n+1}) and undesired (σ_n) singular values of [A;B] is not large and no adequate shift λ_o close to σ_{n+1}^2, can be computed. This is the case **if the set of equations is highly incompatible.** Since in those situations the data are not appropriate for linear modelling and TLS looses its optimal properties (see chapter 4), those problems are rather irrelevant. We can thus conclude that for most adaptive generic one-dimensional TLS problems, the inverse iteration technique is the most adequate iteration process (see section 10.7 and 10.8).

10.6. LANCZOS METHODS

10.6.1. LANCZOS ALGORITHMS

The Lanczos method is an iterative method which bidiagonalizes an arbitrary matrix A or tridiagonalizes a symmetric matrix S directly without any orthogonal updates of the matrix as in the Householder approach (see section 9.2). **The Lanczos bidiagonalization (resp., tridiagonalization) method generates a sequence of bidiagonal (resp., tridiagonal) matrices with the property that its extremal singular values are progressively better estimates of the extremal singular values of the given matrix.** Therefore, the Lanczos methods may be computationally more attractive than Householder bidiagonalization in the computation of the SVD of a matrix. Indeed, long before the bidiagonalization is complete, the largest and/or smallest singular values of the given matrix can already

be accurately obtained. From the associated eigenvectors s_j of the bidiagonal, the singular vectors v_j of the given matrix are derived via back transformation as in (10.47). That's why we compare the Lanczos methods with the classical Householder approach in bidiagonalizing a matrix A or tridiagonalizing the corresponding matrix A^TA, w.r.t. their efficiency in solving TLS problems. In this section we derive the technique and investigate its arithmetic properties.

Let's start with the **Lanczos tridiagonalization**, since it is the basis technique from which later on the Lanczos bidiagonalization and other variations were derived. The derivation of its algorithm can proceed in several ways, e.g. by considering the optimization of the RQ r(x) given by (10.55). See [48, section 9.1] for more details. The elements of the tridiagonal matrix T are computed directly as follows. Let $Q = [q_1, \ldots, q_n]$ be an orthonormal matrix and

$$T_j = \begin{bmatrix} \alpha_1 & \beta_1 & & 0 \\ \beta_1 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_{j-1} \\ 0 & & \beta_{j-1} & \alpha_j \end{bmatrix} \qquad (10.61)$$

and equate columns in $SQ = QT_n$. Then we find :

$$S q_j = \beta_{j-1} q_{j-1} + \alpha_j q_j + \beta_j q_{j+1} \qquad (10.62)$$

for j = 1, ..., n-1. The orthonormality of the q_i implies $\alpha_j = q_j^T Sq_j$. Moreover if

$$r_j = (A - \alpha_j I) q_j - \beta_{j-1} q_{j-1} \qquad (10.63)$$

is nonzero, then $q_{j+1} = r_j/\beta_j$ where $\beta_j = \pm \|r_j\|_2$. Properly sequenced, these formulas define the Lanczos iteration. There is no loss of generality in choosing β_j to be positive. The q_j are called Lanczos vectors. Using the formula $\alpha_j = q_j^T (Sq_j - \beta_{j-1} q_{j-1})$, the whole process can be summarized into the following **algorithm, computing a j by j symmetric tridiagonal matrix T_j with the property that its eigenvalues converge to those of S** (see section 10.6.2) :

Algorithm 10-5 : Lanczos tridiagonalization

Given : - a symmetric n by n matrix S
- a start vector $q_1 \in \mathbb{R}^n$, $\|q_1\|_2 = 1$

$\beta_o = 0$
For j = 1, 2, ...
$u_j = Sq_j - q_{j-1} \beta_{j-1}$
$\alpha_j = q_j^T u_j$
$r_j = u_j - q_j \alpha_j$
$\beta_j = \|r_j\|_2$
if $\beta_j = 0$ then stop
else $q_{j+1} = r_j/\beta_j$
END

Some comments are in order :

- With careful overwriting the process can be implemented with just a pair of n-dimensional vectors [48, algorithm 9.1-1].
- Observe that S is not altered during the entire process. Thus, only a procedure for computing matrix-vector products involving S need be supplied.
- Using the relation (10.4) between the singular values of a rectangular matrix A and the eigenvalues of A^TA, we can use algorithm 10-5 to tridiagonalize $S = A^TA$. **Replacing S by A^TA in algorithm 10-5 and computing Sq_j as $A^T(Aq_j)$, the total number of operations** is

$$K_L (2m + 5)n - n \qquad (10.62)$$

with K_L the number of required Lanczos steps.

- After the tridiagonalization of $S = A^TA$ to T by the Lanczos algorithm 10-5, we must compute the eigenvectors s_j of T associated with the desired eigenvalues of T (and S), e.g. by inverse iteration (see section 10.5.1) or by the symmetric QR algorithm [48, section 8.2] (as said in section 9.3.4, this algorithm can be made more efficient in the same way as PSVD). The desired eigenvectors v_j of S are then obtained by performing the appropriate back transformation (10.47). Generally, **the additional computational cost to compute the eigenvalues or eigenvectors is negligible w.r.t. the tridiagonalization for large n.**

- Observe that the tridiagonals T_j in (10.61), obtained by algorithm 10-5, may have dimension $j < n$, if **a zero β_j is encountered in the Lanczos iteration.** In this case, the iteration halts before complete tridiagonalization since q_1 is contained in a proper invariant subspace. This is one of the several mathematical properties of the method [48, theorem 9.1-1]. However, an exactly zero or even small β_j is a rarity in practice. Consequently, other explanations for the convergence of T_j's eigenvalues must be sought from which computable error bounds on T_j's eigenvalues can be computed. For more details and references, see [98-48, chapter 9]. Of course, these additional tests complicate the algorithm and make it computationally less efficient.

- Moreover, **rounding errors greatly affect the behaviour of Lanczos iteration.** The basic difficulty is caused by **loss of orthogonality** among the Lanczos vectors, a phenomenon that confuses the issue of termination ($\beta_n \neq 0$ in algorithm 10-5) and complicates the relationship between the eigenvalues of S and those of the tridiagonal matrices T_j (see (10.68)). Paige [77-78-79] shows that significant departures from orthogonality can be expected when β_j is small. This happens when an eigenvalue of T_j converges to an eigenvalue of S and is thus not the result of the gradual accumulation of roundoff errors. An examination of Paige's results is the best way to motivate the several **practical easy-to-use Lanczos procedures** (see [48, section 9.2]). Most of these are based on a **complete or a selective reorthogonalization** of each newly computed Lanczos vector [98]. Considerable effect has also been spent in trying to develop a workable Lanczos procedure that does not involve any kind of orthogonal enforcement. Research in this direction focusses on the problem of spurious eigenvalues, i.e. multiple eigenvalues of T_j corresponding to a single eigenvalue of S, which arise because the iteration essentially restarts itself when orthogonality is lost [21-70].

In order to **bidiagonalize** an m by n matrix A, using the Lanczos idea, we can proceed just as we did for Lanczos tridiagonalization. We compute the bidiagonal matrix $J_n = U^T A V$ directly with $U = [u_1, \ldots, u_m]$, $V = [v_1, \ldots, v_n]$ orthonormal and J_j given by

$$J_j = \begin{bmatrix} q_1 & e_2 & & \\ & q_2 & \ddots & 0 \\ & & \ddots & e_j \\ & 0 & & q_j \end{bmatrix} \qquad (10.63)$$

by equating columns in $AV = UB$ and $A^T U = V J_n^T$:

$$Av_j = q_j u_j + e_j u_{j-1} \qquad e_1 u_o = 0 \qquad (10.64)$$

$$A^T u_j = q_j v_j + e_{j+1} v_{j+1} \qquad e_{n+1} v_{n+1} = 0 \qquad (10.65)$$

for $j = 1, \ldots, n$. Defining $r_j = Av_j - e_j u_{j-1}$ and $p_{j+1} = A^T u_j - q_j v_j$, we may conclude from orthonormality that $q_j = \pm \|r_j\|_2$, $u_j = r_j/q_j$, $e_{j+1} = \pm \|p_{j+1}\|_2$ and $v_{j+1} = p_{j+1}/\|e_{j+1}\|_2$. Properly sequenced, these equations define the **Lanczos method for bidiagonalizing a rectangular matrix** :

Algorithm 10-6 : Lanczos bidiagonalization

Given : - an m by n matrix A
- a start vector $v_1 \in \mathbb{R}^n$, $\|v_1\|_2 = 1$

$r_1 = Av_1$

$q_1 = \|r_1\|_2$

$u_1 = r_1 / q_1$

For j = 1, 2, ...

$p_{j+1} = A^T u_j - q_j v_j$

$e_{j+1} = \|r_{j+1}\|_2$

if $e_{j+1} = 0$ then stop

$v_{j+1} = p_{j+1} / e_{j+1}$

$r_{j+1} = Av_{j+1} - e_{j+1} u_j$

$q_{j+1} = \|r_{j+1}\|_2$

if $q_{j+1} = 0$ then stop

$u_{j+1} = r_{j+1} / q_{j+1}$

END

The analytic, algorithmic and numerical developments discussed above for Lanczos tridiagonalization, all carry over naturally to the bidiagonalization. Some additional comments are in order :

- The **Lanczos bidiagonalization algorithm 10-6 requires**

$$K_{LB}\ (2mn + 3(m+n)) - m - n \qquad (10.66)$$

operations with K_{LB} the number of Lanczos steps.

- If q_j or e_j is zero we restart the procedure with a new start vector orthogonal onto the Lanczos vectors already found.
- Observe that we have to compute two vector sequences U and V in order to obtain the bidiagonal.
- It was experimentally found that the bidiagonal J_j in (10.63), obtained after j steps of the Lanczos bidiagonalization algorithm 10-6 on A, is related to the tridiagonal T_j in (10.61) obtained after j steps of the Lanczos tridiagonalization algorithm 10-5 on $S = A^TA$ by :

$$T_j = J_j^T J_j \qquad (10.67)$$

Hence, the eigenvalues and eigenvectors of T_j equal the squared singular values and right singular vectors of J_j. We thus **obtain identical results with Lanczos bidiagonalization of A and Lanczos tridiagonalization of A^TA.** The only differences are that Lanczos tridiagonalization is computationally more efficient but the singular values of A may be computed less accurately from T_j because of the squaring.

10.6.2. CONVERGENCE PROPERTIES AND APPLICABILITY

From the previous analysis, it is clear that Lanczos bidiagonalization of A and Lanczos tridiagonalization of A^TA will have the same convergence properties w.r.t. the singular values of A. The famous **Kaniel-Paige theory** reveals how fast the Lanczos process converges towards the desired singular values. Let us consider the case where the **smallest singular value of A is desired**, as required in TLS problems. Then, we have :

Theorem 10-2 :

Let A be an m by n matrix, given by (9.1). If $\theta_1 \geq \ldots \geq \theta_j$ are the eigenvalues of the tridiagonal matrix T_j obtained after j steps of the Lanczos iteration, then

$$\sigma_n^2 \leq \theta_j \leq \sigma_n^2 + (\sigma_1^2 - \sigma_n^2)\ \tan^2 \varphi_n\ /\ C_{j-1}^2\ (d_n) \qquad (10.68)$$

where $d_n = 1 + 2\rho_n$, $\rho_n = (\sigma_{n-1}^2 - \sigma_n^2)/(\sigma_1^2 - \sigma_{n-1}^2)$ and

$\cos \varphi_n = |q_1^T v_n|$, q_1 start vector, $\|q_1\|_2 = 1$ and $C_{j-1}\ (d_n)$ the value in d_n of the (j-1)-th Chebyshev polynomial.

The proof is given in [48, theorem 9.1-3].

The bounds are quite satisfactory in case of convergence to an extremal singular value. If we wish to obtain the singular value σ_n of A up to a precision ε, it follows from (10.68) that we must require :

$$(\sigma_1^2 - \sigma_n^2)\ \tan^2 \varphi_n\ /\ C_{j-1}^2\ (d_n) \leq \varepsilon^2 \qquad (10.69)$$

Approximating $C_{j-1}(d_n)$ by (10.36) in (10.69) and taking its logarithm, we can derive an upper bound for the number K of required iteration steps or equivalently the dimension K of the tridiagonal T_K for **convergence** of the Lanczos algorithm 10-5 or 10-6 **to the smallest singular values σ_n of A up to a precision ε :**

$$K \leq \frac{\frac{1}{2}\log\ (\sigma_1^2 - \sigma_n^2) + \log\ |\tan \varphi_n| + \log\ (2\ \varepsilon^{-1})}{\log\ (d_n + \sqrt{d_n^2 - 1}\ )} + 1 \qquad (10.70)$$

with $d_n = 1 + 2\rho_n$ and $\rho_n = (\sigma_{n-1}^2 - \sigma_n^2)\ /\ (\sigma_1^2 - \sigma_{n-1}^2)$

Comparing θ_j with the corresponding estimates of σ_n by Chebyshev iteration and inverse iteration (computed straightforwardly from the corresponding singular vector as in (5.5)) reveals new quite surprising results about the convergence rate of the corresponding iterative methods. Indeed, using the proof and notation of theorem 10-2, one immediately proves that convergence of σ_n up to a precision ε is obtained if :

- in case of Chebyshev iteration :

$$(\sigma_1^2 - \sigma_n^2) \tan^2 \varphi_n \max_{i=1,n-1} c_{j-1}^2(d_i) / c_{j-1}^2(d_n) \leq \varepsilon^2 \qquad (10.71)$$

with $d_i = f(\sigma_i^2) = 1 + 2\rho_i$ and $\rho_i = (\sigma_i^2 - z)/(z - y)$ from (10.28)

- in case of inverse iteration :

$$(\sigma_1^2 - \sigma_n^2) \tan^2 \varphi_n [(\sigma_n^2 - \lambda_o) / (\sigma_{n-1}^2 - \lambda_o)]^{2j-2} \leq \varepsilon^2 \qquad (10.72)$$

Taking optimal bounds (10.40) with Chebyshev iteration and since $\max_{i=1,n-1} c_{j-1}^2(d_i) \leq 1$, (10.71) implies (10.69). Thus, the number K of iteration steps required by either method is equally bounded by (10.70). Hence we can conclude that **the Chebyshev iteration process with optimal bounds converges as fast as the Lanczos process w.r.t. an extremal singular value of a matrix** (also for the largest singular value one obtains the same result). Observe however that this result is derived from a comparison in upper bounds of the squared desired singular value. From the properties of the Lanczos process, we know that θ_j in (10.68) is in fact the minimum of the RQ $r(x) = x^T Sx / x^T x$ over all vectors x of span $\{q_1, \ldots, Sq_1, \ldots, S^{j-1} q_1\}$. The Chebyshev estimate of σ_n corresponds with a particular vector x in the same span, namely $x = C_{j-1}^{yz}(S) q_1$ with the bounds y,z given by (10.40). This proves that convergence of the Lanczos process will never be slower than the Chebyshev iteration convergence but only minimally faster as we can deduce from theorem 10-1.

The same parameters influencing the convergence of Chebyshev iteration discussed in section 10.3.3, **also determine the convergence of the Lanczos method.** Hence, the number of required iteration steps decreases if the desired accuracy ε is low, the quality of the start vector q_1 good and the convergence rate $\log (d_n + \sqrt{d_n^2 - 1})$ high. Like Chebyshev iteration, the convergence rate depends not only on the gap between σ_{n-1}^2 and σ_n^2 but also on the relative spread of the squared singular value spectrum $(\sigma_1^2 - \sigma_{n-1}^2)/\sigma_{n-1}^2$. Due to this last parameter, **convergence of the Lanczos process will only be fast for matrices having a dense singular value spectrum.** This influence is even more pronounced by the squaring of the singular values, since it enlarges more the gap between the squared largest singular values than the gap between the squared smallest singular values, i.e.

$$(\sigma_{n-1}^2 - \sigma_n^2) / (\sigma_1^2 - \sigma_{n-1}^2) \leq (\sigma_{n-1} - \sigma_n) / (\sigma_1 - \sigma_{n-1}) \qquad (10.73)$$

Thus, **squaring increases** the convergence rate w.r.t. estimation of the **largest** singular value but **decreases** it w.r.t. estimation of the **smallest** singular value. The influence of all those parameters w.r.t. the upper bound (10.70) of the number of required iteration steps is shown in table 10.1 and experimentally confirmed, as illustrated in fig. 10.2.

TABLE 10.1 : The influence of the desired accuracy ε for the smallest singular value σ_n, the angle φ between the start vector and the singular vector associated with σ_n and the singular value spectrum on the number of required iteration steps computed by (10.70).

	singular values $\sigma_1, \ldots, \sigma_{n-1}, \sigma_n$		
	100, ..., 1, 0.1	10, ..., 1, 0.1	5, ..., 1, 0.1
$\varepsilon = 10^{-17}$			
$\varphi = 0.1°$	1915	180	88
$\varphi = 1°$	2031	192	94
$\varphi = 10°$	2147	204	100
$\varphi = 75°$	2300	219	107
$\varepsilon = 10^{-8}$			
$\varphi = 0.1°$	874	76	37
$\varphi = 1°$	989	88	42
$\varphi = 10°$	1106	100	48
$\varphi = 75°$	1259	115	56
$\varepsilon = 10^{-4}$			
$\varphi = 0.1°$	411	30	14
$\varphi = 1°$	527	42	19
$\varphi = 10°$	643	53	25
$\varphi = 75°$	795	69	33

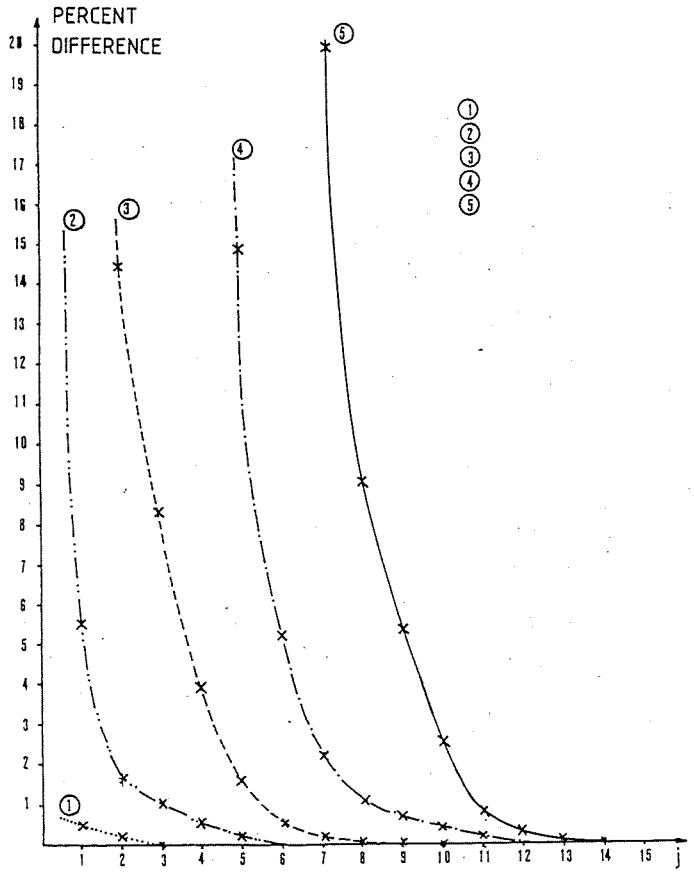


Fig. 10.2.a : Per cent differences between the exact smallest singular value of a 25 by 15 matrix with singular values 10, ..., 1, 0.1 and the smallest singular value of the Lanczos tridiagonal T_j given in (10.61) in function of its dimension j. The start vector is the desired singular vector perturbed by noise with variance (1) 10^{-6}, (2) 10^{-5}, (3) 10^{-4}, (4) 10^{-3} and (5) 10^{-2}.

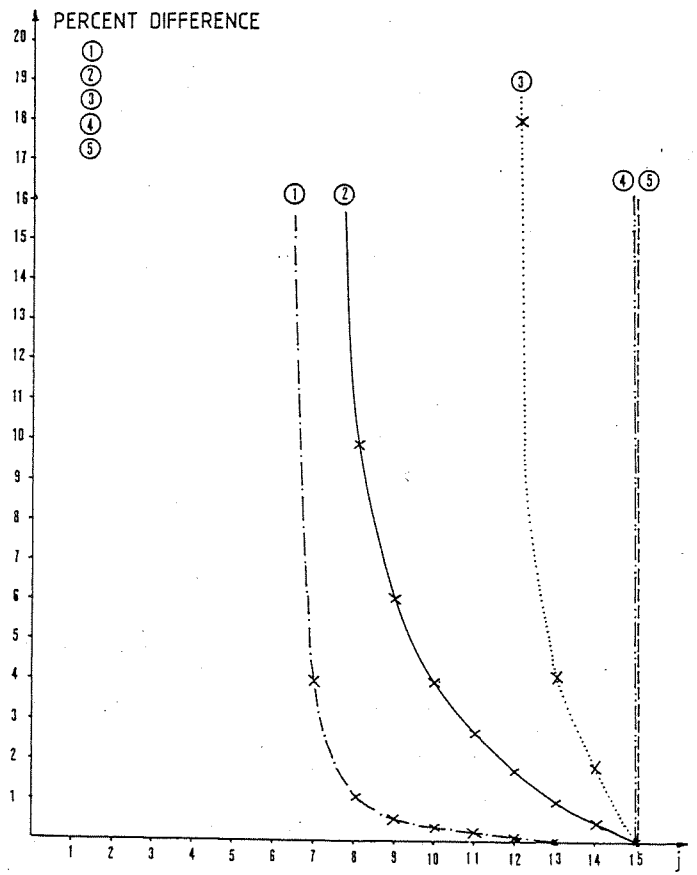


Fig. 10.2.b : Per cent differences between the exact smallest singular value of a 25 by 15 matrix with singular values 100, ..., 1, 0.01 and the smallest singular value of the Lanczos tridiagonal T_j given in (10.61), in function of its dimension j. The start vector is the desired singular vector perturbed by noise with variance (1) 10^{-6}, (2) 10^{-5}, (3) 10^{-4}, (4) 10^{-3}, (5) 10^{-2}.

Just like the Chebyshev and inverse iteration algorithm, **algorithm 10-5 and 10-6 can be used to compute subsequent singular triplets** if we prevent the algorithm to recompute the results already found. This is done by orthogonalizing the newly computed Lanczos vectors q_j in algorithm 10-5 or u_j and v_j in algorithm 10-6 in each step to the singular vectors of A already found. Also extensions to subspace iteration, called block Lanczos methods, are straightforward. See [48, section 9.2].

W.r.t. the TLS problem (0.1) with SVD given in (0.3), one can conclude that **the Lanczos methods can be computationally more efficient than the classical Householder bidiagonalization if :**

- the initial guess vector q_1 is very good
- the desired accuracy ϵ rather low
- the rank of [A;B] known
- the problem highly structured or sparse
- the dimensions m and n large (> 100)
- $m \approx n$
- the gap between desired and undesired singular values large i.e. $\sigma_n^2 >> \sigma_{n+1}^2$
- the relative spread of the squared singular value spectrum given by $(\sigma_1^2 - \sigma_n^2)/\sigma_n^2$ small.

Especially the last requirement is for most TLS problems not satisfied and makes the Lanczos method less efficient than Householder bidiagonalization as soon as the relative spread exceeds 100 for moderate m,n. Moreover, as said in section 10.6.1, the practical Lanczos procedure requires many additional tests and/or orthogonalizations of Lanczos vectors to maintain stability and orthogonality. This additionally reduces the computational efficiency due to an increase in operation counts for each iteration step so that **a study of Lanczos methods for TLS problems is only worthwhile when those problems are large or sparse.**

10.7. OVERALL COMPARISON IN EFFICIENCY OF DIRECT VERSUS ITERATIVE METHODS W.R.T. THE TLS PROBLEM

In this section we compare the iterative methods described in the previous sections w.r.t. their computational efficiency, determined by

- the number K_α of iteration steps required for method α
- the operation counts in each iteration step.

K_α depends primarily on the convergence rate of the iterative method α. From its convergence rate formula, the number of iteration steps can be derived by taking the logarithm, as shown in the previous sections. These estimates are summarized in table 10.2. The operation counts, i.e. the number of multiplications and divisions, in each iteration step, depends on the used method, as well as on its implementation described by the algorithm. By multiplying this number with the number K_α of iteration steps, one obtains the total computational cost of each iterative method α. These results are summarized in table 10.2 for the algorithms described in the previous sections. The operation counts of the classical SVD and PSVD (see chapter 9) are given too.

By **taking the ratio of the total number of operations** in table 10.2, **the computational efficiency of the iterative methods can be compared in function of the dimensions m, n of the original matrix A, the dimension d of the required singular subspace and the number K_α of required iteration steps.** This is done in table 10.3. As long as the number K_α of iteration steps determined by the convergence rate remains smaller than the bound given in table 10.3, the iterative method α is computationally more efficient than the direct or iterative method used in the comparison. Numerical values of these bounds are given in table 10.4, by substituting values of m, n and d. The direct methods classical SVD and PSVD are compared w.r.t. their efficiency in section 9.4.2 (see also table 9.2). Some comments are in order :

- **The computed ratios depend on the algorithm used to implement the respective methods.** The choice of our algorithms (given in the previous sections) is optimized w.r.t. the solution of TLS problems. In particular, the iterative method must converge to singular vectors associated with the **smallest** singular values and the original matrix is assumed to be **general.** This means that sparsity or structure in the

original matrix is not exploited to reduce further the computational cost.

- **Extra calculations** which deal with under- and overflow are **not taken into account**. Also, the additional computations required to maintain stability in Lanczos methods (e.g. by orthogonalizing the Lanczos vectors) and control their level of convergence in a practical easy-to-use Lanczos procedure, are not considered here.

- In comparing the Lanczos methods with Householder bidiagonalization, we did not take into account the additional computations required to obtain the desired singular vectors from the obtained bidiagonal. These operations are generally negligible w.r.t. the bidiagonalization cost and influence only minimally the upper bound on the number of Lanczos steps, computed from table 10.3, for moderate dimensions m, n. It is easily seen that the additional computation time for computing the desired singular vector of an m by n matrix A from the j by j Lanczos tri- or bidiagonal, is a factor j/n better than the cost when starting from the n by n bidiagonal obtained by Householder transformations.

With these comments in mind, the following **conclusions** concerning the **efficiency of solving TLS problems** can be drawn from the tables 10.2, 10.3 and 10.4 :

- **The efficiency of an iterative method in general depends on many parameters. The most important** ones are **those which influence its convergence rate.** This factor is first of all determined by **the quality of the initial information**, such as knowledge of good bounds, shift, initial guess vector. **If these parameters can not be obtained sufficiently accurately, then iterative methods can fail to converge** to the desired n-th singular vector v_n associated with the smallest singular value σ_n of matrix A whose SVD is given by (9.1). This happens with Chebyshev iteration when the bounds y,z are not well defined such that $z < \sigma_1^2$ or $y > \sigma_{n-1}^2$. If the shift λ_o makes the ratio (10.52) smaller than one, then inverse iteration does not converge to the desired singular vector v_n. Note however that a zero shift always ensures convergence to v_n. RQ iteration fails to converge to v_n when the component

TABLE 10.2 : Comparison of the direct and iterative methods w.r.t. their efficiency, determined by their operation counts (2nd column) and convergence rate (3rd column). An m by n matrix A with singular values $\sigma_1, \dots, \sigma_n, \sigma_1 \geq \dots \geq \sigma_n$, and corresponding right singular vectors $v_1, \dots, v_n$ is considered; d base vectors of a singular subspace of A associated with its d smallest singular values $\sigma_{n-d+1}, \dots, \sigma_n$ are desired, up to a precision ε. q_1 is a start vector, $\|q_1\|_2 = 1$, making an angle ϕ_n with v_n, i.e. $\cos\phi_n = |v_n^T q_1|$.

Method	total number of operation counts	Number of required iteration steps
Classical SVD (direct method)	$m < 5n/3$: $2mn^2+2sn^3+(3/2+8s)n^2+(5/2-s)n-2m-10-9s$	cubic
	$m \geqslant 5n/3$: $mn^2+(5/3+2s)n^3+(2+8s)n^2+mn+(4/3-s)n-17-9s$ with s the average number of required QR iteration steps for convergence to one singular vector.	cubic
Partial SVD (direct method)	$m < 5n/3$: $2mn^2-2n^3/3+[2+(4s+1)d]n^2+(8/3+22sd-2sd^2)n-2m-12sd^2-(11s+4)d-7$	cubic
	$m \geqslant 5n/3$: $mn^2+n^3+[5/2+(4s+1)d]n^2+mn+(3/2+22sd-2sd^2)n-12sd^2-(11s+4)d-14$ with s the average number of required QR/QL iteration steps for convergence to one singular vector.	cubic
Chebyshev iteration (algorithm 10-1)	$n(2m+5)K_C$ with K_C the number of required Chebyshev iteration steps for convergence to one singular vector (d = 1)	$K_C \approx (\log(2\varepsilon^{-1})+\log M)/\log(d_n+\sqrt{d_n^2-1})$ with $M = \max_{j=1,n-1} \lvert v_j^T q_1 / v_n^T q_1 \rvert$ or $M = \lvert \tan\varphi_n \rvert$, $d_n = 1+2\rho_n$, $\rho_n = (y-\sigma_n^2)/(z-y)$ and $\sigma_n^2 < y \leqslant \sigma_{n-1}^2$ and $z \geqslant \sigma_1^2$

Inverse iteration (algorithm 10-2)	$mn(n+1)/2+n^3/6+3n^2/4+7n/3+(n^2+5n)K_I$ with K_I the number of required inverse iteration steps for convergence to one singular vector (d=1)	$K_I \approx (\log\varepsilon^{-1}+\log M)/\log(\lvert\sigma_{n-1}^2-\lambda_0\rvert/\lvert\sigma_n^2-\lambda_0\rvert)$ with $M = \max_{j=1,n-1} \lvert v_j^T q_1\rvert/\lvert v_n^T q_1\rvert$ or $M = \lvert\tan\varphi_n\rvert$. λ_0 is the shift
Simultaneous inverse iteration (algorithm 10-3)	$mn(n+1)/2+n^3/6+3n^2/4+7n/3+[dn^2+(3+2d)dn-2d^3/3+2d^2-d/3-4]K_{SI}$ with K_{SI} the number of required simultaneous inverse iteration steps for convergence to d singular vectors	$K_{SI} \approx \dfrac{\log\varepsilon^{-1} + \log\lvert\tan\Phi\rvert}{\log(\lvert\sigma_{n-d}^2-\lambda_0\rvert/\lvert\sigma_{n-d+1}^2-\lambda_0\rvert)}$ with λ_0 the shift, Φ the maximal angle between the n by d start matrix Q_1 and $V_n = [v_{n-d+1}, \ldots, v_n]$, defined by $\sin\Phi = \mathrm{dist}\,(Q_1, V_n)$
RQ iteration (algorithm 10-4)	$mn(n+1)/2+2n^3/3+5n^2/2-n/6-13+(14n-9)K_{RQ}$ with K_{RQ} the number of required RQ iteration steps for convergence to one singular vector (d=1)	cubic
Lanczos methods (algorithms 10-5 and 10-6)	tridiagonalization (algorithm 10-5) : $n(2m+5)K_L - n$ bidiagonalization (algorithm 10-6) : $[2mn+3(m+n)]K_L - m - n$ with K_L the number of required Lanczos iteration steps for convergence of the j-th singular value of the tridiagonal T_j or bidiagonal J_j to $\sigma_n(A)$	$K_L \leqslant \dfrac{\frac{1}{2}\log(\sigma_1^2-\sigma_n^2)+\log\lvert\tan\varphi_n\rvert+\log(2\varepsilon^{-1})}{\log(d_n+\sqrt{d_n^2-1})} + 1$ with $d_n = 1+2\rho_n$, $\rho_n = (\sigma_{n-1}^2-\sigma_n^2)/(\sigma_{L-1}^2-\sigma_n^2)$

TABLE 10.3 : Upper bounds on the number K_α of iteration steps for method α, such that method α (first column) is computationally more efficient, i.e. requires less operation counts, than method β (second column). d base vectors of the singular subspace of an m by n matrix A associated with its d smallest singular values, are desired and s is the average number of required QR/QL iteration steps for convergence to one singular vector with the considered direct method β.

	Method α more efficient than method β		If the number K_α of iteration steps is bounded by
①	Lanczos tridiagonalization (algor. 10-5), d = 1	Partial SVD d = 1	$m < 5n/3 : K_L \leqslant (2m-2n/3+2)n/(2m+5)$ $m \geqslant 5n/3 : K_L \leqslant [(m+n+5/2)n+n]/(2m+5)$
②	Chebyshev iteration (algor. 10-1), d = 1	Classical SVD	$m < 5n/3 : K_C \leqslant (m+sn+4s+3/4)n/(m+5/2)$ $m \geqslant 5n/3 : K_C \leqslant [m+(5/3+2s)n+8s+2+m/n]n/(2m+5)$
③	Chebyshev iteration (algor. 10-1), d = 1	Partial SVD d = 1	$m < 5n/3 : K_C \leqslant (m-n/3+2s+3/2)n/(m+5/2)$ $m \geqslant 5n/3 : K_C \leqslant (m+n+4s+7/2+m/n)n/(2m+5)$
④	Inverse iteration (algor. 10-2), d = 1	Classical SVD	$m < 5n/3 : K_I \leqslant [9m+(12s-1)n+48s+9/2-3m/n]n/(6n+30)$ $m \geqslant 5n/3 : K_I \leqslant [n+(3+4s)n+16s+5/2+m/n]n/(2n+10)$
⑤	Inverse iteration (algor. 10-2), d = 1	Partial SVD d = 1	$m < 5n/3 : K_I \leqslant [9m-5n+24s+27/2-3m/n]n/(6n+30)$ $m \geqslant 5n/3 : K_I \leqslant [3m+5n+24s+33/2+3m/n]n/(6n+30)$
⑥	Chebyshev iteration (algor. 10-1), d = 1	Inverse iteration (algor.10-2) d = 1	$K_C \leqslant [(n+1)n/2+n^2/6+5n/4+(n+5)K_I]/(2n+5)$
⑦	RQ iteration (algor. 10-4), d = 1	Classical SVD	$m < 5n/3 : K_{RQ} \leqslant [19m+(12s-4)n+48s-61n-3m+16-6s-12m/n]/84$ $m \geqslant 5n/3 : K_{RQ} \leqslant [(m+2(1+2s)n+16s-1)n+m+3-2s]/28$
⑧	RQ iteration (algor. 10-4), d = 1	Partial SVD d = 1	$m < 5n/3 : K_{RQ} \leqslant [19m-8n+24s+3)n-3m+17+120s-12m/n]/84$ $m \geqslant 5n/3 : K_{RQ} \leqslant [13m+2n+24s+6)n+3m+10+120s]/84$
⑨	Simultaneous inverse iteration (algor. 10-3)	Partial SVD	$m < 5n/3 : K_{SI} \leqslant [19m-5n+(24s+6)d+15/2)n-m/2]/[6d(n+2d+3)]$ $m \geqslant 5n/3 : K_{SI} \leqslant [13m+5n+6(4s+1)d+21/2)n+3m]/[6d(n+2d+3)]$ (if $d \ll m$ and $d \ll n$)
⑩	Chebyshev iteration (algor. 10-1), d = 1	RQ iteration (algor.10-4) d = 1	$K_C \leqslant [n+1)m/2+2n^2/3+5n/2+14K_{RQ}]/(2m+5)$
⑪	Inverse iteration (algor. 10-2), d = 1	RQ iteration (algor.10-4) d = 1	$K_I \leqslant [(2n^2+7n-10)/4+14K_{RQ}]/(n+5)$
⑫	RQ iteration (algor. 10-4), d = 1	Inverse iteration (algor.10-2) d = 1	$K_{RQ} \leqslant [(n+5)K_I - (2n^2+7n-10)/4]/14$

TABLE 10.4 : Numerical values of the upper bounds on the number K_α of iteration steps for method α such that method α requires less operation counts than method β for specific values of m, n, d and s (or K_β). d base vectors of the singular subspace of an m by n matrix A associated with its d smallest singular values, are desired. s is the average number of required QR/QL iteration steps for convergence to one singular vector using the considered direct method β.

matrix size			K_s^β	pairs α, β of computation methods (1), ... (12) refer to table 10.3										
m	n	d		(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
10	8	1	2	5.3	22.2	8.2	38.0	11.1	3.5	31.9	9.9	....	5.4	5.5	0.0
20	18	1	2	12.0	51.8	15.6	90.2	19.3	6.7	134.2	20.8		10.6	9.5	0.0
30	28	1	2	18.7	81.6	22.5	143.0	26.6	10.0	305.5	34.2		16.2	14.1	0.0
18	10	1	2	7.9	23.0	10.1	54.8	19.3	3.7	53.8	18.9		5.3	6.2	0.0
40	20	1	2	15.2	40.8	17.3	118.6	38.7	6.5	195.0	55.4		9.0	10.4	0.0
50	30	1	2	24.0	68.5	26.6	178.4	52.8	9.7	410.0	99.0		14.1	15.1	0.0
100	50	1	2	37.7	98.4	39.9	312.0	94.0	15.2	1130.3	276.8		21.3	24.8	0.0
20	18	2	2									12.3			
50	30	3	2									20.4			

of the initial guess vector q_1 in the direction of v_n is not largest. Second, the convergence rate of an iterative method depends on **the matrix characteristics** such as its **dimension** m and n, the **spread** of its singular value spectrum (an important factor in Chebyshev iteration and the Lanczos methods) and the **gap** between the singular values associated with the desired and undesired singular vectors. And third, **the desired accuracy** ε influences the convergence rate. Moreover, the efficiency of an iterative method is determined by **the number d of desired singular vectors.**

- **The efficiency of the direct methods on the other hand is determined by only a few parameters,** namely the **dimensions** m and n of the considered matrix. In case of the PSVD method, the efficiency also depends on the desired accuracy ε, the number d of desired singular vectors and the gap of their corresponding singular values w.r.t. the undesired singular values. Moreover, **the direct methods** classical SVD and PSVD **always converge to the desired base vectors for any matrix A.**

- **The iterative methods described here are most efficient when only one singular vector is required** e.g. in generic one-dimensional TLS problems with unique solution. In case of multidimensional TLS problems, the algorithms must be extended to allow subspace iteration. This increases the operation counts in each iteration step and reduces the efficiency of the iterative method.

- Moreover **when using iterative methods, the rank of the matrix must always be known** or equivalently **the number d of desired singular vectors.** The direct methods on the contrary also allow to compute the base vectors of a singular subspace of a matrix associated with its singular values smaller than or equal to a given bound θ. By using deflation techniques or by orthogonalizing the iteration vector to the singular vectors already found in each step, it is however possible to compute subsequent singular triplets with an iterative method until all singular vectors associated to singular values $\leq \theta$ are computed. But hereto, the number of operations in each iteration step is increased and also one additional singular triplet corresponding to a singular value $> \theta$, must be computed to assure convergence to all sin-

gular values $\leq \theta$. It is clear that these operations make the iterative methods less efficient.
This is the case for TLS problems (0.1) when the rank of the matrix [A;B] is determined by a user defined error bound θ or a noise dependent error level, e.g. (1.117), or when the TLS problem is nongeneric and thus a basis of the singular subspace corresponding to the next larger singular value has to be computed. Moreover in case of nongeneric TLS problems, the initial guess vector used for calculating the next singular triplet with an iterative method, is in general of bad quality so that convergence to the desired TLS solution is no longer guaranteed.

- **W.r.t. solving generic one-dimensional TLS problems (0.1) with unique solution, the inverse iteration method is generally superior to the other iterative methods in efficiency** because its convergence rate depends solely on the ratio $|\sigma_n^2 - \lambda_o|/|\sigma_{n+1}^2 - \lambda_o|$ and not on the spread of the squared singular value spectrum of the matrix [A;b], as is the case with Chebyshev iteration and the Lanczos methods. For most TLS problems this ratio is high. This assures fast convergence so that RQ iteration will not perform better. **Depending on the quality of the initial information, inverse iteration is also usually more efficient than the direct methods classical SVD and PSVD in those cases.** This is illustrated in section 10.8 for a real life problem concerning the estimation of frequency response functions.

Besides the efficiency, **other factors may be important in the comparison between iterative and direct methods.** This may influence the choice of the "best" method :

- **Each step of an iterative method gives a new and better estimate** of the solution. This allows to control the level of convergence depending on the perturbations of the data. This is also the case for PSVD.
- **Iteration is easy to code.**
- Furthermore, **most iterative methods** (except inverse iteration and RQ iteration) **do not modify the matrix during the computations.** This is important for sparse or highly structured matrices. Since iteration does not destroy the structure of the matrix it can be exploited to reduce further the computational cost. Moreover, the required memory space can be kept minimal since no intermediate (and full) submatrices associated to the factorization or decomposition, are generated. This may be important in large-scale problems.
- The **explicit squaring A^TA of the matrix A** in inverse iteration, RQ iteration and Lanczos tridiagonalization **decreases the quality of the information stored in A.** This well-known problem has been treated by many authors, see e.g. [67-105]. If ε_m denotes the finite data storage precision, then the elements of A^TA are stored in memory with absolute accuracy ε_m. This means that effects in A smaller than $\sqrt{\varepsilon_m}$, completely disappears in the stored A^TA. This restricts the maximal attainable accuracy in the computation of the desired singular vector and singular value of A by using its relationship with the eigenvectors and eigenvalues of A^TA, to $\sqrt{\varepsilon_m}$. Hence, we must choose the desired accuracy ε larger than $\sqrt{\varepsilon_m}$. For example, when $\varepsilon_m = 10^{-14}$, ε can not be smaller than 10^{-7}. In this case, we may use inverse iteration, RQ iteration and Lanczos tridiagonalization and converge to the desired singular vector up to the desired precision ε.

10.8. EFFICIENCY OF INVERSE ITERATION AND OF THE PARTIAL AND CLASSICAL TLS ALGORITHM W.R.T. THE ESTIMATION OF FREQUENCY RESPONSE FUNCTIONS

10.8.1. PROBLEM DESCRIPTION

In this section we compare the 3 best techniques for solving TLS problems, namely the classical TLS algorithm (see section 1.8.1), the partial TLS algorithm (see section 9.3.4) and the inverse iteration method (see section 10.4.1) w.r.t. their efficiency in a **real life** problem concerning the **estimation of frequency response functions** (FRF). Until today, the FRF remains the most important measurement for experimental structural dynamical analysis. In recent years, technology for measuring FRF's has evolved at a fast pace. Very significant has been the successful application of multiple input measurement of FRF's [4-5], coupled with broad band excitation signals. This measurement technique is also developed at the mechanical department of the K.U.Leuven.

Consider a structure excited simultaneously at n input locations. Its

response is measured at d output locations. The idea is to **estimate the FRF's** $h_1, \ldots, h_n$ **of a linear system between the n inputs and the d outputs.** Assume, for convenience, one output (d=1). Let, at a specific frequency, the **spectra of the true inputs** to the system be represented by $(x_o)_1, \ldots, (x_o)_n$ and of the **true response** by y_o. Assuming a linear system, the following model is valid :

$$(x_o)_1 \, h_1 + \ldots + (x_o)_n \, h_n = y_o \qquad (10.74)$$

The spectra that are actually measured will not represent the true inputs and outputs, due to measurement noise. We represent the measured spectra by $x_1, \ldots, x_n$ for the inputs, by y for the output and the spectra of the noise on the inputs by $\Delta x_1, \ldots, \Delta x_n$ and on the output by Δy. Substituting in equation (10.74) the true spectra by the measured spectra and noise terms, we derive :

$$(x_1 - \Delta x_1) \, h_1 + \ldots + (x_n - \Delta x_n) \, h_n = y - \Delta y \qquad (10.75)$$

Note that an errors-in-variables model is assumed here (see section 4.1). An equation like (10.75) can be written for each measurement sample. Using m measurement samples at the specific frequency, a total of m equations is generated. If m is larger than n, this set is overdetermined. In matrix form, we obtain :

$$[X;Y] \, [H^T;-1]^T = 0 \qquad (10.76)$$

The unknowns $H = [h_1, \ldots, h_n]^T$ and the i-th row of X and Y contains the measured spectra of the n inputs and output respectively. **To solve this overdetermined set of equations for estimates of** $h_1, \ldots, h_n$ **at each frequency point, the TLS technique is appropriate** and shown to be more accurate than the ordinary LS technique [91].

The frequency spectra are measured with broad band excitation and using digital signal analysis. Those measurements are almost always based on the use of the Fast Fourier Transform (FFT) algorithm. It is well-known that **the accuracy of the measured spectra depends heavily on** how well one controls a digital signal processing error, known as **leakage** [6]. Leakage is caused by the violation of a fundamental assumption of the FFT, namely that the measured spectra should be periodic or transient in the observation period. This error can be mathematically corrected

by use of special time windows or by use of special excitation signals [15] such as burst random that makes the signal periodic or transient in the observation period.

A fundamental characteristic of broad band spectra obtained with the FFT, is that the **frequency spacing** Δf **is uniform for the complete band.** This may result in too large a resolution between resonance frequencies and too coarse a resolution around the resonances, especially for lightly damped structures. Therefore other excitation techiques such as stepped sine testing, have been tried [68]. Typically a base band is taken from 0 Hz to 200 Hz with a frequency resolution of 512 equally spaced points such that Δf = 200/512 Hz. In this case we have to solve 512 sets (10.76) of m equations in n unknowns $h_1(f_i), \ldots, h_n(f_i)$, corresponding to 512 frequency points $f_i = i\Delta f$, $0 \leq i \leq 511$, at which m measurement samples are taken, in order to compute the FRF's $h_1(f)$, $\ldots$ $h_n(f)$, at each output. From the previous, it follows that the solution of (10.76) obtained at a frequency point f_i is a good initial guess for the solution of (10.76) at f_{i+1}, except around the resonance frequencies and the end points of the base band. Hence, the use of an iterative method such as inverse iteration is appropriate here, as shown in subsection 10.8.3.

10.8.2. METHOD DESCRIPTION

The TLS estimates of the true (unknown) values $h_1(f_i), \ldots, h_n(f_i)$ of the FRF's at a specific output **are always computed from the coefficients of the eigenvector** v_n**, associated with the smallest eigenvalue** λ_n **of the squared matrix** $[X;Y]^H \, [X;Y]$, derived from (10.76) [68]. The superscript H denotes the complex conjugate of the matrix. We thus use the relationship between the SVD of a matrix $[X;Y]$ and the eigenvalue decomposition of its associated squared, hermitian matrix :

$$[X;Y]^H [X;Y] = \left[\begin{array}{c|c} X^HX & X^HY \\ \hline Y^HX & Y^HY \end{array}\right] = \left[\begin{array}{ccc|c} GX_1X_1 & \cdots & GX_1X_n & GX_1Y \\ \vdots & & \vdots & \vdots \\ GX_nX_1 & \cdots & GX_nX_n & GX_nY \\ \hline GYX_1 & \cdots & GYX_n & GYY \end{array}\right]$$

with $GX_iX_j = GX_jX_i = \sum_{k=1}^{m} x_{ki}^H x_{kj}$ (H denotes the complex conjugate)

$$GX_iY = GYX_i = \sum_{k=1}^{m} x_{ki}^H y_k$$

$$GYY = \sum_{k=1}^{m} y_k^H y_k \qquad (10.77)$$

GX_iX_j is the crosspower, or autopower if i = j, between the measured input spectra x_i and x_j. GX_iY is the crosspower between the measured input spectrum x_i and measured output spectrum y. GYY is the autopower of the output spectrum y. **Those auto- and crosspowers** in (10.77) are immediately computed during data collection and **are the only available data** so that we have to compute the FRF's from the eigenvector equations. Using theorem A.3, it follows by contraposition that (10.77) has a nonrepeated smallest eigenvalue and therefore a unique associated eigenvector if the submatrix X^HX is quasi nonsingular. This is so when the inputs are not correlated [91]. In this case, the TLS problem is generic and inverse iteration may be safely used.

If the response of the system is measured at more than one output, i.e. d > 1, we have a multidimensional TLS problem (see section 1.4). The d TLS solutions $h_{1i}(f_j)$, ..., $h_{ni}(f_j)$, corresponding to each output i = 1, ..., d, at every frequency point f_j, can be computed simultaneously by solving :

$$[X;Y]^H [X;Y][H^T;-I]^T = 0$$

where $H = \begin{bmatrix} h_{11} & \cdots & h_{1d} \\ \vdots & & \vdots \\ h_{n1} & \cdots & h_{nd} \end{bmatrix}$

and

$$[X;Y]^H [X;Y] = \left[\begin{array}{c|c} X^HX & X^HY \\ \hline Y^HX & Y^HY \end{array}\right] = \left[\begin{array}{ccc|ccc} GX_1X_1 & \cdots & GX_1X_n & GX_1Y_1 & \cdots & GX_1Y_d \\ \vdots & & \vdots & \vdots & & \vdots \\ GX_nX_1 & \cdots & GX_nX_n & GX_nY_1 & \cdots & GX_nY_d \\ \hline GY_1X_1 & \cdots & GY_1X_n & GY_1Y_1 & \cdots & GY_1Y_d \\ \vdots & & \vdots & \vdots & & \vdots \\ GY_dX_1 & \cdots & GY_dX_n & GY_dY_1 & \cdots & GY_dY_d \end{array}\right]$$

with GX_iX_j, GX_iY_j and GY_iY_j given in (10.77). (10.78)

H is obtained from the d eigenvectors v_{n-d+1}, ..., v_n associated with the d smallest eigenvalues λ_{n-d+1}, ..., λ_n of $[X;Y]^H [X;Y]$ by solving :

$$HV_{22} = -V_{12} \quad \text{where} \quad \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix} = [v_{n-d+1}, \ldots, v_n] \qquad (10.79)$$

Observe that in order to compute the FRF's at d outputs simultaneously with TLS, we need to know all the crosspowers GY_iY_j between the measured output spectra y_i and y_j. Based on the operation counts of the PSVD and classical SVD algorithm (see section 9.4.2), it follows that **simultaneous computation of the FRF's at all outputs by solving one d-dimensional TLS problem (10.78), is computationally more efficient than solving d one-dimensional TLS problems in case the dimension of the corresponding matrix (10.78) does not increase too much,** i.e. if (n+d)/(n+1) approaches one. Indeed from table 9.1 (case a), it is clear that the number of operations performed on an n by n matrix, is proportional with n^3. Hence, solving a d-dimensional TLS problem (10.78) requires less computations, if the increased computational cost due to the increase in dimension of the matrix, is less than d times the cost of solving a one-dimensional TLS problem (10.76), i.e. if $(n+d)^3 < d(n+1)^3$ approximately. Furthermore, a multidimensional TLS problem requires the solution of a set equations (10.79) (instead of a simple scaling). On the other hand, the number of required QR/QL iteration steps in the PTLS (or PSVD) algorithm increases only slightly (factor less than d) if the set of equations (10.78) is not highly incompatible. This is the case when the gap between the eigenvalues λ_{n-d} and λ_{n-d+1} of (10.78) is large. This is illustrated in the example of section 10.8.3.

Note that we deal with complex matrices. This causes no problem since each complex SVD or complex set of equations **can be transformed into the real domain.** Splitting the complex set XH = Y into its real and imaginary part :

$$(X_r + jX_c)\ (H_r + jH_c) = Y_r + jY_c \tag{10.80}$$

it is easily proven [23] that the solution H is also obtained by solving the real problem :

$$\begin{bmatrix} X_r & -X_c \\ X_c & X_r \end{bmatrix} \begin{bmatrix} H_r \\ H_c \end{bmatrix} = \begin{bmatrix} Y_r \\ Y_c \end{bmatrix} \tag{10.81}$$

Equivalently, the complex SVD of a hermitian matrix X :

$$X(v_{ir} + jv_{ic}) = (X_r + jX_c)\ (v_{ir} + jv_{ic}) = \sigma_i\ (u_{ir} + ju_{ic}) \tag{10.82}$$

where v_{ir}, u_{ir} (resp., v_{ic}, u_{ic}) denote the real (resp., imaginary) part of the right and left singular vector associated with the real singular value σ_i (X is hermitian), can be transformed into the real domain as follows :

$$X_R \begin{bmatrix} v_{ir} \\ v_{ic} \end{bmatrix} = \begin{bmatrix} X_r & -X_c \\ X_c & X_r \end{bmatrix} \begin{bmatrix} v_{ir} \\ v_{ic} \end{bmatrix} = \sigma_i \begin{bmatrix} u_{ir} \\ u_{ic} \end{bmatrix} \tag{10.83}$$

The 2n by 2n real symmetric matrix X_R has the same singular values as the n by n complex matrix X but they are of multiplicity 2. However, all base vectors of its associated singular subspace correspond with the same complex singular vector. Indeed, complex singular vectors and eigenvectors are only defined up to a factor $e^{j\omega}$ [57]. It is thus sufficient to calculate only one singular vector of X_r associated with σ_i, in order to derive the corresponding complex singular vector of X from it.

Finally, the cross- and autopowers GX_iX_j, GY_iY_j can differ several orders of magnitude. Since the errors are proportional to the size of the data, all entries in (10.77) or (10.78) are scaled such that they have approximately the same order of magnitude (see (10.84)).

10.8.3. COMPUTATIONAL RESULTS

In order to compare the efficiency of the classical and partial TLS algorithm and the inverse iteration method in the estimation of FRF's, we make use of the cross- and autopowers of a tested structure from Leuven Measurement & Systems (LMS). The calculations are performed in double precision on the IBM 4381 P03 of the Computer Center of the K.U.Leuven.

Application example : A benzine tank excited simultaneously at **2 input locations** (n = 2) using broad band excitation signals, is used as test structure and the response is measured at **4 output locations** (d = 4). The broad band measurements are performed using a commercial multichannel Fourier analyzer; the DAC of this system is used to generate the excitation signals. The broad band measurements are taken for a base band from 0 Hz to 100 Hz with a frequency resolution of 512 **equally spaced** points. The FRF's $h_1(f)$ and $h_2(f)$ at each output location, are then obtained by solving 512 sets of equations, corresponding to the 512 frequency points $f_i = i\Delta f$ for i = 0, ..., 511 with Δf = 100/512 Hz. By plotting the amplitude of these solutions $h_1(f_i)$ and $h_2(f_i)$ for i = 0, .., 511 in function of the frequency f, the amplitude of the FRF's can be drawn as shown in figures 10.3, 10.4, 10.5, 10.6.

In order to scale the data appropriately, LMS uses the following factor for all sets of equations :

$$\left[\frac{1}{n}\left(\sum_{i=1}^{n}\sqrt{\sum_{k=1}^{512}(GX_iX_i)_k^2}\right)\right] \Big/ \left[\frac{1}{d}\left(\sum_{j=1}^{d}\sqrt{\sum_{k=1}^{512}(GY_jY_j)_k^2}\right)\right] \tag{10.84}$$

$(GX_iX_i)_k$ and $(GY_jY_j)_k$ are the autopowers of the spectra of the i-th input and j-th output respectively (see 10.77), corresponding to the k-th frequency point. By multiplying all crosspowers GY_iY_j with (10.84) and all GX_iY_j with the square root of (10.84), the entries as well as the errors in the matrix of (10.76) or (10.78) have approximately the same order of magnitude.

In order to compare the efficiency of the 3 techniques for solving TLS problems, we register the computation times required for solving each set of equations and take the average. Those average computation times, expressed in milliseconds, are given below. A relative solution accuracy of $\varepsilon = 10^{-7}$ is required. All complex TLS problems are transformed into the real domain.

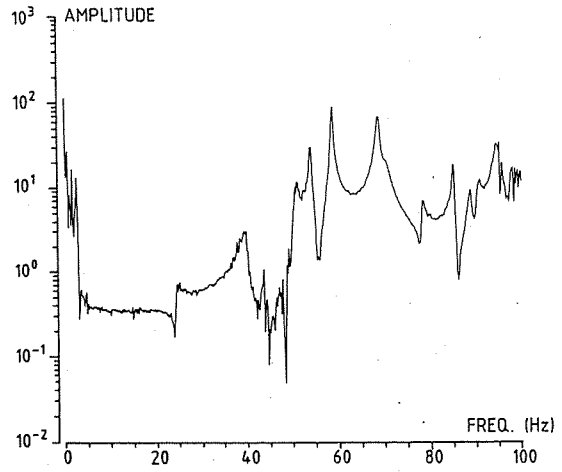


Fig. 10.3.a : Amplitude frequency response function h_1 output 1 (1-dim. TLS problem).

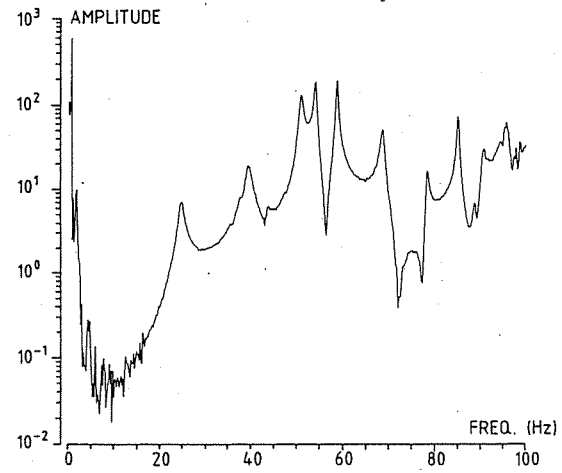


Fig. 10.3.b : Amplitude frequency response function h_2 output 1 (1-dim. TLS problem).

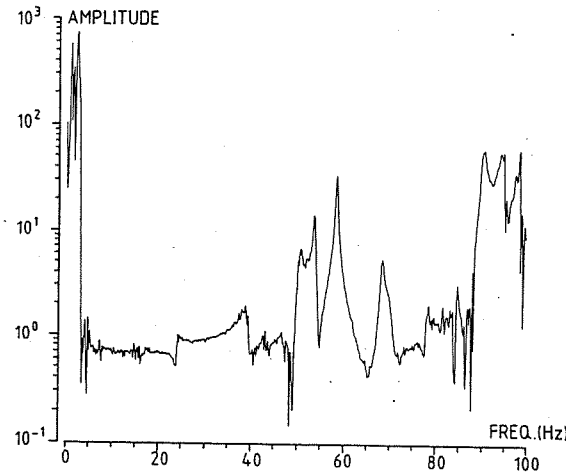


Fig. 10.4.a : Amplitude frequency response function h_1 output 2 (1-dim. TLS problem).

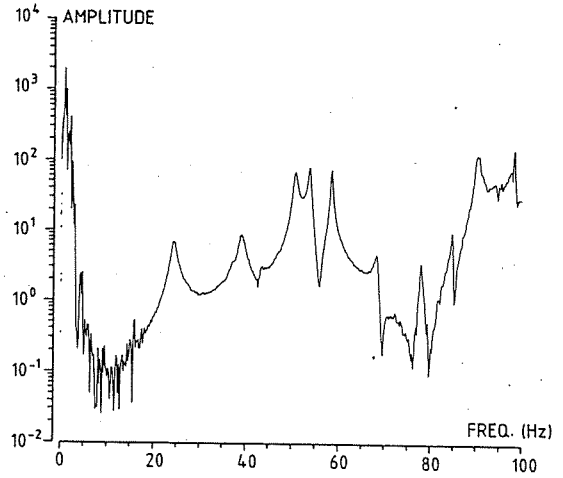


Fig. 10.4.b : Amplitude frequency response function h_2 output 2 (1-dim. TLS problem).

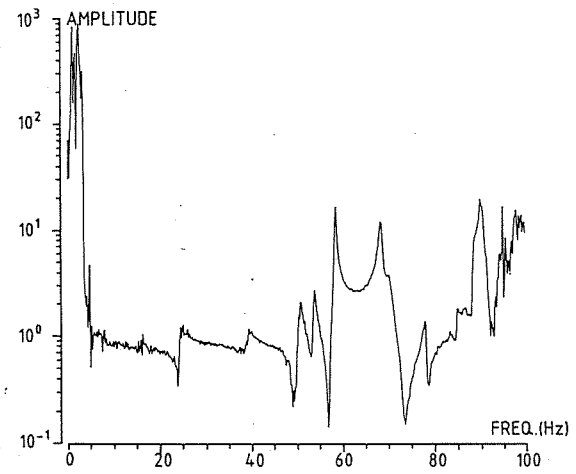


Fig. 10.5.a : Amplitude frequency response function h_1 output 3 (1-dim. TLS problem).

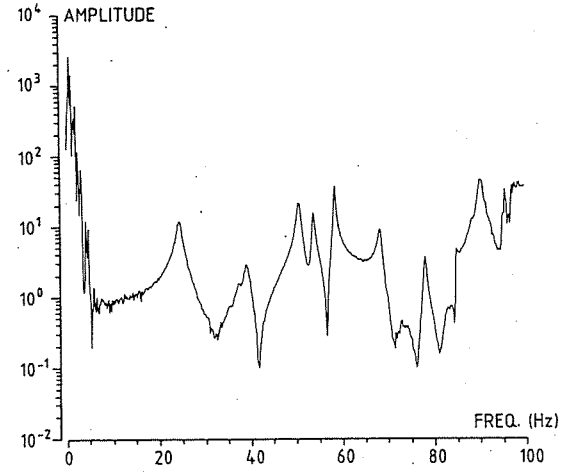


Fig. 10.5.b : Amplitude frequency response function h_2 output 3 (1-dim. TLS problem).

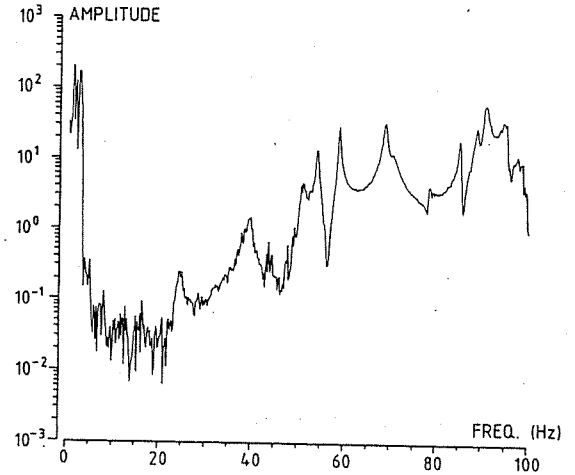


Fig. 10.6.a : Amplitude frequency response function h_1 output 4 (1-dim. TLS problem).

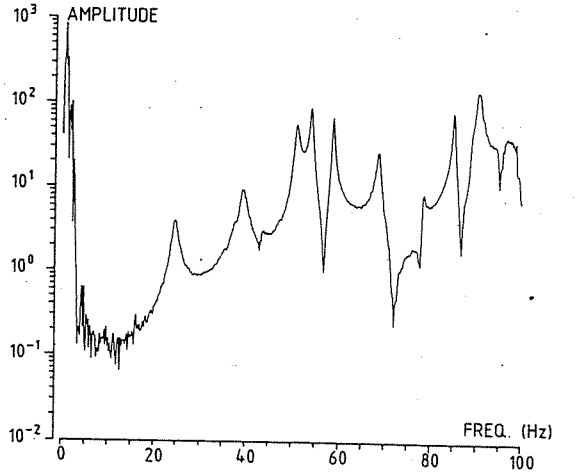


Fig. 10.6.b : Amplitude frequency response function h_2 output 4 (1-dim. TLS problem).

First, we consider **the computation of the FRF's at each output location separately** by solving 4 times 512 one-dimensional TLS problems (10.76) with 2 inputs (n = 2) and one output (d = 1). Using the classical and partial TLS algorithm, the following average CPU times are obtained for computing the response at each of the 4 output locations (see fig. 10.3, 10.4, 10.5 and 10.6) :

	OUTPUT 1		OUTPUT 2		OUTPUT 3		OUTPUT 4	
	PTLS	cl.TLS	PTLS	cl.TLS	PTLS	cl.TLS	PTLS	cl.TLS
Bidiagonalization	4.3	4.3	4.3	4.3	4.4	4.4	4.4	4.4
Initialization V		1.2		1.2		1.2		1.2
Diagonalization	1.1	6.2	1.1	6.3	1.1	6.6	1.1	6.3
Back transformation	0.5		0.5		0.5		0.5	
TLS solution	0	0	0	0	0	0	0	0
TOTAL	5.9	11.7	5.9	11.8	6.0	12.2	6.0	11.9

Observe that **PTLS is about two times faster than the classical TLS** algorithm. This result agrees with our analysis given in section 9.4.2.

For inverse iteration, we take as initial guess vector, the eigenvector associated with the smallest eigenvalue λ_n of the squared matrix $[X;Y]^H [X;Y]$ in (10.76), obtained at the previous frequency point. Only for the first set, we have to start from an arbitrary vector. The following average CPU times and average number of iteration steps are obtained :

	OUTPUT 1	OUTPUT 2	OUTPUT 3	OUTPUT 4
Average CPU time	4.8	4.9	5.0	4.9
Average number of iteration steps	3.8	4.0	4.4	3.9

Observe that **inverse iteration is slightly more efficient than partial TLS**. Similar results are obtained with other test structures obtained from LMS [2].

Instead of solving 512 x 4 one-dimensional TLS problems, we consider now the **computation of the FRF's at all output locations simultaneously**. Hereto, 512 multidimensional TLS problems (10.78) with 2 inputs (n = 2) and 4 output (d = 4) have to be solved. Using the classical and partial TLS algorithm, the following average CPU times are obtained for computing the response at the 4 outputs (similar to fig. 10.3, 10.4, 10.5, and 10.6) simultaneously :

	PTLS	classical TLS
Bidiagonalization	20.4	20.4
Initialization V		7.5
Diagonalization	2.4	30.4
Back transformation	5.0	
TLS solution	3.9	3.9
TOTAL	31.7	62.2

Observe that now the total CPU time with PTLS is much larger than the sum of the CPU times for calculating the FRF's at the outputs separately with PTLS. This has been discussed in section 10.8.2. Since the dimension of the matrix corresponding to the 4-dimensional TLS problem is increased by a factor $(n+d)/(n+1) = 2$ w.r.t. the one-dimensional case, the computational cost may increase with a factor $2^3 = 8$! Hence, the efficiency of the PTLS algorithm can not be improved by computing the FRF's of all outputs simultaneously in one d-dimensional TLS problem unless the number d of outputs is much smaller than the number n of inputs. Since d > n in the example considered here, we can never expect an improvement in efficiency.

Observe also that in this example, the diagonalization cost with PTLS is very small. This is because mostly, the bidiagonal (9.2) obtained by Householder transformations from the 12 by 12 real matrix X_R in (10.83), has a very small ($< \varepsilon$) superdiagonal element e_9. Hence, a 4 by 4 subbidiagonal block is immediately isolated. Since its singular values coincide with the 4 desired smallest singular values of the complex matrix, we can immediately compute the TLS solution from the base vectors

associated with this block, without performing any QR/QL iteration steps. However, this is not the case in general. More refinements must be incorporated in the PTLS algorithm in order to solve any possible complex multidimensional TLS problem in the most reliable and efficient way. Hereto, we better perform all operations in the complex field. This sort of problems has not yet been fully analyzed and requires further research.

10.9. CONCLUDING REMARKS

In this chapter, **four iterative methods** are discussed and **evaluated w.r.t. their computational efficiency in solving TLS problems**. The **Chebyshev iteration** (section 10.3), the **inverse iteration** (section 10.4) and the **Rayleigh quotient iteration** method (section 10.5) are essentially based on the construction of an appropriately chosen sequence of functions $f_k(S)$ such that the matrix-vector product $f_k(S)q_1$, where q_1 is a start vector containing the initial information, converges to the desired singular vector of the matrix S (section 10.2). The **Lanczos method** (section 10.6) however, does not compute the desired singular triplets of a matrix directly but instead, tri- or bidiagonalizes the matrix. Hereto, a sequence of tri- of bidiagonals of growing dimension is generated whose extremal singular values converge to the extremal singular values of the original matrix. Since these extremal values can emerge long before the tri- or bidiagonalization is complete, Lanczos methods can be used as an alternative, possibly more efficient technique than Householder bidiagonalization in situations where only a few of the largest and/or smallest singular values of a matrix are desired. The associated singular vectors are then easily obtained using the QR algorithm or inverse iteration.

By **analyzing the convergence properties of these iterative methods**, it is shown that **inverse iteration is the most promising iterative technique for solving generic, one-dimensional TLS problems with unique solution**, since its convergence rate depends mainly on the ratio between the singular values corresponding to the desired and undesired singular vectors (section 10.4.2). For most TLS problems, this ratio is high,

thereby assuring fast convergence. Hence, RQ iteration which requires more computations in each iteration step, will not perform better. Moreover, the initial guess vector in the RQ iteration must be sufficiently accurate in order to guarantee convergence to the desired singular vector from which the TLS solution can be obtained. Furthermore, since the convergence rate of the **Chebyshev iteration and Lanczos methods** also depends on the relative spread of the squared singular value spectrum of the matrix, their applicability w.r.t. the TLS computation **is restricted to TLS problems, characterized by a very dense singular value spectrum.** Moreover, roundoff errors make the Lanczos methods difficult to use in practice. To maintain their stability and reliability, many additional computations are required which reduce further their efficiency.

Based on the convergence rate and number of operations in each iteration step, the efficiency of all iterative methods described here, can be compared with that of the direct computation methods : classical and partial SVD (section 10.7). It is concluded that **iterative methods are most efficient when only one singular vector is to be computed, their convergence rate is sufficiently high, the desired accuracy is rather low and a good initial guess vector is available. Iterative methods are therefore particularly attractive in solving adaptive TLS problems** since then the solution of a previous set is mostly a good initial guess for the solution of the next set. A real life problem concerning the estimation of frequency response functions is discussed in section 10.8.

All iterative methods described above can be extended to compute several singular triplets or a basis of a singular subspace. These extensions however, reduce the efficiency. Moreover, one **needs to know the dimension of the desired subspace** or equivalently, the number of desired vectors. This **is not the case for the direct computation methods.** Since moreover **their efficiency is less problem dependent**, they have a wider potential use and **are mostly preferred** to iterative methods **in moderately sized problems.**

CHAPTER 11

CONCLUSIONS

The research reported in this thesis is clearly stimulated by the need for a **more global approach to the problem of linear parameter estimation** in case all variables that characterize the system, can not be observed without error. Such problems arise in a broad class of scientific disciplines such as signal processing, automatic control, system identification and in general engineering, statistics, physics, economics, biology, medicine, ... The classical least squares (LS) approach which assumes that all but one variable can be measured without error, is no longer appropriate. New fitting techniques must be devised to compensate for data errors. **The total least squares (TLS) approach** analyzed throughout this thesis, **is such a method of fitting** that is appropriate **when errors occur in all data** and amounts to fitting a best subspace to the data. It was introduced into the field of numerical analysis by Golub and Van Loan and based on the powerful singular value decomposition (SVD). Because of its nice numerical and geometrical properties, we are encouraged to study **the practical applicability of the TLS technique** in different problems of system identification, linear modelling, ... and evaluate its significance w.r.t. the current estimation techniques in each domain. This is the **main objective** of this thesis. In the following section we review the obtained results. This evaluation allows to clarify the specific contents of the main title of this thesis. In the final section a number of suggestions for further research related to the obtained results are made.

11.1. REVIEW OF THE OBTAINED RESULTS

Every linear parameter estimation problem gives rise to an overdetermined set of linear equations AX = B. Whenever both the data matrix A and observation matrix B are inaccurate, the TLS technique is appropriate

for solving this set to obtain the parameters X. In the first 5 chapters of this thesis, different aspects of the TLS technique are analyzed.

In **chapter 1**, we first present the main principle and generic **solution of the one-dimensional TLS problem** (with one observation vector) and compare it geometrically with the classical LS method. Next, **we extend this problem and show how to compute** the TLS solution in case of more than one observation vector and in case some of the columns of the data matrix A have no error. Furthermore, it is shown how TLS singles out the minimum norm solution when the solution of the TLS problem is not unique. Moreover we **generalize the TLS computations to all cases in which the algorithm of Golub and Van Loan fails** to produce a solution and prove that the proposed generalization is still optimal w.r.t. the TLS criteria. Finally, the TLS computation is summarized in one algorithm which includes all the extensions described above.

In **chapter 2**, interesting **algebraic relationships between the TLS and classical LS approach** are proven w.r.t. their solution, residuals, approximation efforts and associated subspaces. In particular, it is shown how the differences between both approaches increase when the equations AX = B are less compatible, when the length of B or X is growing or when A tends to be rank-deficient. They are maximal when B is parallel with the singular vector of A, associated with its smallest singular value. Furthermore, it is shown how TLS is led to a weighted LS problem and assumptions about the underlying perturbation model of both techniques are deduced. They show that many perturbation models correspond with the same TLS solution.

All these properties are useful to get more insight into the **sensitivity** of both techniques **w.r.t. perturbations** on the data, discussed in **chapter 3**. Hereto we evaluate the relative changes in the solutions and related subspaces of the perturbed problem w.r.t. the unperturbed one. First of all, the **worst perturbation effects on their solutions** are investigated, using the condition number. For zero residual problems (which correspond with an unperturbed compatible set of equations), it is shown that the TLS and LS solution are maximally affected under the same conditions of perturbation but for nonzero residual problems, the **perturbation effects on the TLS solution may be worse.** However these

worst perturbation effects are only **obtained for a special choice** of the perturbations of the data and of the orientation of the observation vector. Hence, they are quite unrealistic for other circumstances.
Second, the **sensitivity properties of the singular subspaces related to TLS and LS**, are investigated. Subspace perturbations which maximally affect the accuracy of the TLS problem, are computed. Their effect on the TLS solution accuracy is derived from the one-dimensional TLS problem. Using the perturbation results of the SVD, it is shown that the singular subspaces related to TLS are **less noise sensitive** than those related to LS. This is the case for any zero residual problem and for sufficiently compatible nonzero residual problems. We moreover prove and confirm by simulations that this superiority of TLS w.r.t. LS is more pronounced if the number of observation vectors and their length is increasing but especially if they are oriented along the singular vectors of the data matrix A, associated with its smallest singular values.

If now some information about the error distribution on the data is available, **statistical properties** can be derived for the expected accuracy of the TLS problem (**chapter 4**). It is assumed throughout this chapter that the errors on the data [A;B] are rowwise independently and identically distributed with zero mean and common covariance matrix of the form σ_v^2 I (I is the identity matrix). Under these fairly reasonable assumptions, the TLS solution is shown to be **a strongly consistent estimator of the true** but unknown **parameters of the unperturbed model**. Also strongly consistent estimators for the intercept, if any, of a linear relation present in the data, and for the error variance σ_v^2 are deduced from the TLS results. If the common error distribution has finite fourth moments, all these estimators are proven to be asymptotically normally distributed. Finally, in case the error covariance matrix equals $\sigma_v^2 \Sigma_o$ with Σ_o known and positive definite, it is shown how TLS can still compute strongly consistent estimates by an appropriate transformation of the data. All these asymptotic properties are fairly well approximated in moderately sized problems, as confirmed by simulations.

In **chapter 5, algebraic equivalences and interrelations between the TLS estimator and the classical regression estimators** LS, principal component (PC), ridge regression (RR) and latent root regression (LR) are outlined. The SVD and geometric concepts are used to elucidate their relationships w.r.t. parameter estimation in linear multicollinearity problems. The equivalence between PC and LR in the presence of exactly defined multicollinearities is proven and the differences between LR and TLS are clarified.

In the next three chapters the practical applicability of TLS is proven in a variety of problems.

Chapter 6 studies **the subset selection problem in case all variables involved contain errors.** A new subset selection algorithm SAB-TLS, based on TLS, is presented. Its selection properties and accuracy in parameter estimation and prediction are evaluated w.r.t. the subset selection algorithm SA-LS of Golub, Klema and Stewart. SA-LS should only be used if the variables in the data matrix A are error-free. If not, the r parameters in any selected subset model of r variables should be computed with TLS. If the perturbations of the model Ax = b are sufficiently small, subset selection based on the r-th right singular subspace of A enables a slightly better prediction accuracy in case of bad subset condition. In all other cases, namely in case of well-conditioned or highly perturbed datasets, subset selection based on the r-th right singular subspace of [A;b] is **superior in prediction** accuracy (SAB-TLS). By taking into account the **extra information** provided by the observation vector b, SAB-TLS computes the best subset and moreover **guarantees** a higher and longer **stability** and **reliability of its subset choice.** A computationally attractive variant of SAB-TLS with the same properties, is also derived.

In **chapter 7**, the **applicability of TLS in the estimation of the impulse response of a system by discrete deconvolution** is studied. First, the advantages of TLS w.r.t. LS are illustrated in a simulation example. If the data are perturbed by stationary, zero mean white noise, TLS computes a strongly consistent estimate of the true impulse response, corresponding to the unperturbed system. Next, it is shown how the **TLS technique can be efficiently used in the computation of the renal retention function**, by deconvolving the activity-time curves obtained from the blood and the kidney. Results from simulations, as well as from clinical data, show the advantages of the use of TLS over the current deconvolution techniques which are based on back substitution or Gauss elimination using the smoothed curves. It is concluded that TLS needs

no smoothings and is a powerful, efficient, reliable and stable deconvolution technique.

In **chapter 8**, the **applicability of TLS in the estimation of the parameters of a transfer function model** is discussed and **compared with** the most currently used **instrumental variable (IV) techniques**. The relationships between TLS and the IV methods are established. Like IV, TLS is able to compute strongly consistent estimates but only if the input-output data are obscured by discrete stationary, zero mean **white noise** (IV allows arbitrary noise models). Hence, TLS is more **restrictive** in use but much **simpler** since no IV are to be generated. This is often a crucial problem when using IV methods. Unfortunately, the TLS solution involves **off-line** computations and is thus computationally not so attractive when large sample lengths were used. Finally, the accuracy of both methods w.r.t. parameter estimation in a transfer function model is compared by simulations. It is concluded that TLS outperforms the IV methods when **short sample lengths** are used and when the outputs and possibly the inputs are obscured by stationary, **zero mean white noise**. Mostly, the TLS method is superior in terms of both bias and variance and gives estimates with smaller mean squared error than the ordinary IV methods. The mean squared error becomes comparable by extending the IV sufficiently. The superiority of TLS is particularly clear in case the **zeros** of the polynomial operating on the outputs, **approach the unit circle** or both the inputs and the outputs are noisy. In other cases, TLS and IV methods give quite similar results.

Finally, the **last 2 chapters** show **how to improve the efficiency** of the TLS computations. Since the TLS solution is deduced from **only one** right singular vector or in general, **a basis** of the right singular subspace associated with the smallest singular values of the data [A;B], its computational cost can be considerably reduced by computing the SVD only partially. Hereto, an **improved partial SVD algorithm PSVD** is presented in **chapter 9, computing the singular subspace of a matrix, associated with its smallest singular values**. By analyzing the operation counts and presenting computational results, its relative efficiency w.r.t. classical SVD is demonstrated. Depending on the gap, the desired numerical accuracy and the dimension of the desired subspace, **PSVD can be 3 times faster** than the classical SVD algorithm, while the same accuracy can be

maintained. Based on PSVD, a very efficient and reliable algorithm for solving nonhomogeneous linear equations is also derived. By incorporating the PSVD algorithm into the classical TLS computations, an improved TLS algorithm PTLS is obtained. Its **higher efficiency** w.r.t. the classical TLS algorithm is demonstrated and confirmed with computational results. Typically, **PTLS reduces the computation time with a factor 2.**

If a priori information about the solution of the problem to be solved **is available, the TLS computations can be speeded up in an iterative way.** Four iterative methods, namely Chebyshev iteration, inverse iteration, Rayleigh quotient (RQ) iteration and the Lanczos method are discussed in **chapter 10** and compared w.r.t. their computational efficiency in TLS computations. By analyzing the convergence properties of these methods, it is shown that **inverse iteration is the most promising iterative technique** for solving generic, one-dimensional TLS problems with unique solution, since its convergence rate depends mainly on the ratio between the singular values, corresponding to the desired and undesired singular vectors. For most TLS problems, this ratio is high. This assures fast convergence so that RQ iteration will not perform better. Since the convergence rate of the Chebyshev iteration and Lanczos method also depends on the relative spread of the squared singular value spectrum of the matrix, their applicability in the TLS computations is restricted to problems with a very dense singular value spectrum. Moreover, roundoff errors make the Lanczos methods difficult to use in practice. Based on the convergence rate and the number of operations in each iteration step, the efficiency of all iterative methods is compared with that of the direct computation methods : classical and partial SVD. It is concluded that iterative methods are **most efficient when only one singular vector** is to be computed, their **convergence** rate is sufficiently **high**, the desired accuracy is rather low and a good initial guess vector is available. Iterative methods are therefore particularly attractive in solving slowly varying TLS problems because then, the solution of a previous set is mostly a good initial guess for the solution of the next set. A **real life** problem concerning the estimation of frequency response functions is also discussed. It illustrates the efficiency of inverse iteration and the PTLS algorithm w.r.t. the classical TLS computations. Because the direct computation methods do not require knowledge of the dimension of the desired singular subspace and because

their efficiency is less problem dependent, they have a **wider potential use** and are mostly preferred to iterative methods in **moderately sized** problems.

11.2. SUGGESTIONS FOR FUTURE RESEARCH

Although the entire thesis is devoted to TLS, the analysis and results presented in this thesis do not claim to have solved all problems associated to the TLS technique and explored all possible applications. Further research concerning the following topics is suggested :

- Experiments show that **the TLS technique is sensitive to the scaling of the data.** In order to attain the optimal TLS solution properties (see chapter 4), data must be properly scaled such that the errors on the data are independently and identically distributed with zero mean and common covariance matrix of the form $\sigma_v^2 I$ (I is the identity matrix). Some scaling recommendations are given in section 1.8.1. Not only knowledge of the error distribution is required for computing the appropriate scaling factors but it is not always possible to equilibrate the error matrix appropriately by row and column scaling, e.g. in case of a nonlinear error distribution. Although in the latter case the optimal properties of the TLS solution are not obtained, they can be fairly well approximated and still motivate the use of TLS, as we could observe in some practical applications. Nevertheless, a **more fundamental study of the scaling problem resulting in an adequate scaling strategy**, would certainly enhance the applicability of TLS and convince most criticizers.

- The TLS technique requires a **reliable rank determinator** for computing the correct rank of the data matrix $[A;B]$ (step 2 of algorithm 1-1) and for checking (non)singularity of the matrix Γ (step 4 of algorithm 1-1). If the rank is unknown, the SVD is the most reliable tool to treat rank deficiency. It allows to compute the rank from the number of singular values beyond a certain error-dependent bound. The derivation of this bound in the TLS algorithms presented in this thesis, is based on the maximal minimal signal-to-noise ratio (see section 1.8.1) but other strategies are possible. **A deeper insight into the effects of noise and its distribution onto the SVD of a matrix** is certainly required. It enables the design of more refined rank determinators for computing the correct rank of the data $[A;B]$ and the multiplicity of its singular values. Moreover, it allows to derive reliable and robust rules for checking (non)singularity of matrices in function of the characteristics of the perturbations on the data. This test is required in the last step of the TLS algorithm before computing the TLS solution from the right singular vector(s) of the data.

- **TLS destroys the structure of a matrix.** Indeed, the TLS approximation $[\hat{A};\hat{B}]$ of the data $[A;B]$ with minimal approximation effort, is no longer structured as $[A;B]$. For instance, the Vandermonde structure in polynomial fitting problems is destroyed by TLS. Also, many problems in system identification loose their Toeplitz or Hankel structure when using TLS. Further research is thus needed **to extend the TLS criteria when the problem structure should be preserved** in the computed TLS approximation $[\hat{A};\hat{B}]$.

- The higher efficiency of the partial SVD and TLS algorithm (see chapter 9), is obtained by modifying the classical SVD computations in three ways : by delaying the initialization phase (section 9.3.1), by diagonalizing the bidiagonal only partially (section 9.3.2) and third, by an appropriate choice between QR and QL iteration steps (section 9.3.3). **The same modifications can now be applied to the generalized SVD algorithm** [80] **or the (generalized) symmetric eigenvalue decomposition algorithm** in so far as they are based on the QR algorithm [48, section 8.2 - 8.6]. They improve the efficiency of those algorithms when only a basis of the singular or eigen subspace associated with the smallest generalized singular values or (generalized) eigenvalues, is required. Also, only minor modifications in the improved algorithm are required to compute a basis of the subspace of a matrix associated with its **largest** (generalized) singular values or eigenvalues.

- Another suggestion for further work is **the transposition of the TLS computations into the complex field.** This would certainly improve the efficiency of the TLS computations on complex data compared to the computations into the real domain after the appropriate transform-

ations, e.g. in the estimation of frequency response functions for experimental, structural, dynamical analysis (see section 10.8). Simultaneously, it enhances the applicability of TLS in other problems dealing with complex data.

- Furthermore, the TLS algorithms presented in this thesis, do not yield good on-line computation schemes, e.g. in recursive form. These schemes enable to update the solution at a low computational cost as soon as new data are coming in. Indeed, appending one row to the matrix [A;B] requires to start the TLS computations from scratch by treating [A;B] as a matrix about which we know nothing. The development of an **efficient on-line version** of the TLS computations would certainly open new domains for TLS, e.g. in signal processing.

- Finally, the study of the use of TLS in subset selection (chapter 6), system parameter identification (chapter 7 and 8) and renography (section 7.3) shows the advantages in reliability, accuracy, efficiency, ... of the TLS method analyzed in this thesis, and confirms its practical applicability. Also in other interesting problems of modal analysis [91], inverse scattering [99] and frequency estimation [85], the applicability of the TLS technique is proven. Nevertheless, there still remain a lot of **unexplored problems where TLS should be successfully applied** and improve the quality of the results. It is hoped that this work will aid and **stimulate any reader** of it **to apply TLS** in his own applications and problems.

REFERENCES

[1] J.O. AASEN, "On the reduction of a symmetric matrix to tridiagonal form", BIT, Vol. 11, pp. 233-242, 1971.

[2] M. ADAMS and R. BUCKINCX, "Een efficiënte en betrouwbare bereke-ningsmethode voor de totale kleinste kwadratenbenadering", Eng. Thesis, Dept. of Electr. Eng., K.U.Leuven, June 1986.

[3] R.J. ADCOCK, "A problem in least squares", The Analyst, Vol. 5, pp. 53-54, 1878.

[4] R. ALLEMANG, "Investigation of some multiple input/output frequency response function modal analysis techniques", Ph. D. Dissertation, Dept. of Mech. & Ind. Eng., University of Cincinnati, 1980.

[5] R. ALLEMANG, R. ROST and D. BROWN, "Dual input estimation of frequency response functions for experimental modal analysis of automotive structures", paper No. 820193 of the Society of Automotive Engineers (SAE), 25 pp.,1982.

[6] R. ALLEMANG, R. ZIMMERMAN and D. BROWN, "Techniques for reducing noise in frequency response measurements", presented in session on Application of System Identification techniques, Winter Annual Meeting of the Amer. Society of Mech. Engineers (ASME), 1979.

[7] T.W. ANDERSON, "The 1982 Wald memorial lectures estimating linear statistical relationships", Ann. Statist., Vol. 12, No. 1, pp. 1-45, 1984.

[8] K.J. ASTRÖM, "Introduction to stochastic control theory", Academic, New York, 1970.

[9] K.J. ASTRÖM and P. EYCKHOFF, "System identification, a survey", Automatica, Vol. 7, pp. 123-162, 1971.

[10] D.C. BARBER and K.S. NIJRAN, "Factor analysis of dynamic radionuclide studies", Proc. 3rd World Congr. Nucl. Med. and Biol., Paris, pp. 31-34, 1982.

[11] V. BARWELL and A. GEORGE, "A comparison of algorithms for solving symmetric indefinite systems of linear equations", ACM Trans. Math. Software, Vol. 2, pp. 242-251, 1976.

[12] F.L. BAUER, "Optimally scaled matrices", Numer. Math., Vol. 5, pp. 73-87, 1963.

[13] C. BONIVENTO and R. GUIDORZI, "Parametric identification of linear multivariable systems", Proc. Joint Automatic Contr. Conf. (JACC), St.-Louis, paper No. 5-C6, pp. 342-347, 1971.

[14] H. BOWDLER, R.S. MARTIN, C. REINSCH and J.H. WILKINSON, "The QR and QL algorithms for symmetric matrices", Numer. Math., Vol. 11, pp. 293-306, 1968.

[15] D. BROWN, G. CARBON and K. RAMSEY, "Survey of excitation techniques applicable to the testing of automotive structures", paper No. 770029 of the Amer. Society of Automotive Engineers (SAE), 20 pp., 1977.

[16] J.R. BUNCH, "Analysis of the diagonal pivoting method", SIAM J. Numer. Anal., Vol. 8, pp. 656-680, Dec. 1971.

[17] J.R. BUNCH and L. KAUFMAN, "Some stable methods for calculating inertia and solving symmetric linear systems", Math. Comput., Vol. 31, pp. 163-179, Jan. 1977.

[18] J.R. BUNCH and B.N. PARLETT, "Direct methods for solving symmetric indefinite systems of linear equations", SIAm J. Numer. Anal., Vol. 8, pp. 639-655, Dec. 1971.

[19] T.F. CHAN, "An improved algorithm for computing the singular value decomposition", ACM Trans. Math. Software, Vol. 8, pp. 72-83, 1982.

[20] K.L. CHUNG, "A course in probability theory", Harcourt, Brace and World, New York, 1968.

[21] J. CULLUM and R.A. WILLOUGHBY, "Lanczos and the computation in specified intervals of the spectrum of large, sparse, real symmetric matrices", in Sparse Matrix Proc., 1978, ed. I.S. Duff and G.W. Stewart, SIAM Publications, Philadelphia, 1979.

[22] C. DAVIS and W.M. KAHAN, "The rotation of eigenvectors by a perturbation III", SIAM J. Numer. Anal., Vol. 7, No. 1, pp. 1-46, 1970.

[23] B. DE MOOR, IWONL-Report, Dept. of Electr. Eng., K.U.Leuven, 1984.

[24] B. DE MOOR and J. VANDEWALLE, "An adaptive singular value decomposition algorithm based on generalized Chebyshev recursions", Proc. Conf. on Mathematics in Signal Processing (in press), Bath, U.K., Sep. 21-23, 1985.

[25] B. DE MOOR and J. VANDEWALLE, "A geometrical approach to the maximal corank problem in the analysis of linear relations from noisy data", Proc. 25th IEEE Conf. on Decision and Control, Athens, Dec. 7-10, pp. 1990-1995, 1986.

[26] B. DE MOOR and J. VANDEWALLE, "A geometrical strategy for the identification of state space models of linear multivariable systems with singular value decomposition", Proc. 3rd Int. Symposium on Application of Multivariable System Techniques (in press), Plymouth, U.K., April 13-15, 1987.

[27] B.L. DIFFEY and J.R. CORFIELD, "The data bounding technique in discrete deconvolution", Med. & Biol. Eng., Vol. 14, pp. 478, 1976.

[28] B.L. DIFFEY, F.M. HALL and J.R. CORFIELD, "The ^{99m}Tc-DTPA dynamic renal scan with deconvolution analysis", J. Nucl. Med., Vol. 17, pp. 352-355, 1976.

[29] J.J. DONGARRA, J.R. BUNCH, C.B. MOLER and G.W. STEWART, "LINPACK user's guide", SIAM, Philadelphia, 1979.

[30] N.R. DRAPER and H. SMITH, "Applied regression analysis", 2nd ed., John Wiley, New York, 1981.

[31] G. ECKART and G. YOUNG, "The approximation of one matrix by another of lower rank", Psychometrica, Vol. 1, pp. 211-218, 1936.

[32] P. EYCKHOFF, "System identification", John Wiley, London, 1974.

[33] J.S. FLEMING, "Measurement of Hippuran plasma clearance using a gamma camera", Phys. Med. Biol., Vol. 22, pp. 526-530, 1977.

[34] J.S. FLEMING and B.A. GODDARD, "A technique for the deconvolution of the renogram", Phys. Med. Biol., Vol. 19, pp. 546-549, 1974.

[35] J.S. FLEMING and R.W. KENNY, "A comparison of techniques for the filtering of noise in the renogram", Phys. Med. Biol., Vol. 22, pp. 359-364, 1977.

[36] B. FRIEDLANDER, "The overdetermined recursive instrumental variable method", IEEE Trans. Automat. Contr., Vol. AC-29, No. 4, April 1984.

[37] B. FRIEDLANDER, P. STOICA and T. SÖDERSTRÖM, "Instrumental variable methods for ARMA parameter estimation". Preprints 7th IFAC/IFORS Symposium on Identification and System Parameter Estimation, York, U.K., July 3-7, Vol. 1, pp. 29-36, 1985.

[38] F.R. GANTMACHER, "The theory of matrices", Chelsea Publishing Company, New York, 1959.

[39] M. GEVERS and B. BASTIN, "What does system identification have to offer ?", Proc. 6th IFAC Symposium on Identification and System Parameter Estimation, Tutorial Workshop on System Identification, Washington D.C., June 1982.

[40] L.J. GLESER, "Calculation and simulation in errors-in-variables regression problems", Mimeograph Series 78-5, Statistics Dept., Purdue University, 1978.

[41] L.J. GLESER, "Estimation in a multivariate "errors in variables" regression model : Large sample results", Ann. Statist., Vol. 9, No. 1, pp. 24-44, 1981.

[42] L.J. GLESER and G.S. WATSON, "Estimation of a linear transformation", Biometrika, Vol. 60, No. 3, pp. 525-534, 1973.

[43] G.H. GOLUB, Personal letter, Nov. 30, 1983.

[44] G.H. GOLUB, V. KLEMA and G.W. STEWART, "Rank degeneracy and least squares problems", Technical Report TR-456, Dept. of Computer Science, University of Maryland, College Park, 1976.

[45] G.H. GOLUB and C. REINSCH, "Singular value decomposition and least squares solutions", Numer. Math., Vol. 14, pp. 403-420, 1970.

[46] G.H. GOLUB and G.W. STEWART, "A generalization of the Eckart-Young approximation theorem", Manuscript NA-86-35, Numer. Anal. Project, Comput. Science Dept., Stanford University, March 1986.

[47] G.H. GOLUB and C.F. VAN LOAN, "An analysis of the total least squares problem", SIAM, J. Numer. Anal., Vol. 17, No. 6, pp. 883-893, Dec. 1980.

[48] G.H. GOLUB and C.F. VAN LOAN, "Matrix computations", The John Hopkins University Press, Baltimore, Maryland, 1983.

[49] R.F. GUNST and R.L. MASON, "Biased estimation in regression : An evaluation using the mean squared error", J. Amer. Statist. Soc., Vol. 72, No. 359, pp. 616-628, 1977.

[50] R.F. GUNST and R.L. MASON, "Regression analysis and its application : a data-oriented approach", Marcel Dekker Inc., New York, 1980.

[51] R.F. GUNST, J.T. WEBSTER and R.L. MASON, "A comparison of least squares and latent root regression estimators", Technometrics, Vol. 18, No. 1, pp. 75-83, 1976.

[52] P.C. HANSEN, "SVD-theory and applications", Report No. 84-05, Technical Highschool, Lyngby, Denmark, Nov. 1984.

[53] D.M. HAWKINS, "On the investigation of alternative regressions by principal component analysis", Appl. Statist., Vol. 22, pp. 275-286, 1973.

[54] R.R. HOCKING, "The analysis and selection of variables in linear regression", Biometrics, Vol. 32, pp. 1-49, March 1976.

[55] R.R. HOCKING, "Developments in linear regression methodology 1959-1982", Technometrics, Vol. 25, No. 3, pp. 219-230, Aug. 1983.

[56] A.E. HOERL and R.W. KENNARD, "Ridge regression - 1980. Advances, algorithms and applications", Amer. J. Math. and Manag. Sciences, Vol. 1, No. 1, pp. 5-83, 1981.

[57] Y.S. HUNG and A.G.J. MacFARLANE, "Multivariate feedback : A quasi-classical approach", Lecture Notes in Control and Information Sciences 40, Springer-Verlag, New York, 1982.

[58] A.J. JAKEMAN and P.C. YOUNG, "Refined instrumental variable methods of recursive time-series analysis, Part II : multivariable systems", Int. J. Control, Vol. 29, pp. 621-6444, No. 4, 1979.

[59] L.S. JENNINGS and M.R. OSBORNE, "A direct error analysis for least squares", Numer. Math., Vol. 22, pp. 322-332, 1974.

[60] R.E. KALMAN, "System identification from noisy data", in Dynamical Systems II, ed. A.R. Bednarek and L. Cesari, Academic Press, New York, pp. 135-164, 1982.

[61] T. KATO, "Perturbation theory for linear operators", Springer, New York, 1966.

[62] M.G. KENDALL and A. STUART, "The advanced theory of statistics", 4th ed., Vol. 2, Griffin, London, 1979.

[63] R.W. KENNY, D.M. ACKERY, J.S. FLEMING, B.A. GODDARD and R.W. GRANT, "Deconvolution analysis of the scintillation camera renogram", Br. J. Radiol., Vol. 48, pp. 481-486, 1975.

[64] R.H. KETELLAPPER, "The relevance of large-sample properties of estimators for the errors-in-variables model : A Monte Carlo study", Comm. Statist., Ser. B, Vol. 11, No. 5, pp. 625-634, 1982.

[65] R.H. KETELLAPPER, "On estimating parameters in a simple linear errors-in-variables model", Technometrics, Vol. 25, No. 1, pp. 43-47, 1983.

[66] T.C. KOOPMANS, "Linear regression analysis of economic time series", DeErven F. Bohn, N.V. Haarlem, The Netherlands, 1937.

[67] C.L. LAWSON and R.J. HANSEN, "Solving least squares problems", Prentice-Hall, Englewood Cliffs, New York, 1974.

[68] J. LEURIDAN, D. DE VIS, H. VAN DER AUWERAER and F. LEMBREGTS, "A comparison of some frequency response function measurement techniques", Proc. 4th Int. Modal Analysis Conference, Los Angeles, Feb. 3-6, pp. 908-918, 1986.

[69] M.J. LEVIN, "Estimation of a system pulse transfer function in the presence of noise", IEEE Trans. Automat. Contr., Vol. AC-9, No. 3, pp. 229-235, July 1964.

[70] J. LEWIS, "Algorithms for sparse matrix eigenvalue problems", Technical Report STAN-CS-77-595, Dept. of Computer Science, Stanford University, 1977.

[71] L. LJUNG, "Aspects on the system identification problem", Signal Processing, Vol. 4, pp. 445-456, 1982.

[72] L. LJUNG, "Identification criteria and their asymptotic properties", Proc. 6th IFAC Symposium on Identification and System Parameter Estimation, Tutorial Workshop on System Identification, Washington D.C., June 1982.

[73] A. MADANSKY, "The fitting of straight lines when both variables are subject to error", J. Amer. Statist. Assoc., Vol. 54, pp. 173-205, 1959.

[74] J. MANDEL, "Use of the singular value decomposition in regression analysis", The Amer. Statist., Vol. 36, No. 1, pp. 15-24, Feb. 1982.

[75] P.A.P. MORAN, "Estimating structural and functional relationships", J. Multivariate Anal., Vol. 1, pp. 232-255, 1971.

[76] P.H. O'REILLY, R.A. SHIELDS and H.J. TESTA, "Nuclear medicine in urology and nephrology", Butterworth, London, 1979.

[77] C.C. PAIGE, "The computation of eigenvalues and eigenvectors of very large sparse matrices", Ph. D. Thesis, University of London, 1971.

[78] C.C. PAIGE, "Error analysis of the Lanczos algorithm for tridiagonalizing a symmetric matrix", J. Inst. Math. Appl., Vol. 18, pp. 341-349, 1976.

[79] C.C. PAIGE, "Accuracy and effectiveness of the Lanczos algorithm for the symmetric eigenproblem", Lin. Alg. & Its Applic., Vol. 34, pp. 235-258, 1980.

[80] C.C. PAIGE and M.A. SAUNDERS, "Towards a generalized singular value decomposition", SIAM J. Numer. Anal., Vol. 18, No. 3, pp. 398-405, June 1981.

[81] B.N. PARLETT, "The Rayleigh quotient iteration and some generalizations for nonormal matrices", Math. Comput., Vol. 28, pp. 679-693, 1974.

[82] B.N. PARLETT, "The symmetric eigenvalue problem", Prentice-Hall, Englewood Cliffs, New York, 1980.

[83] K. PEARSON, "On lines and planes of closest fit to points in space", Philos. Mag., Vol. 2, pp. 559-572, 1901.

[84] G. PETERS and J.H. WILKINSON, "Inverse iteration, ill-conditioned equations and Newton's method", SIAM Rev., Vol. 21, No. 3, 339-360, July 1979.

[85] M.A. RAHMAN and K.B. YU, "Improved frequency estimation using total least squares approach", Proc. Int. Conf. on acoust., speech and signal proc. (ICASSP), Tokyo April 7-11, Paper No. 27.14, 1986.

[86] J. REEVE and J.C.W. CRAWLEY, "Quantitative radioisotope renography : the derivation of physiological data by deconvolution analysis using a single-injection technique", Clin. Science & Mol. Med., Vol. 47, pp. 317-330, 1974.

[87] O. REIERSOL, "Confluence analysis by means of lag moments and other methods of confluence analysis", Econometrica, Vol. 9, pp. 1-23, 1941.

[88] P.M. REILLY and H. PATINO-LEAL, "A Bayesian study of the errors-in-variables models", Technometrics, Vol. 23, No. 3, pp. 221-231, 1981.

[89] C.H. REINSCH, "Smoothing by spline functions", Numer. Math., Vol. 10, pp. 177-183, 1967.

[90] T.J. RIVLIN, "The Chebyshev polynomials", John Wiley, New York, 1976.

[91] R. ROST and J. LEURIDAN, "A comparison of least squares and total least squares for multiple input estimation of frequency response functions", ASME Paper No. 85-DET-105 presented at the ASME Design Engineering Division Conference and Exhibit on Mechanical Vibration and Noise, Cincinnati, Ohio - Sept. 10-13, 1985.

[92] H. RUTISHAUSER, "Computation aspects of F.L. Bauer's simultaneous iteration method", Numer. Math., Vol. 13, pp. 4-13, 1969.

[93] Y. SAAD, "Chebyshev acceleration techniques for solving nonsymmetric eigenvalue problems", Math. Comput., Vol. 42, No. 166, pp. 567-588, 1984.

[94] A. SAVITSKY and J.R. GOLAY, "Smoothing and differentiation of data by simplified least squares procedures", Anal. Chem., Vol. 36, pp. 1627-1639, 1964.

[95] P. SCHMIDT, "Econometrics", Marcel Dekker Inc., New York, 1976.

[96] H. SCHNEEWEISS, "Consistent estimation of a regression with errors in the variables", Metrika, Vol. 23, pp. 101-115, 1976.

[97] H. SCHWETLICK and V. TILLER, "Numerical methods for estimating parameters in nonlinear models with errors in the variables", Technometrics, Vol. 27, No. 1, pp. 17-24, 1985.

[98] D.S. SCOTT, "Analysis of the symmetric Lanczos process", UCB-ERL Technical Report M78/40, University of California, Berkeley, 1978.

[99] M.T. SILVIA and E.C. TACKER, "Regularization of Marchenko's integral equation by total least squares", J. Acoust. Soc. Amer., Vol. 72, pp. 1201-1207, 1982.

[100] B.T. SMITH, J.M. BOYLE, Y. IKEBE, V.C. KLEMA and C.B. MOLER, "Matrix eigensystem routines : EISPACK guide", 2nd ed., Springer-Verlag, New York, 1970.

[101] F.W. SMITH and W.B. HILTON, "Monte Carlo evaluation of methods for pulse transfer function estimation", IEEE Trans. Automat. Contr., Vol. AC-12, No. 5, Oct. 1967.

[102] T. SÖDERSTRÖM and P. STOICA, "Comparison of some instrumental variable methods - consistency and accuracy aspects", Automatica, Vol. 17, pp. 101-115, 1981.

[103] T. SÖDERSTRÖM and P. STOICA, "Instrumental variable methods for system identification", Springer-Verlag (LNICS series), Berlin, 1983.

[104] P. SPRENT, "A generalized least-squares approach to linear functional relationships", J. Roy. Statist. Soc., Ser. B, Vol. 28, pp. 278-297, 1966.

[105] J. STAAR, "Concepts for reliable modelling of linear systems with application to on-line identification of multivariable state space descriptions", Doctoral Dissertation, Dept. of Electr. Eng., K.U. Leuven, June 1982.

[106] E.L. STIEFEL, "Kernel polynomials in linear algebra and their numerical application", Appl. Math. Ser. Nat. Bureau of Standards, Vol. 49, pp. 1-22, 1955.

[107] C.M. STEIN, "Multiple regression contributions to probability and statistics", in Essays in Honor of Harold Hotelling, ed. I. Olkin, Stanford University Press, pp. 424-443, 1960.

[108] G.W. STEWART, "Error and perturbation bounds for subspaces associated with certain eigenvalue problems", SIAM Rev., Vol. 15, No. 4, pp. 727-764, Oct. 1973.

[109] G.W. STEWART, "On the invariance of perturbed null vectors under column scaling", Numer. Math., Vol. 44, pp. 61-65, 1984.

[110] G.W. STEWART, "Rank degeneracy", SIAM J. Sci. Stat. Comput., Vol. 5, No. 2, pp. 403-413, June 1984.

[111] P. STOICA, B. FRIEDLANDER and T. SÖDERSTRÖM, "Least-squares, Yule-Walker, and overdetermined Yule-Walker estimation of AR parameters : a Monte Carlo analysis of finite-sample properties", Int. J. Control, Vol. 43, No. 1, pp. 13-27, 1986.

[112] P. STOICA and T. SÖDERSTRÖM, "Optimal instrumental variable estimation and approximate implementation", IEEE Trans. Automat. Contr., Vol. AC-28, pp. 757-772, 1983.

[113] P. STOICA and T. SÖDERSTRÖM, "Optimal instrumental-variable methods for identification of multivariable linear systems", Automatica, Vol. 19, pp. 425-429, 1983.

[114] M.L. THOMPSON, "Selection of variables in multiple regression : Part I. A review and evaluation", Int. Statist. Rev., Vol. 46, pp. 1-20, 1978.

[115] M.L. THOMPSON, "Selection of variable in multiple regression : Part II. Chosen procedures, computation and examples", Int. Statist. Rev., Vol. 46, pp. 129-146, 1978.

[116] M.E. VALENTINUZZI and E.M. MONTALDO VOLACHEC, "Discrete deconvolution", Med. & Biol. Eng., Vol. 13, pp. 123-125, 1975.

[117] A. VAN DER SLUIS, "Condition, equilibration, and pivoting in linear algebraic systems", Numer. Math., Vol. 15, pp. 74-86, 1970.

[118] A. VAN DER SLUIS, "Stability of the solution of linear least squares problems", Numer. Math., Vol. 23, pp. 241-254, 1975.

[119] J. VANDEWALLE, 'Signalen en ketens", course notes, V.T.K., Leuven, 1982.

[120] S. VAN HUFFEL, "A reliable, efficient deconvolution technique based on total linear least squares for calculating the renal retention function", M. Sc. Thesis in Biomed. Eng., K.U.Leuven, June 1985.

[121] S. VAN HUFFEL, "Documented Fortran programs of the classical total least squares computation, the partial singular value decomposition and the partial total least squares computation", Int. Report ESAT-KUL 87/11, ESAT Lab., Dept. of Electr. Eng., K.U.Leuven, April 1987.

[122] S. VAN HUFFEL, J. VANDEWALLE and J. STAAR, "The total linear least squares problem : properties, applications and generalization", Int. Report, ESAT Lab., Dept. of Electr. Eng., K.U.Leuven, Dec. 1984.

[123] S. VAN HUFFEL and J. VANDEWALLE, "The use of total linear least squares techniques for identification and parameter estimation", Preprints 7th IFAC/IFORS Symposium on Identification and System Parameter Estimation, York, U.K., July 3-7, Vol. 2, pp. 1167-1172, 1985.

[124] S. VAN HUFFEL and J. VANDEWALLE, "The use and applicability of the total least squares technique in linear regression analysis", Int. Report, ESAT Lab., Dept. of Electr. Eng., K.U.Leuven, June 1985.

[125] S. VAN HUFFEL and J. VANDEWALLE, "Subset selection using the total least squares approach in collinearity problems with errors in the variables", Lin. Alg. & Its Applic. (in press), 1987.

[126] S. VAN HUFFEL and J. VANDEWALLE, "Algebraic relationships between classical regression and total least-squares estimation", Lin. Alg. & Its Applic. (in press), 1987.

[127] S. VAN HUFFEL and J. VANDEWALLE, "The application of the total least squares technique in linear parameter estimation", Journal A (in press), 1987.

[128] S. VAN HUFFEL, J. VANDEWALLE, M. Ch. DE ROO and J.L. WILLEMS, "Reliable and efficient deconvolution technique based on total linear least squares for calculating the renal retention function", Med. & Biol. Eng. & Comput., Vol. 25, pp. 26-33, Jan. 1987.

[129] S. VAN HUFFEL, J. VANDEWALLE and A. HAEGEMANS, "An efficient and reliable algorithm for computing the singular subspace of a matrix, associated with its smallest singular values", J. Comput. & Applied Math. (in press), 1987.

[130] S. VAN HUFFEL, J. VANDEWALLE and J. STAAR, "The total linear least squares problem : formulation, algorithm and applications", Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS), Montréal, May 7-10, pp. 328-331, 1984.

[131] L.H. VAN STEKELENBURG, A. KOOMAN and J.F.W. TERTOOLEN, "A three-compartmental model for the transport and distribution of hippuran", Phys. Med. Biol., Vol. 21, pp. 74-84, 1976.

[132] J.T. WEBSTER, R.F. GUNST and R.L. MASON, "Latent root regression analysis", Technometrics, Vol. 16, No. 4, pp. 513-522, 1974.

[133] P.A. WEDIN, "Perturbation bounds in connection with the singular value decomposition", BIT, Vol. 12, pp. 99-111, 1972.

[134] J.H. WILKINSON, "The algebraic eigenvalue problem", Clarendon Press, Oxford, 1965.

[135] S. WOLD, A. RUHE, H. WOLD and W.J. DUNN, "The collinearity problem in linear regression, the partial least squares (PLS) approach to generalized inverses", SIAM J. Sci. Stat. Comput., Vol. 5, No. 3, pp. 735-743, 1984.

[136] D. YORK, "Least squares fitting of a straight line", Can. J. of Physics, Vol. 44, pp. 1079-1086, 1966.

[137] P. YOUNG, "Some observations on instrumental variable methods of time series analysis", Int. J. Control, Vol. 23, pp. 539-612, 1976.

[138] P.C. YOUNG, "Recursive estimation and time-series analysis : An introduction", Springer-Verlag, Berlin, 1976.

[139] P. YOUNG, "The instrumental variable method : A practical approach to identification and system parameter estimation", Preprints 7th IFAC/IFORS Symposium on Identification and System Parameter Estimation, York, U.K., July 3-7, Vol. 1, pp. 1-14, 1985.

APPENDIX A

THE SINGULAR VALUE DECOMPOSITION

One of the most important decompositions in matrix computations is the Singular Value Decomposition (SVD) [48, section 2.3].

Theorem A.1 : Singular Value Decomposition

If $A \in \mathbb{R}^{m \times n}$, then there exist orthonormal matrices

$U = [u_1, \ldots, u_m] \in \mathbb{R}^{m \times m}$ and $V = [v_1, \ldots, v_n] \in \mathbb{R}^{n \times n}$ such that

$$U^T A V = \Sigma = \text{diag}(\sigma_1, \ldots, \sigma_p) \qquad (A.1)$$

where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p \geq 0$ and $p = \min\{m,n\}$

The elements σ_i in Σ are called **singular values** of matrix A and their set is called the **singular value spectrum** $\sigma(A)$ of A. The columns u_i of U (resp., v_i of V) are called the left (resp., right) **singular vectors** of matrix A, and their set is called the left (resp., right) **singular basis** of A. The triplets (u_i, σ_i, v_i) are called the **singular triplets** of matrix A. It is easy to verify that ($p = \min\{m,n\}$) :

$$A = \sum_{i=1}^{p} \sigma_i u_i v_i^T \quad \text{, called the } \textbf{dyadic decomposition} \qquad (A.2)$$

$$A v_i = \sigma_i u_i \quad \text{and} \quad A^T u_i = \sigma_i v_i \qquad i = 1, \ldots, p \qquad (A.3)$$

A most illustrative way of looking at the SVD is by considering the linear map associated to the matrix A :

$$A : \mathbb{R}^n \rightarrow \mathbb{R}^m : x \rightarrow y = Ax = U \Sigma V^T x \qquad (A.4)$$

This linear operator A maps all unit vectors $x \in \mathbb{R}^n$ onto a hyperellipsoid E. The length of its semi-axes is precisely given by the singular values of the matrix A. This is illustrated geometrically in fig. A.1 for the case m = n = 2.

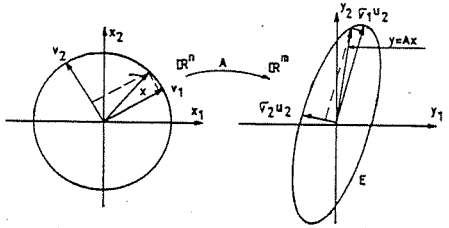


Fig. A.1 : Geometric illustration of the SVD properties of a linear map A.

$$A = \begin{bmatrix} 0.578 & -0.224 \\ 1.19 & 0.837 \end{bmatrix} \text{ with SVD}$$

$$U \Sigma V^T = \begin{bmatrix} 0.259 & -0.966 \\ 0.966 & 0.259 \end{bmatrix} \begin{bmatrix} 1.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} 0.866 & -0.5 \\ 0.5 & 0.866 \end{bmatrix}^T$$

and vector $x = \begin{bmatrix} 0.6 \\ 0.8 \end{bmatrix}$ is mapped onto $y = Ax = \begin{bmatrix} 0.167 \\ 1.383 \end{bmatrix}$ [105].

The SVD reveals a great deal about the structure of a matrix as evidenced now. We partition the SVD (A.1) of A as follows :

$$A = \underset{r \quad m-r}{[U_1 ; U_2]} \cdot \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{matrix} r \\ m-r \end{matrix} \cdot \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \begin{matrix} r \\ n-r \end{matrix} = U_1 \Sigma_1 V_1^T \qquad (A.5)$$
$$\qquad\qquad\qquad

One of the most valuable aspects of the SVD is that it enables us to deal sensibly with the concept of matrix rank. Rounding errors and fuzzy data make rank determination often a nontrivial exercise. In this regard the SVD is particularly useful because it permits us to quantify the notion of **near rank deficiency** and define the concept of numerical rank :

Theorem A.2 : Eckart-Young theorem

Let the SVD of $A \in \mathbb{R}^{m \times n}$ be given by theorem A.1.

If $k < r = \text{rank}(A)$ and $A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$

then

$$\min_{\text{rank}(B)=k} \|A-B\|_2 = \|A-A_k\|_2 = \sigma_{k+1} \qquad \text{(A.8)}$$

or equivalently :

$$\min_{\text{rank}(B)=k} \|A-B\|_F = \|A-A_k\|_F = \sqrt{\sum_{i=k+1}^{p} \sigma_i^2} \qquad p = \min\{m,n\} \qquad \text{(A.9)}$$

In 1936, Eckart and Young [31] gave an elegant constructive solution to this problem in terms of what is now called the SVD of A. Therefore, theorem A.2 is called the **Eckart-Young theorem**. It says that the smallest singular value of A is the 2-norm distance of A to the set of all rank-deficient matrices. If the smallest σ_i, $i = r+1, \ldots, p$ of A in (A.5) are assumed to be numerically zero, i.e. zero within the accuracy of the given data or within the precision of the computations, then A is said to have **numerical rank** r.

Finally, a last important theorem that presents the **interlacing property for singular values**, gives bounds on the perturbation of the singular values due to the removal of a column or row of a matrix. It is a direct consequence of Cauchy's interlacing theorem for eigenvalues of symmetric matrices [82]:

Theorem A.3 : Interlacing theorem for singular values

Let A be an m by n matrix. Let k be an integer, $1 \leq k \leq n$. Let B be the m by n-k matrix which results from A by deleting k of its columns. Then

the ordered singular values β_i of B interlace with those α_i of A as follows :

For $j = 1, \ldots, \min\{m,n-k\}$ (for convenience : $\alpha_i = 0$ if $i > \min\{m,n\}$) :

$$\alpha_j \geq \beta_j \geq \alpha_{j+k} \qquad \text{(A.10)}$$

Alternatively, for $j = 1, \ldots, \min\{m,n\}$ (for convenience :

$\beta_i = \begin{cases} 0 & \text{if } i > \min\{m,n-k\} \\ +\infty & \text{if } i < 0 \end{cases}$) :

$$\beta_{j-k} \geq \alpha_j \geq \beta_j \qquad \text{(A.11)}$$

If B results from A by deleting one column of A (k=1), then theorem A.3 yields :

$$\text{if } m \geq n : \alpha_1 \geq \beta_1 \geq \alpha_2 \geq \beta_2 \geq \ldots \geq \beta_{n-1} \geq \alpha_n \geq 0 \qquad \text{(A.12)}$$

$$\text{if } m < n : \alpha_1 \geq \beta_1 \geq \alpha_2 \geq \beta_2 \geq \ldots \geq \alpha_m \geq \beta_m \geq 0 \qquad \text{(A.13)}$$

APPENDIX B

STATISTICAL PROPERTIES OF AN ESTIMATOR

Let us in the first place consider what we mean by "**estimation**". Assume that the population under study is distributed in a form which is completely determinate but for the value of some parameter ξ. We have to choose a method for estimating ξ. It will clarify our ideas if we draw a distinction between the method or rule of estimation, which we call an **estimator**, and the value to which it gives rise in particular cases, the **estimate**. The distinction is the same as that between a function f(y), regarded as defined for a range of the variable y, and the particular value which the function assumes, say f(a), for a specified value of y equal to a. Our problem is not to find estimates, but to find estimators. We do not reject an estimator because it may give a bad result in a particular case (in the sense that the estimate differs materially from the true value). We should only reject it if it gave bad results in the long run, that is to say, if the distribution of possible values of the estimator were seriously discrepant with the true value of ξ. **The merit of the estimator x is judged by the distribution of estimates to which it gives rise**, i.e. by the properties of its sampling distribution [62].

It is well-known that the **distribution of a random variable x** is characterized by its mean and its variance, i.e. the square of its standard deviation which describes the extent to which values of the variable x are scattered around their mean. The **mean** of x is defined by :

$$\text{mean } (x) = E(x) \tag{B.1}$$

where E stands for the expectation or the expected value of the estimator between brackets. Taking enough samples x^k of x, k = 1, ..., m, E(x) can be estimated by the average value $\bar{x}$, called the **sample mean** :

$$\bar{x} = \sum_{k=1}^{m} x^k / m \tag{B.2}$$

The **variance** of x is defined by :

$$\text{var } (x) = E\,(x - E(x))^2 \tag{B.3}$$

If m samples x^k of x, k = 1, ..., m, are available, var (x) can be estimated by

$$\text{the } \textbf{sample variance} = \sum_{k=1}^{m} (x^k - \bar{x})^2 / m \tag{B.4}$$

More generally, if $x = [x_1, \ldots, x_n]^T$ is a random **vector** of n elements x_i, then its expectation or mean is itself a vector containing the expectations of each of the elements x_i of x. In this case the distribution of x is also characterized by the covariance between its elements. The **covariance** between the elements x_i and x_j of x is defined by :

$$\text{cov } (x_i, x_j) = E[(x_i - E(x_i))(x_j - E(x_j))] \tag{B.5}$$

If m vector samples x^k of x, k = 1, ..., m, are available, cov (x_i, x_j) can be estimated by

$$\text{the } \textbf{sample covariance} = [\sum_{k=1}^{m} (x_i^k - \bar{x}_i)(x_j^k - \bar{x}_j)] / m \tag{B.6}$$

Variances and covariances between the elements of x are characterized in its covariance matrix Σ_x. The **covariance matrix** of a random vector x is defined by a matrix whose diagonal entries are the variances of the individual random elements in x and whose off-diagonal entries are the covariances between any pair of different elements x_i and x_j in x. Notationally,

$$\text{cov } (x) = \Sigma_x = \begin{bmatrix} \text{var } (x_1) & \cdots & \text{cov } (x_1, x_n) \\ \text{cov } (x_2, x_1) & & \text{cov } (x_2, x_n) \\ \vdots & & \vdots \\ \text{cov } (x_n, x_1) & \cdots & \text{var } (x_n) \end{bmatrix} \tag{B.7}$$

The covariance matrix is symmetric.

These concepts allow us to define the following **properties of an estimator x** [62] :

- **Unbiasedness** : Knowing that the estimates of x will vary from sample to sample, we would like to ensure that most of these estimates are in some sense close to the true value ξ. One property of an estimator

that often provides "close" estimators is that of unbiasedness. If, on the average, an estimator neither tends to over - nor underestimate the true parameter value, it is said to be an unbiased estimator. Another way of stating that **an estimator is unbiased** is to declare **that its expectation equals the true parameter.** Notationally, if we require that for all sample lengths m and ξ the mean value of x is ξ, we define an unbiased estimator x by the relation

$$E(x) = \xi \qquad (B.8)$$

Note that this does not guarantée that any single estimated value equals the true parameter value, only that **if many data sets are analyzed** and estimates are calculated for each data set, **the average estimated value will tend to equal the true parameter value.** For example, the sample variance (B.4) is not an unbiased estimator of the variance (B.3) because its expectation equals (m-1)/m var (x). Therefore, the unbiased estimator $\sum_{k=1}^{m} (x^k - \bar{x}) / (m - 1)$ is usually preferred to the sample variance.

- **Consistency** : It is a very desirable property for **an estimator to improve its accuracy with increasing m.** For instance, if the variance of the sampling distribution of an estimator decreases with increasing m, it is necessary that its central value should tend to ξ. Otherwise the estimator would have values differing systematically from the true value. We therefore define the following property for a suitable estimator. An estimator x_m, computed from a sample of m values, is said to be a **consistent estimator** of the true value ξ if x_m **converges to ξ as m tends to infinity.** Several convergence concepts are possible [20].

  When dealing with consistency we use "**convergence with probability one**" (w.p. 1), also called **strong consistency.** This means that the probability for the event ($x_m \rightarrow \xi$ as $m \rightarrow \infty$) is 1. Notationally, we have :

$$\lim_{m \rightarrow \infty} x_m = \xi \qquad \text{w.p. 1} \qquad (B.9)$$

  In the statistical literature, the normal choice is "**convergence in probability**" which is a weaker concept (implied by, but not equivalent to convergence w.p. 1). Then it is required that the probability P :

$$P(|x_m - \xi| > \varepsilon) \rightarrow 0 \quad \text{as} \quad m \rightarrow \infty \quad \text{for every } \varepsilon > 0 \qquad (B.10)$$

or notationally equivalent :

$$\operatorname{plim}_{m \rightarrow \infty} x_m = \xi \qquad (B.11)$$

A still weaker but useful concept is "**convergence in distribution**", i.e. the distribution density of the estimator x_m must converge to the distribution function of ξ.

Consistent estimators are not necessarily unbiased. Thus neither property implies the other. But a consistent estimator whose asymptotic distribution has finite mean value, must be asymptotically unbiased.

- **Minimum variance** : Under fairly general conditions there exists a bound below which the variance of an unbiased estimator can not fall. This bound is given by :

$$\text{var}(x) = E(x - f(\xi))^2 \geq (f'(\xi))^2 / E[(\delta \log L / \delta \xi)^2] \qquad (B.12)$$

  where x is the unbiased estimator of some function $f(\xi)$ of ξ, $f'(\xi)$ is its first derivative, δ is the partial differentiation operator and L is the likelihood of x.

  (B.12) is the fundamental inequality for the variance of an estimator, often known as the **Cramer-Rao** inequality. We call (B.12) the **minimum variance bound** for the estimation of $f(\xi)$. An estimator which attains this bound for all ξ is called a **minimum variance estimator.**

- **Efficiency** : We now turn to consideration of **large sample** properties. Even if there is no minimum variance estimator for each value of m, there will often be one **as m tends to infinity.** Since most of the estimators we use are asymptotically normally distributed by virtue of the Central Limit theorem, the distribution of such an estimator will depend for large samples on only two parameters : its mean value and its variance. If it is a consistent estimator it will commonly be asymptotically unbiased. This leaves the variance as the means of discriminating between **consistent**, asymptotically normal estimators of the same parametric function. Among such estimators, one **with minimum variance in large samples** is called an **efficient estimator.** If we compare asymptotically normal estimators in large samples, we may reasonably set up a **measure of efficiency**, something we do not attempt to

do for small m and arbitrarily distributed estimators. We define the efficiency of any other estimator, relative to an efficient estimator, as the reciprocal of the ratio of sample numbers required to give the estimators equal sampling variances, i.e. to make them equally precise.

- **Minimum mean squared error** : It is reasonable to argue that the presence of bias should not necessarily outweigh small sampling variance in an estimator. What we are really demanding of an estimator x is that it should be "**close**" **to the true value** ξ. Therefore, we consider its mean squared error (MSE) around that true value, instead of its own expected value. We have at once :

$$MSE\ (x) = E\ (x - \xi)^2 = \underbrace{E\ (x - E(x))^2}_{\text{var } (x)} + \underbrace{(E(x) - \xi)^2}_{\text{squared bias } (x)} \qquad (B.13)$$

the cross-product term on the right being equal to zero. The last term on the right is simply the square of the bias of x in estimating ξ. If x is unbiased, this last term is zero, and MSE becomes variance. In general, however, the minimization of MSE gives different results. Minimum MSE estimators are not much used. The objection to them is a practical, rather than a theoretical one. Minimum variance unbiased estimators are more tractable because they avoid the difficulty by insisting on unbiasedness.

The criteria of estimation which we have so far discussed, namely consistency, unbiasedness, minimum variance or minimum mean squared error and efficiency are reasonable guides in assessing the statistical properties of an estimator. They are used in chapter 4 and 5 of this thesis.

A P P E N D I X C

NEDERLANDSE SAMENVATTING

C.1. INLEIDING

De drijfveer van het onderzoek verricht in deze thesis is duidelijk de nood aan een **veralgemening van het lineair parameterschattingsprobleem** dat meetfouten op alle veranderlijken die het systeem kenmerken, in rekening brengt. Dergelijke problemen treden op in vele, sterk verscheiden wetenschapsdomeinen zoals signaalverwerking, automatische controle, systeemidentificatie en meer algemeen in statistiek, natuurkunde, economie, biologie, geneeskunde, ... Ze kunnen doorgaans beschreven worden door een lineaire vergelijking :

$$a_1 x_1 + \ldots + a_n x_n = b \qquad (C.1)$$

$a_1, \ldots, a_n$ en b zijn de veranderlijken. $x = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$ bevat de parameters die het systeem kenmerken. Het basisprobleem bestaat erin deze ongekende parameters x te schatten met behulp van verschillende waarnemingen van de veranderlijken. Dit leidt tot een **overgedetermineerd** stelsel van m vergelijkingen in n onbekenden (m > n) :

$$Ax = b \qquad (C.2)$$

De i-de rij van de gegevensmatrix A en de waarnemingsvector b bevatten de metingen van de veranderlijken $a_1, \ldots, a_n$ en b respectievelijk.

Indien zowel A als b onderhevig zijn aan fouten, is de **klassieke** (of gewone) **kleinste kwadraten (KK) benadering** niet meer geschikt. Inderdaad, deze methode laat slechts meetfouten toe op 1 veranderlijke van het systeem. Nieuwe benaderingstechnieken moeten ontwikkeld worden die **meetfouten op alle gegevens** verwerken. De **totale kleinste kwadraten (TKK) benadering** die in deze thesis bestudeerd wordt, is een dergelijke benaderingstechniek. Ze werd door Golub en Van Loan [47] ingevoerd in de numerieke analyse en is gebaseerd op de krachtige singuliere waardenontbinding (SWO) [48, sec. 2.3]. Haar beloftevolle numerieke en geometrische eigenschappen sporen ons aan om de **TKK benadering toe te passen** op verschillende problemen in systeemidentificatie, modelvorming, ... Het belang en de **praktische bruikbaarheid** van deze methode volgt uit een vergelijking met de meest gangbare schattingsmethoden in elk domein. Dit is de **voornaamste doelstelling** van dit werk.

Dit parameterschattingsprobleem waarin alle betrokken veranderlijken niet exact kunnen waargenomen worden, is zeker niet nieuw. Inderdaad, het eenvoudigste geval (n = 1) waarin de meetpunten door een rechte benaderd worden, werd reeds sinds 1978 bestudeerd [3]. Nochtans werd deze techniek pas recent uitgebreid tot meerdere veranderlijken. Dit gebeurde door Golub en Van Loan [47] vanuit de numerieke analyse en onafhankelijk daarvan door Gleser [41] vanuit de statistiek. Beide auteurs veralgemeenden de TKK benadering eveneens voor problemen met meerdere waarnemingsvectoren B in het rechterlid van (C.2). In deze thesis wordt de TKK methode vooral bestudeerd vanuit de **lineaire algebra**. De statistische eigenschappen komen slechts kort aan bod.

C.2. HET TOTALE KLEINSTE KWADRATENPROBLEEM : FORMULERING, UITBREIDINGEN EN ALGORITME

Het TKK probleem dat aangewezen is voor lineaire problemen AX = B waarin zowel A als B onderhevig zijn aan fouten, wordt in **hoofdstuk 1** volledig doorgelicht. Het kan als volgt geformuleerd worden.

TKK definitie :

Gegeven een stelsel van m vergelijkingen AX = B, $B \in \mathbb{R}^{m \times d}$ in n×d onbekenden X. Een **TKK oplossing** is elke oplossing X van het stelsel

$$\hat{A}X = \hat{B} \qquad (C.3)$$

waarin $\hat{A}$ en $\hat{B}$ voldoen aan de volgende criteria :

$$R(\hat{B}) \subset R(\hat{A}) \qquad (C.4)$$

$$\|[\Delta\hat{A};\Delta\hat{B}]\|_F = \|[A-\hat{A};B-\hat{B}]\|_F \text{ is minimaal} \qquad (C.5)$$

De bepaling van een $[\Delta\hat{A};\Delta\hat{B}]$ die voldoet aan (C.4-C.5), wordt het **TKK probleem** genoemd.

Uit (C.5) volgt dat het **TKK benaderingseffect steeds kleiner** is dan het benaderingseffect van de klassieke KK methode. Dit verschil in benadering komt vooral tot uiting in een extreem voorbeeld (zie sectie 1.2) en in de geometrische voorstelling van beide problemen in de kolom- en rijruimte van $[A;B]$ (sectie 1.3).

Voor **generische**, ééndimensionale TKK problemen met unieke oplossing herleidt de TKK oplossing zich tot een eenvoudige scalering (1.16) van de rechtersinguliere vector van $[A;b]$ die hoort bij zijn kleinste singuliere waarde. Dit geldt voor problemen met slechts 1 waarnemingsvector b (ééndimensionaal), waarin de TKK oplossing uniek is en kan berekend worden met het algoritme van Golub en Van Loan (generisch). Deze problemen kunnen nu **uitgebreid** worden. In het geval van **meerdere** waarnemingsvectoren B ($d > 1$) wordt de TKK oplossing (1.25) berekend uit een basis van de rechtersinguliere deelruimte van $[A;B]$ die hoort bij zijn d kleinste singuliere waarden. Indien deze singuliere waarden echter samenvallen met andere singuliere waarden van $[A;B]$, is de TKK oplossing niet meer uniek. Om toch een unieke oplossing te selecteren, berekent de TKK methode in dit geval de **minimum norm** oplossing (1.41). Bovendien wordt in dit werk de TKK oplossing **veralgemeend** voor alle **niet generische** problemen. Dit zijn problemen waarin het algoritme van Golub en Van Loan **faalt** doordat de generische TKK oplossing naar **oneindig** gaat. De eigenschappen van deze niet generische problemen worden bewezen in sectie 1.6. Voor deze problemen wordt de TKK oplossing veralgemeend door bijkomende rechtersinguliere vectoren van $[A;B]$ die horen bij grotere singuliere waarden, te betrekken in de berekening. Deze veralgemeende TKK oplossing blijft bovendien beantwoorden aan de TKK criteria (C.4-C.5) voor elk multidimensionaal probleem. Verder wordt de TKK methode ook uitgebreid tot problemen waarin de gegevensmatrix A **enkele exacte kolommen** bevat. In dit geval wordt de oplossing bekomen door een TKK probleem van kleinere dimensie op te lossen, na een voorafgaandelijke QR factorisatie van de exacte kolommen.

De berekening van de TKK oplossing voor alle besproken problemen wordt samengevat in 1 algoritme, dat we het **klassieke TKK algoritme** noemen omdat het gebaseerd is op dat van Golub en Van Loan. Een volledig gedocumenteerd Fortranprogramma van dit algoritme is gegeven in [121].

C.3. EIGENSCHAPPEN VAN HET TKK PROBLEEM EN HUN VERBAND MET DEZE VAN HET KK PROBLEEM

In **hoofdstuk** 2 wordt de TKK techniek op een **analytische** manier **rechtstreeks** vergeleken met de klassieke kleinste kwadraten (KK) methode **vanuit verschillende invalshoeken.** Niet alleen de **oplossingen** maar ook de **residu's** van beide methoden worden vergeleken. Verder bestuderen we ook hun **benaderingeffecten** om een stelsel van vergelijkingen compatibel te maken. En tenslotte wordt met een geschikte afstandsfunctie de **afstand** beschreven **tussen hun respectievelijke deelruimten** waarin de gegevens A en B geprojecteerd worden om het stelsel van vergelijkingen compatibel te maken. Voor al deze verschillende aspecten van het probleem worden interessante verbanden tussen beide methoden afgeleid en worden opmerkelijke eigenschappen bewezen. Deze brengen de **gelijkenissen en verschilpunten** tussen beide benaderingen aan het licht.

Bovendien geven deze eigenschappen meer inzicht in de **gevoeligheid** van beide technieken voor perturbaties op de gegevens. Inderdaad, verschillende berekende bovengrenzen tonen duidelijk aan dat de **TKK oplossing** van een stelsel $AX = B$ het **sterkst verschilt** van de gewone KK oplossing als **B evenwijdig is met de singuliere vector van A die hoort bij zijn kleinste singuliere waarde.** Hetzelfde kan gezegd worden over de residu's. Tevens wordt in dit geval de verhouding van het TKK benaderingseffect op het gewone KK benaderingseffect minimaal en wordt de afstand tussen de deelruimten van de TKK benadering enerzijds en de gewone KK benadering anderzijds maximaal. Zoals geïllustreerd is in hoofdstuk 3 komt in dit geval ook de nauwkeurigheid van de TKK oplossing bij perturbatie van alle gegevens het sterkst tot uiting.

Verschillende andere eigenschappen bewijzen dat de **TKK benadering in toenemende mate verschilt** van de klassieke KK benadering wanneer de ver-

gelijkingen AX = B **minder compatibel** zijn, wanneer de waarnemingsmatrix B (en dus ook de oplossing X) **groter** wordt in norm, of wanneer A **niet van volle rang** dreigt te zijn. De belangrijkste **invloedsparameters** hiervoor zijn de kleinste singuliere waarden van A en [A;B], de oriëntatie van B in de kolomruimte van A en de lengte van B of X (in norm). Diezelfde parameters bepalen ook de relatieve verschillen in gevoeligheid tussen beide methoden (hoofdstuk 3).

Vermeldenswaard is ook het feit dat verschillende **perturbatiemodellen** corresponderen met eenzelfde TKK oplossing. Inderdaad, de TKK methode maakt enkel veronderstellingen over de verdeling van de fouten in de rechtersinguliere deelruimte van [A;B] die hoort bij zijn kleinste singuliere waarden. Selecteren we nu het model met minimale norm, dan bekomen we de TKK correctiematrix $[\Delta\hat{A};\Delta\hat{B}]$ in (C.5). Verder wordt aangetoond hoe de TKK methode een som van gewogen gekwadrateerde residu's minimiseert en zich aldus herleidt tot een **gewogen KK probleem.**

Tenslotte vermelden we nog dat verschillende eigenschappen, bewezen in hoofdstuk 2, enkel gelden wanneer de d kleinste singuliere waarden van [A;B] aan elkaar **gelijk** zijn (d is het aantal waarnemingsvectoren in B). Deze eis lijkt zeer beperkend. Doch indien de gegevens A en B waarnemingen zijn van een **exacte relatie** $A_o X = B_o$ en de meetfouten op elke rij in [A;B] onafhankelijk en identiek verdeeld zijn met gemiddelde nul en gemeenschappelijke covariantiematrix van de vorm $\sigma^2 I$ (I is de identiteitsmatrix), dan is deze eis **statistisch voldaan** en blijven de eigenschappen **zinvol.**

C.4. DE GEVOELIGHEID VAN HET TKK PROBLEEM EN ZIJN VERBAND MET HET KK PROBLEEM

In **hoofdstuk** 3 bestuderen we de gevoeligheid van de TKK en de KK methode. Hiertoe vergelijken we de relatieve afwijkingen van beide benaderingen ten gevolge van perturbatie van het overeenkomstig ongeperturbeerd probleem $A_o x = b_o$. Vermits de TKK methode speciaal ontworpen is voor geperturbeerde problemen, is een dergelijke gevoeligheidsstudie van het allergrootste belang voor ieder die de TKK methode wil toepassen.

Eerst en vooral worden de **perturbatieëffecten** bestudeerd, die de **nauwkeurigheid van beide oplossingen het sterkst aantasten.** De sterkte van deze effecten wordt kwantitatief uitgedrukt door het **conditiegetal** van het probleem $A_o x = b_o$. Dit getal drukt uit hoeveel maal de relatieve fout op de oplossing x groter kan zijn dan de relatieve fout op de gegevens A_o en b_o. Indien helemaal geen informatie over de foutenverdeling op de gegevens beschikbaar is, moeten deze maximale perturbatieëffecten opgevat worden als een verlies aan nauwkeurigheid in de berekende oplossing dat daadwerkelijk kan optreden. De **voorwaarden** waarbij dit grootste verlies aan nauwkeurigheid optreedt, worden afgeleid uit de berekende bovengrenzen voor de fout op de oplossing. Zo wordt aangetoond dat de maximale **absolute** fout enerzijds en de maximale **relatieve** fout anderzijds in de oplossing bereikt worden onder **verschillende** voorwaarden van perturbatie en oriëntatie van de gegevens. In **problemen waarin** $b_o \in R(A_o)$ is het maximale verlies aan nauwkeurigheid in de TKK en de KK oplossing **even groot** doch voor andere problemen waarin $b_o \notin R(A_o)$ kan de **nauwkeurigheid van de TKK oplossing sterker aangetast** worden. Nochtans moet hier gewezen worden op het feit dat deze perturbatieëffecten die het verlies aan nauwkeurigheid maximiseren, enkel optreden bij een speciale oriëntatie van b_o en een bijzondere perturbatie van de gegevens. In andere omstandigheden is dit verlies aan nauwkeurigheid veel kleiner.

Vervolgens worden de **perturbatieëffecten op de SWO** bestudeerd. Uitgaande van een gevoeligheidsanalyse voor singuliere deelruimten [133], wordt voor het TKK en het KK probleem de **maximale afstand** berekend tussen hun geperturbeerde singuliere deelruimten en de overeenkomstige ruimten van het ongeperturbeerd probleem. Hieruit besluiten we dat de **deelruimten verbonden met het TKK probleem minder gevoelig** zijn dan deze verbonden met het KK probleem. Dit is het geval voor elk probleem waarin $b_o \in R(A_o)$ maar ook voor elk ander voldoende compatibel probleem waarin $b_o \notin R(A_o)$. De TKK methode is des te meer nauwkeurig dan de gewone KK methode naarmate het aantal waarnemingsvectoren B_o en hun lengte toenemen doch vooral naarmate B_o (of de oplossing X) dichter aanleunt bij de singuliere vector van A_o die hoort bij zijn kleinste singuliere waarde. Wanneer B_o evenwijdig is met deze laatstgenoemde vector, is de **verwachte winst in nauwkeurigheid** van de TKK oplossing **maximaal.** Dit wordt bevestigd in onze simulaties (zie fig. 3.4, 3.5 en 3.6).

Tenslotte worden ook nog speciale deelruimteperturbaties afgeleid die de nauwkeurigheid van het TKK probleem **maximaal** aantasten. Hiertoe wordt de grootst mogelijke hoek berekend waarover deze deelruimten kunnen roteren onder invloed van perturbaties. Het effect van deze perturbaties op de nauwkeurigheid van de TKK oplossing kan berekend worden voor één-dimensionale problemen.

C.5. DE STATISTISCHE EIGENSCHAPPEN VAN HET TKK PROBLEEM

Zodra **informatie over de foutenverdeling** op de gegevens beschikbaar is, kunnen statistische eigenschappen van het TKK probleem afgeleid worden (zie **hoofdstuk 4**). Deze eigenschappen beoordelen de **verwachte kwaliteit van de TKK methode** als parameterschatter in een model waarin alle betrokken veranderlijken onderhevig zijn aan fouten. In tegenstelling met hoofdstuk 3, doen ze geen uitspraak over de kwaliteit van een TKK oplossing voor een bijzondere keuze van de foutenmatrix.

Statistische eigenschappen zijn steeds gekoppeld aan een model waarin de parameters moeten geschat worden. De meest succesvolle benadering, rechtstreeks verbonden met het TKK concept, is **het model met "fouten in alle veranderlijken"**. Dit model beschouwt alle waarnemingen als afkomstig van ongekende exacte waarden plus meetfouten. Deze exacte waarden worden verondersteld te voldoen aan een lineair verband, b.v. (C.1). Verder wordt verondersteld dat de fouten op de waarnemingen (b.v. de rijen van [A;b] in (C.2) of meer algemeen de rijen van [A;B] in het geval van meerdere waarnemingsvectoren in B) onafhankelijk en identiek verdeeld zijn met gemiddelde nul en gemeenschappelijke covariantiematrix. Wanneer deze laatste van de vorm $\sigma^2 I$ is, kan bewezen worden dat de TKK oplossing van het model een **sterk consistente** schatter is van de echte doch ongekende parameters X van het model [41-124]. Dit betekent dat de TKK oplossing van AX = B met waarschijnlijkheid één convergeert naar de echte doch ongekende parameters van het model als het aantal waarnemingen (of vergelijkingen) m naar oneindig gaat. Het is welbekend in lineaire regressie dat de gewone KK oplossing altijd inconsistent is in dit geval.

Behalve een parameterschatter X kunnen ook **schatters voor het assensnijpunt** van het lineair model (indien verschillend van nul) en de **foutenvariantie** σ^2 berekend worden uit resultaten van de TKK benadering. Onder redelijke voorwaarden wordt bewezen dat ook deze schatters sterk consistent zijn, onafhankelijk van de gemeenschappelijke verdeling van de fouten op de gegevens. Wanneer deze verdeling eindige momenten van vierde orde heeft, dan zijn al deze schatters **asymptotisch normaal** verdeeld. Wat de **praktische haalbaarheid** van de veronderstellingen betreft, wordt aangetoond dat ze zeer redelijk zijn. In het bijzonder wordt bewezen dat de TKK methode nog steeds consistente schatters kan berekenen indien de covariantiematrix de meer algemene vorm $\sigma^2 \Sigma_o$ heeft waarbij Σ_o gekend is en positief definiet is. Hiertoe moeten de gegevens $[A;B]^T$ op gepaste wijze getransformeerd worden door vermenigvuldiging langs links met $\Sigma_o^{-\frac{1}{2}}$. We mogen dus besluiten dat de **TKK methode een sterk consistente schatter berekent in alle gevallen waarin de covariantiematrix gekend is, op een evenredigheidsfactor na.** Bovendien blijven alle hierboven vermelde eigenschappen van de schatter goed benaderd in toepassingen met een **middelmatig** aantal waarnemingen m, ondanks het feit dat asymptotische eigenschappen gebruikt worden om de kwaliteit van de parameterschatters te beoordelen. Dit is bevestigd in simulaties (zie fig. 4.1).

C.6. VERBANDEN TUSSEN TKK SCHATTING EN KLASSIEKE LINEAIRE REGRESSIE

In **hoofdstuk 5** wordt de TKK schatter van de parameters in een lineair model met "fouten in alle veranderlijken" vergeleken met de klassieke schatters van de parameters in een lineair regressiemodel. Het is belangrijk beide schattingsproblemen die op het eerste gezicht sterk op elkaar gelijken, goed te onderscheiden. Het **klassiek lineair model met "fouten in alle veranderlijken"** veronderstelt altijd een **exact**, weliswaar ongekend **lineair verband**, b.v. (C.1), tussen de echte veranderlijken van het model waarvan geen enkel foutloos kan waargenomen worden. Het **lineair regressiemodel** daarentegen veronderstelt dat alle onafhankelijke veranderlijken a_i wel **foutloos** kunnen waargenomen worden doch laat bijkomend toe dat de afhankelijke veranderlijke b nog beïnvloed wordt door andere factoren dan meetfouten alleen. Dit betekent dat de veranderlijken nu niet noodzakelijk moeten voldoen aan een lineaire relatie (C.1).

De voornaamste parameterschatter in een lineair regressiemodel is de gewone KK oplossing. Ondanks zijn zeer goede eigenschappen, kan deze schatter zeer grote varianties bezitten wanneer de gegevens **multicollineair** zijn. Dit betekent dat de kolommen van A in (C.2) bijna afhankelijk zijn zodat A zijn volle rang verliest. Door een kleine vertekening in de KK schatter in te voeren, kan de variantie van de KK oplossing sterk verminderd worden. Deze gewijzigde KK schatters worden **vertekende schatters** genoemd en vertonen een beter gedrag in problemen met multicollineariteiten [49]. Drie vertekende schatters worden hier besproken, namelijk hoofdcomponentenregressie, randregressie ("ridge regression") en latente wortelregressie ("latent root regression"). Met behulp van de **SWO** in hun definities, worden hun onderlinge verbanden duidelijk, alsook hun relatie tot de TKK schatter.

De SWO en **geometrische concepten** worden nu gebruikt om allerlei **algebraische equivalenties** tussen de klassieke regressieschatters en de TKK schatter af te leiden in problemen met exact gedefinieerde multicollineariteiten. Zo wordt bewezen dat hoofcomponentenregressie **equivalent** is met latente wortelregressie. Ook worden de **verschillen** in benadering tussen latente wortelregressie en de TKK benadering verduidelijkt. Indien (C.2) een exacte oplossing heeft, wordt deze gevonden door vrijwel alle methoden [126].

C.7. SELECTIE VAN EEN DEELMODEL DOOR MIDDEL VAN DE TKK BENADERING IN PROBLEMEN MET MULTICOLLINEARITEITEN EN FOUTEN IN ALLE VERANDERLIJKEN

Zoals vermeld in hoofdstuk 5, verwijst het begrip multicollineariteit naar de situatie waarin een bijna exact lineair verband bestaat tussen de veranderlijken in het model. Hierdoor worden de coëfficiënten van de gewone KK oplossing onstabiel. Om dit te verhelpen, worden vertekende schatters gebruikt. Deze kunnen weliswaar de dimensie van de gegevensruimte verminderen, b.v. door rangverlaging, doch **alle** veranderlijken van het oorspronkelijke model blijven deel uitmaken van deze schatters. In veel toepassingen is het **wenselijk om niet alleen de dimensie van de gegevensruimte te verlagen maar ook het aantal veranderlijken in het model**. De manier waarop in een model, b.v. (C.1), veranderlijken a_i geselecteerd worden die het best het gedrag van de veranderlijke b kunnen benaderen, wordt **deelmodelselectie** genoemd. Dit probleem wordt in **hoofdstuk 6** behandeld voor het geval dat geen enkele veranderlijke in model (C.1) foutloos kan waargenomen worden. Dit betekent dat alle elementen van A en b in (C.2) onderhevig zijn aan fouten. Met behulp van de TKK benadering wordt een nieuw **deelmodelselectiealgoritme SAB-TLS** afgeleid [124-125] en vergeleken met het gangbaar deelmodelselectiealgoritme SA-LS van Golub, Klema en Stewart [44-48, sec. 12.2]. Hun respectievelijke **selectieëigenschappen** en **nauwkeurigheid in parameterschatting** en **voorspelling** worden grondig geanalyseerd. Wat voorspelling betreft, beschouwen we hier enkel het geval waarin het berekend deelmodel dient om het gedrag te voorspellen in het echte, niet geperturbeerde systeem, b.v. in hypothesen en simulatiestudies. Het TKK concept is inherent met dit soort voorspelling verbonden.

Uit ons onderzoek kunnen volgende **besluiten** getrokken worden. Het **SA-LS algoritme** is enkel aangewezen als de veranderlijken in de gegevensmatrix A **foutloos** kunnen bekomen worden. Zodra de veranderlijken niet meer foutloos zijn en enkel geperturbeerde gegevens beschikbaar zijn, is de TKK benadering aangewezen om de meest nauwkeurige parameterschatters te berekenen in elk geselecteerd deelmodel. Indien nu de **perturbaties** op de waarnemingen in A en b **voldoende klein** zijn en indien de vergelijkingen van het deelmodel **slecht geconditioneerd** zijn, wordt het gedrag van de niet geperturbeerde veranderlijke b_o nauwkeuriger voorspeld met het deelmodel geselecteerd op basis van de r-de rechtersinguliere deelruimte $V_r'^T$ van A die hoort bij zijn r grootste singuliere waarden (r wijst op de grootte van het deelmodel). In alle andere gevallen zijn voorspellingen met het deelmodel geselecteerd op basis van de r-de rechtersinguliere deelruimte V_r^T van [A;b] **nauwkeuriger** en is dus het SAB-TLS algoritme aangewezen. Dit is het geval voor **goed geconditioneerde** stelsels of bij **toenemende perturbatie** van de gegevens zodat de variantie van de fouten een bepaalde drempel overschrijdt. Deze drempel is functie van de r-de singuliere waarde van A.
Bovendien maakt SAB-TLS gebruik van **bijkomende informatie in de waarnemingsvector** b en waarborgt aldus een **betere stabiliteit en betrouwbaarheid in de selectie van het deelmodel**, zelfs voor aanzienlijke perturbaties van alle gegevens. Eveneens wordt een **variante** van het SAB-TLS al-

goritme ontwikkeld. Vermits dit algoritme rekenefficiënter is en bovendien een even nauwkeurig deelmodel berekent, wordt deze variante aanbevolen.

C.8. HET GEBRUIK VAN DE TKK BENADERING IN DE SCHATTING VAN IMPULSRESPONSIES DOOR MIDDEL VAN DISCRETE DECONVOLUTIE

In hoofdstuk 7 en 8 bestuderen we hoe de TKK methode kan toegepast worden in **systeemidentificatie** en **parameterschatting in het tijdsdomein**. Deze toepassingen illustreren hoe ieder kan tewerk gaan om betere resultaten te bekomen. Verwacht wordt dat de TKK methode even succesvol kan aangewend worden in veel andere domeinen en toepassingen.

In **hoofdstuk 7** gaan we na hoe de TKK methode kan gebruikt worden voor het schatten van de impulsresponsie van een systeem. De **impulsresponsie** kenmerkt volledig het in- en uitgangsgedrag van een systeem en speelt een belangrijke rol in de studie van de interne relaties van vele industriële en biologische systemen. Elk proces dat kan gemodelleerd worden als een **lineair, tijdsinvariant, causaal** systeem met **begintoestand nul**, beantwoordt aan volgend **impulsresponsiemodel** [119] :

$$y(t) = \sum_{k=o}^{t} u(k)\ h(t-k) \qquad (C.6)$$

Hierin zijn y(t), u(t) en h(t) de uitgang, ingang en impulsresponsie respectievelijk op tijdstip t. Wanneer nu de in- en uitgangen kunnen waargenomen worden gedurende een voldoende lang tijdsinterval t = 1,..., m kan de impulsresponsie h(t), t = 1, ..., n van lengte n berekend worden door **discrete deconvolutie**. Wanneer m $\geq$ n kan het systeem volledig geïdentificeerd worden. Dit veronderstelt dat het systeem een **eindige** toestandsdimensie heeft. Door nu (C.6) uit te schrijven voor t = 1, ..., m bekomen we een **overgedetermineerd** stelsel van vergelijkingen. De oplossing van dit stelsel is de impulsresponsie van het systeem. Indien zowel de in- als uitgangen van het systeem niet nauwkeurig kunnen opgemeten worden, is de TKK methode aangewezen [123]. Zijn deze in- en uitgangen bovendien **geperturbeerd door stationaire witte ruis** met gemiddelde nul, dan berekent de TKK methode een **sterk consistente** schatter van de exacte impulsresponsie van het overeenkomstig ongeperturbeerd sys-

teem. Een simulatievoorbeeld (zie fig. 7.2) illustreert duidelijk de winst in nauwkeurigheid van de TKK methode in vergelijking met de gewone KK methode.

De **praktische bruikbaarheid van de TKK benadering** als deconvolutietechniek wordt verder duidelijk in een **concrete toepassing uit de renografie** [120-128]. Een radioactief gemerkte stof die door de nieren geklaard wordt, wordt in een ader van de nierpatiënt gespoten. De radioactiviteit over elke nier en in het bloedvat dat de nier bevloeit, wordt opgemeten in functie van de tijd. Door deze activiteit-tijdscurven afkomstig van bloed en nier te deconvolueren, bekomt men de **renale retentiefunctie** die zeer nuttige klinische gegevens bevat over de nierwerking. Resultaten afkomstig uit simulaties en uit klinische gegevens tonen duidelijk de **voordelen van de TKK methode boven het matrixalgoritme**. Deze laatste methode is de meest gangbare deconvolutietechniek voor de berekening van de renale retentiefunctie [34-63-131] en is gebaseerd op terugsubstitutie of Gausseliminatie. Het is welbekend in de lineaire algebra dat deze methode zeer gevoelig is aan fouten op de gegevens. Om foutenvoortplanting zo veel mogelijk tegen te gaan, worden de oorspronkelijke curven **uitgemiddeld**.

We mogen besluiten dat de **TKK methode krachtiger, efficiënter en stabieler** is dan het matrixalgoritme, zelfs indien de curven worden uitgemiddeld. Bovendien kan het matrixalgoritme enkel problemen van **volle rang** aan. Dit betekent dat het faalt in de berekening van de retentiefunctie als de gegevens resulteren in een (bijna) singulier stelsel van vergelijkingen. De TKK techniek daarentegen kan op een eenvoudige manier de rang van de gegevensmatrix verlagen en een betrouwbare retentiefunctie waarborgen.
Bovendien **vereist de TKK methode helemaal geen uitmiddeling** van de curven. Inderdaad, de nauwkeurigheid van de retentiefunctie, berekend met de TKK methode, verbetert nauwelijks door de curven uit te middelen (zie fig. 7.4, 7.5 en 7.6).

C.9. HET GEBRUIK VAN DE TKK BENADERING IN DE SCHATTING VAN DE PARAMETERS IN OVERDRACHTSFUNCTIEMODELLEN

In **hoofdstuk 8** vergelijken we de TKK benadering en de gebruikelijke instrumentele variabelenmethode (IV) met betrekking tot parameterschatting in een **overdrachtsfunctiemodel** :

$$A(q^{-1})\, y_o(t) = B\,(q^{-1})\, u_o(t)$$

$$u(t) = u_o(t) + \Delta u(t)$$

$$y(t) = y_o(t) + \Delta y(t) \qquad \text{(C.7)}$$

$A(q^{-1})$ en $B(q^{-1})$ zijn veeltermen in de achterwaartse schuifoperator q^{-1}. Dit betekent dat $q^{-1}y(t) = y(t-1)$. $u_o(t)$ en $y_o(t)$ zijn de onmeetbare, ruisvrije in- en uitgangen. $u(t)$ en $y(t)$ zijn de beruisde opgemeten in- en uitgangen en $\Delta u(t)$ en $\Delta y(t)$ stellen het gezamenlijk effect voor van alle stochastische perturbaties op de in- en uitgangen respectievelijk.

Beide methoden zijn in staat **sterk consistente** parameterschatters te berekenen. In de IV methode [102-103] wordt de schatter bekomen door de matrix van de waarnemingen langs links te vermenigvuldigen met een geschikte rechthoekige IV matrix. De elementen van deze matrix, **instrumentele variabelen (IV)** genoemd, moeten zo sterk mogelijk **gecorreleerd zijn met de waarnemingen** en **ongecorreleerd met de perturbaties.** Op deze manier bekomt men een **vierkant** stelsel waaruit de **gewone IV schatter** berekend wordt. Door bijkomende IV te nemen, kan de nauwkeurigheid van de IV schatter verbeterd worden. Deze **uitgebreide IV schatter** wordt bekomen als oplossing van een **overgedetermineerd** stelsel van vergelijkingen. De IV methode bestaat in recursieve vorm en is dus zeer geschikt voor "**on-line**" parameterschatting. Dit betekent dat de gegevens ogenblikkelijk kunnen verwerkt worden zodra ze beschikbaar zijn. De **keuze van de IV** blijft echter een **cruciaal** probleem. Meestal bestaan de IV uit vertraagde en gefilterde waarden van de in- en uitgang. De meeste IV zijn enkel geschikt voor ruisvrije ingangen. Nochtans kunnen sommige IV ook consistente schatters berekenen als er ruis op de ingangen is.

De **TKK methode** berekent een sterk consistente schatter van de modelparameters in (C.7) wanneer de uitgangen en mogelijks ook de ingangen geperturbeerd zijn door discrete stationaire witte ruis met gemiddelde nul. De TKK methode kan opgevat worden als een **variante van de IV methode** waarin de IV matrix gegeven wordt door de TKK benadering van de gegevensmatrix, en functie is van het tijdstip t en de modelparameters. In tegenstelling met de IV methoden hoeven **geen IV gegenereerd** te worden. De TKK methode is bijgevolg **eenvoudiger** in gebruik doch **minder algemeen** (IV methoden laten willekeurige ruismodellen toe). Spijtig genoeg kan de TKK oplossing enkel "**off-line**" berekend worden. Dit betekent dat eerst alle gegevens beschikbaar moeten zijn alvorens de berekening kan gestart worden. Hierdoor is de TKK methode minder rekenefficiënt en dus niet zo aantrekkelijk als er veel gegevens zijn.

Tenslotte wordt **door simulaties** de **nauwkeurigheid** van beide methoden met betrekking tot parameterschatting in (C.7) **vergeleken** (zie fig. 8.1 en 8.2). We veronderstellen hierbij dat de in- en uitgangen waargenomen worden gedurende een **beperkt** tijdsinterval (weinig gegevens beschikbaar) en dat de uitgangen en mogelijks ook de ingangen geperturbeerd zijn door **stationaire witte ruis** met gemiddelde nul. In die omstandigheden zijn de vertekening en de variantie van de **TKK oplossing altijd beter** dan die van de **gewone IV schatter**. Bijgevolg berekent de TKK methode schattingen met een kleinere gemiddeld kwadratische fout. Deze fout wordt bij beide methoden **vergelijkbaar door voldoende bijkomende IV** te nemen. De **superioriteit** van de TKK methode is **vooral duidelijk** wanneer de **nulpunten** van de veelterm $A(q^{-1})$ de **eenheidscirkel naderen** of wanneer **zowel de in- als uitgangen onderhevig** zijn aan **ruis.** In andere gevallen berekenen de TKK methode en de IV methode vergelijkbare resultaten.

C.10. RECHTSTREEKSE VERBETERING IN REKENSNELHEID VAN DE TKK BEREKENINGEN

Tot nu toe was onze studie hoofdzakelijk toegespitst op de **nauwkeurigheid** van de TKK methode in problemen met perturbaties en multicollineariteiten. Haar praktisch nut is bewezen in verscheidene toepassingen. Deze analyse van het TKK probleem moet de lezer overtuigen van het belang van de TKK methode voor zijn eigen toepassingen. Niettemin kan hij nog bezwaren hebben. Inderdaad, ondanks het feit dat de TKK benadering

de nauwkeurigheid kan verbeteren, blijft zijn rekentijd onaanvaardbaar lang in toepassingen waarin de rekensnelheid zeer belangrijk is b.v. voor bewerkingen in reële tijd. Daarom trachten we nu **de efficiëntie van het klassieke TKK algoritme te verbeteren.** Dit kan op een **iteratieve** manier (zie hoofdstuk 10) of **rechtstreeks (hoofdstuk 9)** als volgt.
Vermits de TKK oplossing van (C.2) volledig bepaald is door **slechts 1 rechtersinguliere vector** of in het algemeen een basis van de rechtersinguliere deelruimte die hoort bij de kleinste singuliere waarden van [A;B], kan de rekensnelheid aanzienlijk verbeterd worden door de **SWO slechts gedeeltelijk te berekenen.** Hiertoe wordt het klassieke SWO algoritme op 3 manieren gewijzigd. Vooreerst worden de Householdertransformaties van de bidiagonalisatie enkel toegepast op de basisvectoren van de gewenste singuliere deelruimte. Ten tweede wordt de bidiagonaal slechts gedeeltelijk gediagonaliseerd totdat convergentie is opgetreden voor alle singuliere waarden die horen bij de gewenste singuliere deelruimte. En ten derde kan de convergentiesnelheid van de iteratieve diagonalisatie verbeterd worden door een geschikte keuze tussen QR en QL iteratiestappen. Dit verbeterd **"partieel SWO" algoritme PSVD** reduceert de rekentijd aanzienlijk en blijft even nauwkeurig als het klassieke SWO algoritme. PSVD berekent op een efficiënte en betrouwbare manier een basis voor de linker- en/of rechtersinguliere deelruimte van een matrix die hoort bij zijn kleinste singuliere waarden. De dimensie van de gewenste deelruimte kan vastgelegd worden of bepaald worden door een bovengrens op deze kleinste singuliere waarden [129].

Een analyse van het aantal bewerkingen toont de **relatieve efficiëntie** van PSVD ten opzichte van de klassieke SWO. Dit wordt bevestigd door talrijke rekenvoorbeelden. Afhankelijk van de afstand tussen de singuliere waarden die horen bij de gewenste en ongewenste singuliere vectoren, de gewenste numerieke nauwkeurigheid en de dimensie van de gewenste deelruimte, kan **PSVD driemaal sneller** zijn dan het klassieke SWO algoritme. PSVD kan **succesvol toegepast** worden **in de TKK berekeningen** [2] maar ook in de berekening van de **nulruimte** van een matrix en bij het oplossen van stelsels van **homogene** en **niet homogene lineaire vergelijkingen.** In alle toepassingen waarin bovenvermelde problemen optreden kan de betere efficiëntie van PSVD naar waarde geschat worden. Op basis van PSVD is eveneens een zeer efficiënt en betrouwbaar algoritme afgeleid om niet homogene vergelijkingen op te lossen.

Door nu dit partieel SWO algoritme PSVD in de TKK berekeningen te gebruiken, wordt een **verbeterd "partieel TKK" algoritme PTLS** bekomen. Een gedocumenteerd Fortranprogramma van PSVD en PTLS is gegeven in [121]. Een vergelijking in het aantal bewerkingen, alsook talrijke rekenvoorbeelden, bewijzen dat **PTLS efficiënter** is dan het klassieke TKK algoritme. Meestal **daalt de rekentijd met een factor 2.**

C.11. ITERATIEVE VERBETERING IN REKENSNELHEID VAN DE TKK BEREKENINGEN

Met behulp van **a priori informatie**, b.v. de TKK oplossing op een vorig tijdstip, kan de rekentijd van de TKK methode verder dalen. Inderdaad, wanneer traag variërende stelsels van vergelijkingen op elk tijdstip moeten opgelost worden, verschillen de TKK oplossingen van 2 opeenvolgende tijdstippen weinig van elkaar. Met andere woorden, de TKK oplossing op tijdstip t kan gebruikt worden als een **goede initiële schatting** voor de oplossing op t + 1 in een **iteratief** algoritme. **Vier iteratieve** methoden worden in **hoofdstuk 10** vergeleken met betrekking tot hun efficiëntie in TKK problemen. De winst in rekentijd is het grootst voor **één-dimensionale generische TKK problemen** met **unieke** oplossing omdat dan slechts één singuliere vector moet berekend worden.

De **Chebysheviteratie** [134-24], de **inverse iteratie** [82-48, sec. 7.6] en de **Rayleigh quotient** (RQ) **iteratiemethode** [48, sec. 8.5] zijn voornamelijk gebaseerd op de generatie van een goed gekozen rij van functies $f_k(S)$ zodanig dat het matrix-vectorprodukt $f_k(S)q_1$ convergeert naar de gewenste singuliere vector van de matrix S. Hierin is q_1 de startvector van het iteratieproces, die de initiële informatie bevat. De **Lanczosmethode** [48, sec. 9] daarentegen berekent de singuliere vectoren van een matrix niet rechtstreeks maar tri- of bidiagonaliseert de matrix. Hiertoe wordt een rij gegenereerd van tri- of bidiagonaalmatrices met stijgende dimensie waarvan de extreme singuliere waarden convergeren naar de extreme singuliere waarden van de oorspronkelijke matrix. Deze extreme waarden kunnen reeds lang voor het einde van het tri- of bidiagonalisatieproces bereikt worden. Daarom kunnen Lanczosmethoden toegepast worden als een alternatief voor Householderbidiagonalisatie in gevallen waarin slechts enkele van de grootste en/of kleinste singuliere waarden

van de matrix gewenst zijn. De overeenkomstige singuliere vectoren worden dan gemakkelijk berekend met behulp van het QR algoritme of inverse iteratie.

Een analyse van de **convergentieëigenschappen** van deze iteratieve methode toont aan dat **inverse iteratie de meest beloftevolle iteratieve techniek** is om generische ééndimensionale TKK problemen met unieke oplossing op te lossen. Dit is vooral omdat zijn convergentiesnelheid hoofdzakelijk bepaald wordt door de verhouding van de singuliere waarden die horen bij de gewenste en ongewenste singuliere vectoren. Voor de meeste TKK problemen is deze verhouding groot wat een snelle convergentie waarborgt. Dit verklaart waarom RQ iteratie met meer berekeningen per iteratiestap, niet efficiënter is. Bovendien moet de initiële schatting in RQ iteratie voldoende nauwkeurig zijn om convergentie te waarborgen naar de gewenste singuliere vector waaruit de TKK oplossing bekomen wordt. Ook Chebyshev-**iteratie en Lanczosmethoden** presteren niet beter vermits hun convergentiesnelheid ook nog bepaald wordt door de relatieve spreiding van het gekwadrateerd singuliere waardenspectrum. Dit beperkt hun toepasbaarheid tot TKK problemen met een **zeer dicht singuliere waardenspectrum**. Bovendien verminderen **afrondingsfouten** de praktische bruikbaarheid van de Lanczosmethoden. Om voldoende stabiliteit en betrouwbaarheid te kunnen waarborgen, zijn veel bijkomende berekeningen vereist ten koste van efficiëntie.

Op basis van de **convergentiesnelheid** en het **aantal bewerkingen** per iteratiestap, kunnen alle iteratieve methoden **in efficiëntie vergeleken** worden met de rechtstreekse berekeningsmethoden : de klassieke en partiële SWO (zie hoofdstuk 9). De iteratieve methoden zijn het efficiëntst wanneer slechts één singuliere vector moet berekend worden, wanneer hun convergentiesnelheid voldoende hoog is, de gewenste nauwkeurigheid eerder laag en een goede initiële schatting gekend is. Daarom zijn iteratieve methoden vooral aangewezen voor het oplossen van traag variërende TKK problemen vermits dan de oplossing van een vorig stelsel meestal een goede initiële schatting is voor de oplossing van het volgend stelsel.

Tenslotte worden inverse iteratie, het klassieke en partiële TKK algoritme PTLS in efficiëntie vergeleken in een **concreet voorbeeld uit de modale analyse**, namelijk de berekening van **frequentieresponsiefuncties**

[68]. Hiertoe wordt een mechanische structuur gelijktijdig op een aantal plaatsen, ingangen genoemd, geëxciteerd. De responsie wordt opgemeten op een aantal andere plaatsen, uitgangen genoemd. De frequentieresponsiefuncties op elke uitgang worden dan verkregen door zoveel stelsels van vergelijkingen op te lossen als aangegeven wordt door de frequentieresolutie, b.v. 512 stelsels als een gelijke frequentieresolutie van 100/512 Herz vereist wordt over de hele bandbreedte van 100 Herz. Hierbij is een oplossing op een vorig frequentiepunt meestal een goede initiële schatting voor het volgende. De resultaten tonen aan dat inverse iteratie minimaal sneller is dan het partiële TKK algoritme indien slechts 1 uitgang beschouwd wordt. Beide methoden zijn ongeveer tweemaal zo snel als het klassieke TKK algoritme.

De besproken iteratieve methoden kunnen **veralgemeend** worden tot het berekenen van meer dan één singuliere vector of een basis van een singuliere deelruimte. Deze uitbreidingen verminderen echter de efficiëntie. Bovendien moet de **dimensie** van de gewenste deelruimte of het aantal gewenste vectoren steeds **a priori gekend** zijn. Dit is niet nodig bij de **rechtstreekse berekeningsmethoden**.
Vermits de **efficiëntie** van deze laatste methoden bovendien **minder probleemafhankelijk** is, is hun toepassingsgebied **ruimer** en worden ze meestal verkozen boven de iteratieve methoden in problemen met een **matig** aantal vergelijkingen.

C.12. BESLUITEN

Elk lineair parameterschattingsprobleem resulteert in een **overgedetermineerd** stelsel van lineaire vergelijkingen AX = B. Wanneer zowel de gegevensmatrix A als de waarnemingsmatrix B onderhevig zijn aan fouten, is de TKK methode aangewezen om dit stelsel op te lossen. In de eerste 5 hoofdstukken van deze thesis worden verschillende aspecten van het TKK probleem geanalyseerd. De **mogelijke TKK problemen** met hun **rekenkundige aspecten** worden besproken en de **eigenschappen** van de TKK benadering worden vergeleken met die van andere bestaande technieken. De volgende 3 hoofdstukken bestuderen de **praktische toepasbaarheid** van de TKK benadering in verschillende problemen. De laatste 2 hoofdstukken tenslotte to-

nen aan hoe de **efficiëntie** van de TKK berekeningen kan verbeterd worden. We geven nu een overzicht van de voornaamste bijdragen van deze thesis, alsook suggesties voor verder onderzoek.

C.12.1. OVERZICHT VAN DE VOORNAAMSTE BIJDRAGEN

Een **eerste bijdrage** van dit werk is de **veralgemening van de TKK methode** tot alle gevallen waarin het algoritme van Golub en Van Loan **faalt** in de berekening van de TKK oplossing. Onze analyse is hoofdzakelijk gebaseerd op de eigenschappen van de krachtige SWO en leidt tot de ontwikkeling van een **veralgemeend TKK algoritme** dat alle uitbreidingen omvat (hoofdstuk 1).

Een tweede belangrijk punt in de analyse van de TKK methode is zijn **geldigheid** of anders gezegd, het toepassingsgebied van de TKK oplossing. Dit vergt een grondige studie van **al zijn eigenschappen** voor alle mogelijke voorwaarden die (kunnen) optreden. Een vergelijking met andere bestaande methoden belicht hierbij beter de betekenis en de toepasbaarheid van de TKK methode. Ons antwoord op deze geldigheidsvraag vormt de **tweede hoofdbijdrage** van dit werk. Niet alleen de **algebraïsche** eigenschappen van de TKK methode worden grondig geanalyseerd en vergeleken met deze van de klassieke KK methode (hoofdstuk 2). Doch ook de **statistische** eigenschappen (hoofdstuk 4) komen aan bod, alsook de verbanden met de klassieke methoden in **lineaire regressie** (hoofdstuk 5). Bovendien wordt de **gevoeligheid** van de TKK benadering bestudeerd en vergeleken met deze van de klassieke KK benadering voor verschillende perturbatieëffecten op de gegevens. Dit gebeurt in een theoretische gevoeligheidsanalyse. De resultaten van deze studie worden bevestigd in simulaties (hoofdstuk 3).

Een derde belangrijk aspect dat dient onderzocht te worden is de **praktische toepasbaarheid** van de TKK benadering in het oplossen van uiteenlopende problemen in verschillende wetenschapsgebieden zoals signaalverwerking, systeemtheorie, geneeskunde, ... Dit vereist genoeg inzicht in de eigen noden en specifieke problemen binnen elk gebied. Op basis van deze informatie kan de **beste strategie** ontwikkeld worden om het probleem volgens de TKK benadering op te lossen. In dit werk worden zulke **procedures** ontworpen om verschillende praktische problemen op te lossen, b.v.

voor de selectie van een deelmodel (hoofdstuk 6) en in systeemidentificatie voor het schatten van de impulsresponsie door discrete convolutie (hoofdstuk 7) en de parameters in een overdrachtsfunctiemodel (hoofdstuk 8). Deze worden dan vergeleken met de meest gangbare oplossingstechnieken in elk domein. Hun **performantie** wordt experimenteel geëvalueerd door simulaties of door concrete gegevens. Dit vormt de **derde bijdrage** van dit werk.

En tenslotte speelt **de efficiëntie** van de TKK berekeningen een belangrijke rol in tal van toepassingen, b.v. voor bewerkingen in reële tijd. Een zorgvuldige analyse van de TKK oplossing en de SWO laat toe de rekensnelheid van het klassieke TKK algoritme **rechtstreeks** te verbeteren. Dit resulteert in de ontwikkeling van een nieuw, efficiënter en even betrouwbaar algoritme (hoofdstuk 9). Indien bovendien **a priori informatie beschikbaar** is (b.v. de TKK oplossing op een vorig tijdstip wanneer traag variërende stelsels van vergelijkingen moeten opgelost worden op elk tijdstip), kan de rekensnelheid verbeterd worden door de TKK oplossing **iteratief** te berekenen (hoofdstuk 10). Een diepgaande analyse van de **convergentieëigenschappen** en het **aantal rekenkundige bewerkingen** van de bestaande iteratieve methoden laat toe de efficiëntie en toepasbaarheid van deze methoden in de TKK berekeningen te evalueren en maakt een vergelijking met de rechtstreekse berekeningsmethoden mogelijk. Deze analyse biedt de gebruiker de mogelijkheid **de meest efficiënte methode** te kiezen overeenkomstig zijn eigen noden en probleemkarakteristieken. Dit vormt de **vierde hoofdbijdrage** van deze thesis.

C.12.2. SUGGESTIES VOOR VERDER ONDERZOEK

Ondanks het feit dat deze thesis volledig aan het TKK probleem gewijd is, vergen bepaalde aspecten van het probleem nog verder onderzoek.

- Experimenten tonen aan dat de **TKK methode gevoelig is voor scalering van de gegevens**. Om de optimale eigenschappen van de TKK oplossing (zie hoofdstuk 4) te bekomen, moeten de gegevens $[A;B]$ zo gescaleerd zijn dat de fouten op elke rij onafhankelijk en identiek verdeeld zijn met gemiddelde nul en gemeenschappelijke covariantiematrix van de vorm $\sigma^2 I$ (I is de identiteitsmatrix). Enkele aanbevelingen worden

gegeven in sectie 1.8.1. Niet alleen moet de foutenverdeling gekend zijn om de juiste scaleringsfactoren te berekenen, doch het is niet altijd mogelijk om de foutenmatrix op gepaste wijze te transformeren door rij- en kolomscalering. Dit is het geval bij niet lineaire verdelingen. Ondanks het feit dat de optimale eigenschappen van de TKK oplossing in dit geval niet bereikt worden, kunnen ze in sommige concrete toepassingen nog goed benaderd worden zodat de TKK benadering zinvol blijft. Niettemin is een **meer fundamentele studie van het scaleringsprobleem** aangewezen om meer **robuuste scaleringsstrategieën** te ontwerpen en aldus het toepassingsgebied van de TKK methode uit te breiden.

- De TKK benadering vereist een **betrouwbare rangbepalingsmethode** om de juiste rang van de gegevensmatrix $[A;B]$ te berekenen en het al dan niet singulier zijn van de deelmatrix Γ in (1.41) te testen. De betrouwbaarste methode is de SWO. De rang volgt rechtstreeks uit het aantal singuliere waarden groter dan een bepaalde foutenafhankelijke grens. De berekening van deze grens in de TKK algoritmen voorgesteld in deze thesis, berust op de maximale minimale signaal-ruisverhouding [105] maar andere strategieën zijn mogelijk. Een **dieper inzicht in de ruiseffecten en de ruisverdeling op de SWO** van een matrix laat toe meer verfijnde rangbepalingsmethoden te ontwerpen.

- De **TKK methode vernietigd de structuur** van een matrix. Inderdaad, de TKK benadering $[\hat{A};\hat{B}]$ van de gegevens $[A;B]$ is niet langer gestructureerd als $[A;B]$. Bijvoorbeeld in systeemidentificatie verliezen problemen hun Toeplitz- of Hankelstructuur door de TKK benadering. Verder onderzoek is vereist om de **TKK criteria uit te breiden** wanneer de structuur in de berekende benadering $[\hat{A};\hat{B}]$ moet behouden blijven.

- De **wijzigingen in de klassieke SWO** die resulteren in een efficiënter partieel SWO en TKK algoritme kunnen ook **toegepast worden op de veralgemeende SWO en op de (veralgemeende) symmetrische eigenwaardenontbinding** in zoverre hun algoritmen gebaseerd zijn op het QR algoritme. Hierdoor kan de efficiëntie van deze algoritmen verbeterd worden bij de berekening van een basis van de singuliere deelruimten of eigendeelruimte die hoort bij de kleinste veralgemeende singuliere waarden of (veralgemeende) eigenwaarden. Ook zijn slechts kleine wijzigingen

in het verbeterd algoritme nodig om een basis te berekenen van de deelruimte van een matrix die hoort bij zijn **grootste** singuliere waarden of eigenwaarden.

- Een andere suggestie is de **uitbreiding van de TKK berekeningen tot het complexe veld.** Dit laat toe de complexe TKK oplossing efficiënter te berekenen vermits transformaties van de gegevens naar het reële veld dan overbodig zijn. Dit is b.v. het geval bij de berekening van de frequentieresponsiefuncties in modale analyse (zie sectie 10.8). Deze uitbreiding **opent nieuwe perspectieven** voor de TKK methode in **complexe** problemen.

- Verder zijn de **TKK algoritmen** in deze thesis **niet zo aantrekkelijk voor ogenblikkelijke verwerking van de gegevens ("on-line").** Inderdaad, dit vereist dat de oplossing onmiddelijk kan herberekend worden met een minimaal aantal bijkomende bewerkingen zodra nieuwe gegevens beschikbaar zijn. Dit is niet het geval voor de TKK algoritmen in deze thesis. Deze vereisen immers dat de berekeningen van de TKK oplossing volledig opnieuw uitgevoerd worden. De ontwikkeling van een **efficiënte "on-line" versie** voor de TKK berekeningen opent beslist nieuwe toepassingsgebieden voor de TKK methode, b.v. in signaalverwerking.

- Tenslotte toont het gebruik van de TKK benadering in de selectie van een deelmodel (hoofdstuk 6), in systeemidentificatie (hoofdstuk 7 en 8) en in renografie (sectie 7.3) de **voordelen** van de TKK methode wat betrouwbaarheid, nauwkeurigheid en efficiëntie betreft en bewijst zijn praktische toepasbaarheid. Ook in andere interessante problemen van modale analyse [68], inverse verstrooiing [99] en frequentieschatting [85] is het nut van de TKK benadering bewezen. Niettemin kan de **TKK benadering nog in tal van andere, niet besproken problemen de kwaliteit van de resultaten verbeteren.** We hopen dat dit werk elke lezer zal in staat stellen en stimuleren om de TKK benadering toe te passen op zijn concrete problemen.