KATHOLIEKE UNIVERSITEIT TE LEUVEN

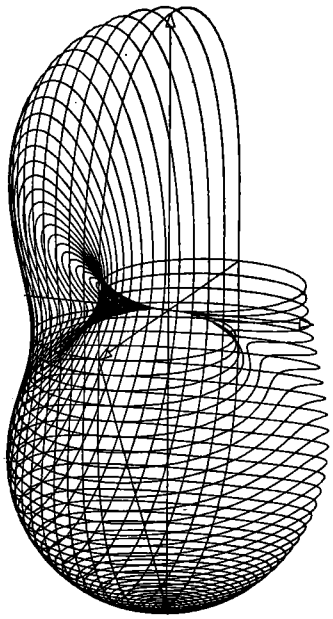
FAKULTEIT DER TOEGEPASTE WETENSCHAPPEN

Departement Electrotechniek

Afdeling E.S.A.T.

CONCEPTS FOR RELIABLE MODELLING OF LINEAR SYSTEMS WITH APPLICATION TO ON-LINE IDENTIFICATION OF MULTIVARIABLE STATE SPACE DESCRIPTIONS

ir. J. STAAR



Werk voorgedragen tot het bekomen
van de graad van
Doctor in de Toegepaste Wetenschappen

Promotor : Dr. Ir. J. Vandewalle

Juni 1982

VOORWOORD.

Dit doctoraatsproefschrift is het resultaat van onderzoek, verricht in de systeemgroep van de afdeling ESAT aan het departement Electrotechniek van de fakulteit Toegepaste Wetenschappen, KULeuven.

Een doctoraat is zoals het krijgen van een eerste tand: het kan blijkbaar niet zonder pijn. Tanden komen echter vanzelf. 't Zou gelogen zijn te beweren dat dit werk van een leien dakje is gegaan en 't zou oneerlijk zijn de steun te vergeten die dit voorwoord nodig heeft gemaakt.

Zonder mijn ouders was ik niet geweest, en had ik nooit ingenieursstudies aangevat.

Zonder mijn promotor J. Vandewalle was ik blijven lezen, narekenen, twijfelen en dromen. Van hem heb ik geleerd dat ook wetenschappelijke kennis ordelijk kan worden opgebouwd. Dat een degelijk uitgewerkt eenvoudig idee tot meer leidt dan tien wilde bespiegelingen... En dat er mensen zijn, die zelfs om 11 uur 's avonds systeemtheorie nog interessant blijven vinden.

Aan prof. R. Van Overstraeten ben ik dank verschuldigd voor het ongewoon hoog wetenschappelijk niveau van zijn ESAT laboratorium, voor de steun en het vertrouwen op de ogenblikken dat ik die nodig had, voor zijn kalm, zelfzeker en kristalhelder doorzicht, en voor zijn konsekwent beleid. Een voorbeeld dat in mijn hoofd is gegrift.

Ik dank het NFWO en de leden van jury, die mijn mandaat hebben gesteund, voor het vertrouwen en voor de onberispelijke administratieve behandeling van mijn dossier.

Ik dank de leden van het leescomite voor hun grondige maar soepele doorname van het manuscript. En ik dank de leden van de jury voor hun bijzondere interesse voor het voorgebrachte werk.

Marcel, U ben ik veel verschuldigd. ADAPT, waarrond zoveel heeft gedraaid, waarmee zoveel is uitgetest: het is Uw werk. Ik heb de beste herinneringen aan onze samenwerking, ik hoop dat het U plezier zal doen de laatste ontwikkelingen van AIDENT in dit werk terug te vinden.

Ik mag Frans Claeys en Rik Debock niet vergeten: de verzorgde implementatie van ASOLVE, de axonometrische plots in deze tekst, en het onvoorwaardelijk enthousiasme waarmee zij hun eindwerk hebben uitgevoerd, waren voor mij een aangename verrassing: er zijn nog echte studenten.

En dan de hele bende met Lou, Jan, Mia, Yvo, Robert, Guy, et les autres. Voor het goed humeur en de zwanzerijen, die mij het relatieve van al die matrix- en andere algebra hebben bijgebracht.

En dan kom ik tenslotte bij U, Mieke. Maar dat... is nog een heel ander verhaal.

Heverlee, juni 1982.

TABLE OF CONTENTS

PREFACE

INTRODUCTION

PART I : THE SINGULAR VALUE DECOMPOSITION

Chapter 1 Basic facts.

- Historical note.
- Classical results on existence and uniqueness.

Chapter 2 A conceptual analysis.

- Geometric properties of a matrix, seen as a linear operator.
- Energetic properties of columns or rows in a matrix.
- Stochastic properties of columns or rows in a matrix.
- Signal-to-Signal and Signal-to-Noise Ratios.
- Some more practical concepts, related to SVD.

PART II : NUMERICALLY RELIABLE TECHNIQUES AND ALGORITHMS.

Chapter 3 SVD-properties of sensitivity and numerical stability.

- Sensitivity properties of the SVD.
- Numerical computation of the SVD.
- Sensitivity properties of a set of linear equations.
- Numerical computation of tne solution of sets of linear equations.

Chapter 4 On Global Linear Least Squares approximation.

- On minimizing $\|A\xi - y\|$.
- The Global LLS approach.
- On the computation of GLLS solutions.
- Conclusion.

Chapter 5 ASVD: a new algorithm for the recursive computation of the Singular Value Decomposition of large low-rank matrices.

- General description of the algorithm.
- Illustration and proof of theorem 5-1.
- Some additional comments.
- Accuracy and convergence of the algorithm.
- An acceleration method.
- Storage requirements and operation count.
- Conclusions.

Chapter 6 ASOLVE: an new algorithm for the recursive computation of the LS solution of a set of linear equations.

- General description of the algorithm.
- A first evaluation.
- Refinement of the basic algorithm.
- Conclusion.

PART III : NUMERICALLY RELIABLE RESULTS IN LINEAR SYSTEM THEORY.

Chapter 7 On the numerical implications of the choice of a reference node in the nodal analysis of large circuits.

- Introduction.
- Main results.
- Examples.
- Conclusion.

Chapter 8 On the relationship between Singular Value Decomposition, and the Principal Axes and Moments of Inertia of a rigid body

- Introduction.
- Basic theorem and some examples.
- Computational aspects.
- Conclusion.

Chapter 9 On numerically reliable realization.

- Realization of exactly realizable truncated impulse responses.
- Types of solution strategies.
- The approximation problem.
- Feasibility analysis.
- Examples.

Chapter 10 On adaptive on-line identification of a linear State Space Representation from measured input-output data.

- On a General Purpose, Multivariable Adaptive Control scheme.
- A some more detailed discussion of SSR identification.
- Implementation and examples.

CONCLUSION

REFERENCES

APPENDIX: SOME BASIC DEFINITIONS AND GENERAL NOTATIONS

INTRODUCTION

This thesis is about linear modelling. One more, you could think. And you would be right if it were indeed the N-th scrupulously detailed report on m among the M ever reported methods available today in the literature ... with of course a concluding variant M+1, freshly invented for the occasion and conscienciously added to the list of M-m remaining algorithms, to be evaluated one day by whoever would have the courage -and the patience- to continue this encyclopaedic exercise in scientific archeology.

In contrast, this thesis is about concepts.

The reason is physiological. Being informed that I would have to proofread at least once my own thesis, my stomach sent me the warning that it would not digest one more formula crunching piece of literature until it had recovered completely from its chronical algebraitis linearis of the past few years. I promote courage, but in face of such a threat, I surrendered. And frankly speaking, I think you ought to be grateful to my stomach.

I have always admired the fascinating conceptual power of linear algebra. It lead to the succesful structural analysis of high-degree, multivariable systems, to the Kalman filter, the Luenberger observer and Optimal Multivariable Controller. At the same time I hate its leniency to meaningless formal manipulation, which lead to the publication of an uncountable number of algebraically correct, but conceptually obscure, insignificant and cumbersome equivalent formulations, to the satisfaction and glory of their authors, and to the confusion, misunderstanding, bore, anger, frustration, discouragement and apathy of their (unexperienced) readers.

The main objective of this document is not to exhibit this kind of formal ability. Instead, the author will try to share with the reader some essential ideas, concepts, and conclusions that appeared important to him in his work and study of linear modelling, during the past four years. Most results are on the numerical aspects of computations related to linear modelling, an area considered long as noman's land inbetween system theory and computer science. An embarrassing aspect in this discipline is, that its subject is often directly related to well established algebraic modelling theory and pure numerical analysis. Clearly, the final result will rely on concepts, familiar at least to one of these fields. But the difficulty is precisely in finding the proper link, unifying the two approaches into one global mathematical construct, that meets the "standards of good practice" of BOTH algebraic system theory and numerical analysis.

In many cases, this is not trivial.

More: an essential concept may be hidden behind an apparently insignificant formal trick. Usually one does not appreciate the importance of such a "detail" until it smashes a few noses. Therefore, we will regularly entertain the reader with small numerical disasters: unsuspected difficulties, encountered in a larger modelling context. Usually they are the ones that initially confused the author, but finally made him understand the correct way to look at a general type of problems that regularly appear in modelling procedures. It is hoped that the reader will experience the same pleasure, and gain as much insight as did the autnor, when he went through the whole procedure of surprise, unbelief, confusion... and AHA-erlebnis of the simple concepts on which the solution could finally be based.

Let me now give a review of the subjects that will be treated.

1. The first part will completely be devoted to the concept of SINGULAR VALUE DECOMPOSITION (SVD), in itself. SVD is an eigenvalue-like concept, but with remarkable geometric and numerical qualities. The basic purpose is to familiarize the reader with the inherent power of this concept in a linear modelling context. For this, we will recall some important classical results, and add a few new ones.

   After a brief historical note, and a short review on classical existence and uniqueness properties in chapter 1, a conceptual analysis is presented in chapter 2. Fundamental SVD-related geometric properties of a matrix will be described: the classical interpretation of the SVD of a matrix seen as a linear operator, and a new interpretation of the SVD of a matrix seen as a sequence of (column- or row-) vectors. For the latter, the concepts of "oriented energy" and "Minimal Signal-to-Signal Ratio" for vector sequences, observed from a subspace, will be introduced, and their fundamental properties will be established. Some more common properties will be mentioned along.

2. Part two is devoted to the numerical aspects of some fundamental linear problems.

   Chapter 3 reviews the sensitivity properties of SVD and of the solution of a set of linear equations. The essential difference between the condition of these problems, and the numerical stability of the algorithms available to solve them, is discussed in the light of the conceptual analysis of chapter 3. A nice pictural view at the classical 2-norm condition number of a MATRIX, and the 2-norm condition number of a SET OF EQUATIONS will readily follow from this analysis.

   Chapter 4 discusses a new Linear Least Squares approximation technique for the solution of an overdetermined set of linear equations. The key idea is that for sets of equations where both the matrix and the Right Hand Side are inaccurate, a global minimization of the reconstruction - error on the matrix TOGETHER with the RHS is more meaningful than minimizing the reconstruction error on the RHS only. Evidence of this statement will be given in a simple example, and the computation of Global Linear Least Squares solutions will be discussed.

   Finally, we will introduce two new algorithms. Chapter 5 presents ASVD, an algorithm particularly suitable for the computation of partial Singular Value Decompositions of sequences of slowly changing low rank matrices. Chapter 6 presents ASOLVE that is particularly suitable for the computation of the LLS solutions of slowly varying sets of linear equations. Both algorithms guarantee a reliability equal to the classical SVD or SVD-based algorithms; both continuously USE the elements of the matrix, WITHOUT CHANGING THEM (allowing tremendous storage cut downs when the matrix is highly structured); both continuously UPDATE the solution in an iterative manner (allowing efficient adaptive implementation when successive sets with close solutions have to be solved).

   These algorithms are the key ingredients in the on-line identification algorithm AIDENT of chapter 10. They will contribute to the improvement of speed-, and storage-efficiency with a factor $\alpha.N$ over previous algorithms of the same type and reliability. (N is the amount of measured data permanently in memory for the identification.)

3. The third part in this thesis is devoted to three different domains of application.

Chapter 7 discusses the numerical implications of the choice of a reference node in the nodal analysis of large circuits: ALGEBRAICALLY, an indeterminate description (in which no node is chosen as a reference) is strictly equivalent to a determinate description (in which one node is taken as the ground, and disappears in the description). NUMERICALLY however, one description may be ill conditionned whereas the other may be well conditionned. The type of description that will lead to the most reliable analysis depends on the circuit itself. We discuss this numerical phenomenon, derive tight bounds on the possible variation in condition from one description to another, and illustrate the idea with two extreme examples. The final question of course is how to determine, from a preliminary analysis of the circuit, the type of description to be used for a reliable analysis. This question remains unanswered.

An application in the domain of analytical mechanics is found in chapter 8. It is well known that the Principal Axes and Moments of Inertia (PAMI) of a rigid body with respect to a fixed point O are the eigenvectors and eigenvalues of its inertia tensor in an orthogonal basis with origin O. This simple definition, together with the fact that the inertia tensor is symmetric and semidefinite positive (which guarantees a numerically stable eigenstructure decomposition), has lead to the widespread misconception that the eigenstructure decomposition of the inertia tensor is the indicated way to compute the PAMI of a rigid body.

This is incorrect.

The misconception is based on the implicit assumption that the definition of the PAMI as the eigenstructure of the inertia tensor is the only possible one. This is not so, and unfortunately the computation of the inertia tensor is a numerically unstable operation. We will prove that there exists an algebraically equivalent SVD-based formulation for the characterisation of the PAMI, which DOES allow stable computation, and we will indicate its most appropriate variants for several practical situations. The philosophical conclusion of this chapter should be that reliable algebra is guaranteed only if its development is guided by numerical arguments from its earliest statements on.

The remaining two chapters belong to the domain of multivariable automatic process control. Chapter 9 treats a problem that has been the authors concern for quite a long time: the reliable realization of a State Space representation from a given discrete-time impulse response sequence. Chapter 10 considers the problem of the complete identification of a State Space Representation, directly from the measurement of its (multivariable) input-output behaviour.

The attentive reader will see most of the results obtained in the first six chapters reappear in chapters 9 and 10. This is no coincidence, and it is the author's hope that their joint application in a unifying context will help the reader to appreciate the relative significance of each of the ideas, developped in this thesis.

The resulting algorithm AIDENT is believed to be the first on-line model-tracking procedure for multivariable State Space Realizations, that is implementable on a standard microcomputer, even for model degrees up to 15.

Chapter 1

THE SINGULAR VALUE DECOMPOSITION: BASIC FACTS.

The basic purpose of this chapter is to familiarize the reader with the Singular Value Decomposition (SVD), which is an eigenvalue-like concept, with remarkable geometric and numerical qualities. In this first chapter, we present a review of the main classical results, as an introduction to the more conceptual analysis of chapter 2.

A. HISTORICAL NOTE.

The idea of Singular Value Decomposition is quite old. It seems that its existence for real square matrices was examined around 1877 by Jordan and Beltrami [1]. Its formal statement, generalized to complex matrices, was established by Autonne in 1902 [2], and its final formulation for general MxN complex matrices was finally achieved by Eckart and Young in 1939 [3].

Yet, it was not until the recent work of Golub, that the numerical and algorithmic problems, related to the reliable finite-precision computation of an SVD, were described [4] and finally solved [5].

Today, its availability as reliable standard software [10] has promoted its use in linear system theory [11]: besides direct application in the geometric approach to linear state space representations [13], the singular values of the transfer matrix of a system with multiple inputs and outputs have recently been shown to be of critical importance in the design of robust controllers [12, 5 papers].

In this thesis, we will show its usefulness in the analysis of numerical qualities of certain types of representations of electrical circuits ([110], and chapter 7); in the computation of Principal Axes and Moments of Inertia (PAMI) for mechanical systems ([111], and chapter 8); in the quantization of the persistent excitation of a multivariable input signal for identification purposes ([102],[105], and chapter 10); and in the implementation of reliable realization ([101, 103, 106] and chapter 9) and identification ([102, 104, 114] and chapter 10) algorithms.

B. CLASSICAL RESULTS ON EXISTENCE AND UNIQUENESS.

In this section, we don't yet bother about conceptual interpretation. (See chapter 2). We merely state some basic algebraic facts in a purely formal setting. The definitions, conventions and notations for matrices, their manipulation, and their properties are the usual ones. If necessary, the reader can consult the general overview in the Appendix.

Theorem 1-1 (The Autonne-Eckart-Young theorem):

For any COMPLEX matrix A, there exists a COMPLEX factorization

$$A_{MN} = U_{MM} \Sigma_{MN} V_{NN} \qquad (1\text{-}1)$$

in which the factor Σ is REAL, PSEUDODIAGONAL, NONNEGATIVE DEFINITE, and the factors U and V are COMPLEX, HERMITIAN. (Indices usually denote the dimensions of the matrix, but are omitted if not strictly necessary).

□

This formulation for complex matrices is included for completeness (it is the most general formulation, as stated by Eckart and Young [3]), and because of its conceptual beauty. However, in the remainder of this thesis, all statements and algorithms will be restricted to REAL matrices:

Corollary 1-1 (restriction to real matrices):

For any REAL matrix A, there exists a REAL factorization

$$A_{MN} = U_{MM} \Sigma_{MN} V_{NN} \qquad (1\text{-}2)$$

in which the factor Σ is REAL, PSEUDODIAGONAL, NONNEGATIVE DEFINITE, and the factors U and V are REAL, ORTHONORMAL.

□

Nomenclature

- The elements σ_{pp} in Σ, often denoted as σ_p, are called SINGULAR VALUES of matrix A, and their set is called the SINGULAR SPECTRUM $\sigma(A)$ of A.
- The COLUMNS u_m of U and the ROWS v_n of V are called the left resp., right SINGULAR VECTORS of matrix A, and their set is called its left resp., right SINGULAR BASIS.
- The 3-couples (u_p, σ_p, v_p) are called the singular triplets of matrix A.
- The spaces

  $$S_U^p = \text{Span} (u_1 , \ldots , u_p) \qquad (1\text{-}3a)$$

  $$S_V^p = \text{Span} (v_1 , \ldots , v_q) \qquad (1\text{-}3b)$$

  are called the p-th principal left resp., q-th principal right singular spaces.

Example 1-1: the SVD of a real 2x2 matrix, with 3 decimals:

$$\begin{bmatrix} 0.578 & -0.224 \\ 1.190 & 0.837 \end{bmatrix} = \begin{bmatrix} 0.259 & -0.966 \\ 0.966 & 0.259 \end{bmatrix} \cdot \begin{bmatrix} 1.5 & 0 \\ 0 & 0.5 \end{bmatrix} \cdot \begin{bmatrix} 0.866 & 0.500 \\ -0.500 & 0.866 \end{bmatrix} \qquad (1\text{-}4a)$$

is formally represented as:

$$A = U \cdot \Sigma \cdot V \qquad (1\text{-}4b)$$

The singular values are 1.5 and 0.5. The matrices U and V are orthonormal:

$$U^t.U = U.U^t = I \quad \text{and} \quad V^t.V = V.V^t = I \qquad (1\text{-}5)$$

Corollary 1-2 (on the existence of the DYADIC decomposition):

$$A = \sum_{p=1}^{P} \sigma_p \, u_p \, v_p \qquad (1\text{-}6)$$

with P = Min(M,N). Eq.(1-6) shows that one can, with no loss of generality, assume the singular values to be ordered in decreasing magnitude:

$$\sigma_1 \geqslant \sigma_2 \geqslant \cdots \quad \sigma_P \geqslant 0 \qquad (1\text{-}7)$$

□

Theorem 1-2 (on algebraic uniqueness):

a. The singular spectrum is unique.

b. The singular bases are unique iff all singular values are distinct.

c. For a set of p equal, NONZERO singular values, the associated left and right singular subspaces are unique, i.e.

if $$\sigma_{p_1} = \sigma_{p_1+1} = \cdots = \sigma_{p_1+p-1} \neq 0 \qquad (1\text{-}8a)$$

then the spaces $$S_U^p = \text{Span}(u_{p_1}, u_{p_1+1} \cdots u_{p_1+p-1}) \qquad (1\text{-}8b)$$

and $$S_V^p = \text{Span}(v_{p_1}, v_{p_1+1} \cdots v_{p_1+p-1}) \qquad (1\text{-}8c)$$

are unique. Within these spaces, the singular bases can be chosen freely, provided they are orthonormal, and mutually related by:

$$u_k^t \, A = \sigma_k \, v_k \qquad k = p_1, \ldots, p_1+p-1 \qquad (1\text{-}9)$$

d. In the case of p ZERO singular values, both left and right subbases can be chosen freely within their (unique) associated left resp., right singular subspaces.

□

Proofs of the above classical existence and uniqueness theorems are found in [2,3] (originals), [14] (numerical context), [11] (control application context) and [8] (didactical context). They are not repeated here.

The uniqueness theorem is stated in a typically algebraic form. The abrupt transition from uniqueness to non-uniqueness, when two singular values come close, and finally become equal, is a severe numerical weakness. Therefore we will reconsider this problem in a numerically more meaningful way in chapter 3.

Chapter 2

THE SINGULAR VALUE DECOMPOSITION: A CONCEPTUAL ANALYSIS.

In this second chapter we will complete the review of the main classical SVD properties, which were described in chapter 1, and add some new results. Conceptual and practical significance will follow from considering their interpretation from different viewpoints, that will appear to be of particular interest for our modelling purposes in chapters 7 to 10.

A. GEOMETRIC PROPERTIES OF A MATRIX, SEEN AS A LINEAR OPERATOR.

A most illustrative way of looking at SVD is by considering the linear map associated to the matrix A of example 1-1 of chapter 1 (Fig. 2-1):

$A : S^2 \rightarrow S^2 : x \rightarrow y=Ax$, factored as

$$U \cdot \Sigma \cdot V \cdot x = y \qquad (2\text{-}1a)$$

$$\begin{bmatrix} 0.259 & -0.966 \\ 0.966 & 0.259 \end{bmatrix} \cdot \begin{bmatrix} 1.5 & 0 \\ 0 & 0.5 \end{bmatrix} \cdot \begin{bmatrix} 0.866 & 0.500 \\ -0.500 & 0.866 \end{bmatrix} \cdot \begin{bmatrix} 0.6 \\ 0.8 \end{bmatrix} = \begin{bmatrix} 0.167 \\ 1.383 \end{bmatrix} \qquad (2\text{-}1b)$$

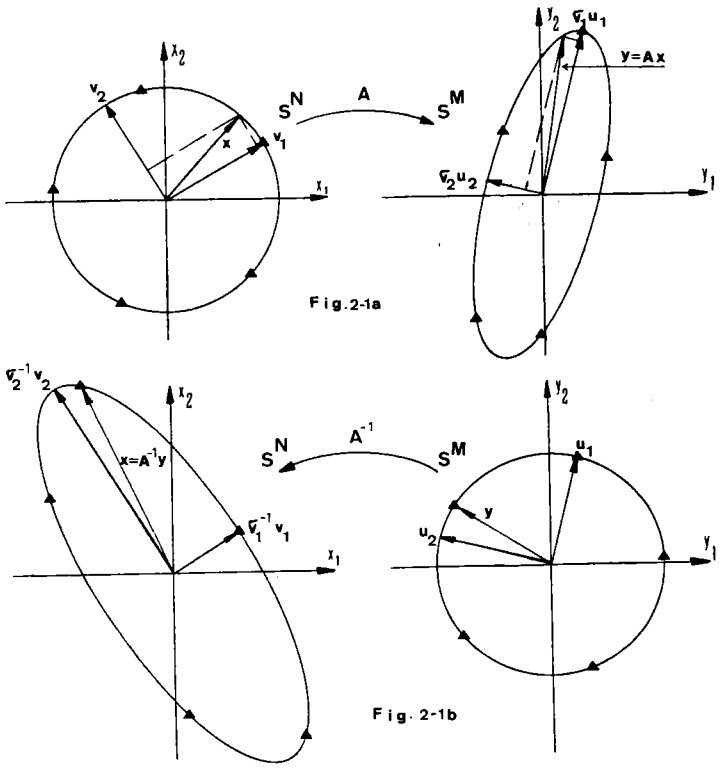


Fig.2-1: Geometric illustration of the SVD properties of a linear map A.

Fig.(2-1) and eq.(2-1) can be combined into the following interpretation:

- Multiplication with V decomposes vector x in the orthonormal right singular basis v_n, n=1 ... n=N=2.

  Indeed : $(Vx)_n = v_n x$ is the orthonormal projection of x on v_n (2-2)

- Multiplication with Σ stretches the obtained components along this basis with the singular values as scaling factors.

  Indeed : $(\Sigma Vx)_p = \sigma_p . (v_p x)$, p=1 ... p=P=Min(M,N)=2 (2-3a)

  and possibly: $(\Sigma Vx)_n = 0 . (v_n x)$, n=P+1 ... n=N (2-3b)

- Multiplication with U finally reconstructs the vector ΣVx, component by component, in the orthonormal left singular basis u_m, m=1 ... m=M=2.

  Indeed : $y = U.(\Sigma Vx) = \sum_{p=1}^{P} u_p . (\sigma_p v_p x) + \sum_{m=P+1}^{M} 0 . u_m$ (2-4)

In conclusion, the set of ALL possible vectors of unit length, forming a HYPERSPHERE IN THE RIGHT SINGULAR BASIS V, is transformed into a HYPERELLIPSOID, DESCRIBED IN THE LEFT SINGULAR BASIS U by the successive actions of orthogonal decomposition, stretching, and orthogonal reconstruction. The p-th main axis of the resulting hyperellipsoid is oriented along the p-th left singular vector, and has a length precisely equal to the p-th singular value.

If A is square and nonsingular, the inverse map can be represented and analysed in a completely analogous fashion (Fig. 2-1b). For a unit hypersphere in the range space, one obtains a hyperellipsoid in the solution space, with main axes $\sigma_1^{-1} . v_1$ and $\sigma_2^{-1} . v_2$.

The decomposition of the map A: $x \rightarrow y$ and its inverse A^{-1}: $y \rightarrow x$ in this manner is well known and appreciated in the litterature [8]. It is a very powerful conceptual tool in the analysis of (generalized) inversion, conditioning and rank of a set of linear equations, as will be illustrated in chapter 3. It is also the basis of our algorithm ASVD in chapter 5.

However, the above interpretation of SVD is not the only possible one. Singular Value Decompositions of matrices, seen as collections of column vectors, can provide very interesting information on the "oriented energy distributions" of these sets of vectors, and on the "oriented signal-to-signal and signal-to-noise ratios" relating two sets of vectors. The next section introduces these new concepts, and establishes the link with SVD.

B. ENERGETIC PROPERTIES OF COLUMNS OR ROWS IN A MATRIX.

Any finite set of vectors (e.g. a sequence of N samples of an M-vector signal) can be put together as the columns in an MxN matrix:

$$A \triangleq \left[a_1 \quad a_2 \quad \ldots \quad a_N \right] \qquad (2\text{-}5)$$

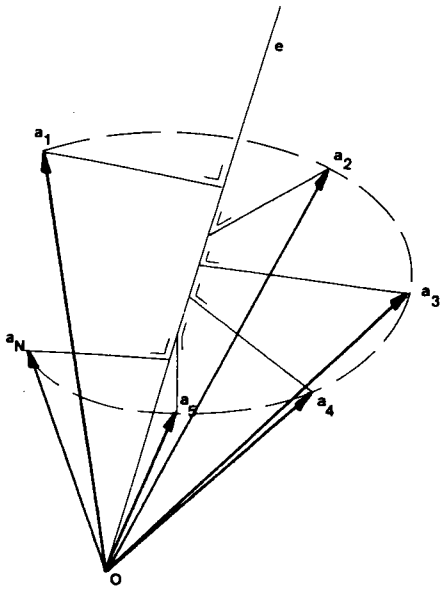


Fig.2-2: Illustration of oriented energy measurement.

The sum of the squared orthogonal projections of these vectors, on any unit VECTOR e in their co-domain (Fig.2-2), is defined as the ENERGY OF THIS SET, MEASURED IN DIRECTION e [106]:

$$\forall e \in S_U^M : \quad E_e \triangleq \sum_{n=1}^{N} (e^t . a_n)^2 \qquad (2\text{-}6a)$$

The sum of the squared orthogonal projections of these vectors, on any SUB-SPACE S in their range, is defined as the ENERGY OF THIS SET, MEASURED IN S:

$$\forall S \subset S_U^M : \quad E_S \triangleq \sum_{n=1}^{N} \left\| P_S(a_n) \right\|_2^2 \qquad (2\text{-}6b)$$

where $P_S(a_n)$ denotes the orthogonal projection of a_n into the subspace S.

THEOREM 2-1 (On the oriented energy measurement of a sequence of vectors):

Consider any set of M-vectors which form the columns of the MxN matrix A, with singular value decomposition $A = U \Sigma V$. Then:

$$1. \quad \sum_{n=1}^{N} \| a_n \|_2^2 = \sum_{p=1}^{P} \sigma_p^2 \qquad (2\text{-}7a)$$

$$2. \quad E_{u_m} \triangleq \sum_{n=1}^{N} (u_m^t . a_n)^2 = \sigma_m^2 \quad , m = 1 \ldots M \qquad (2\text{-}7b)$$

$$3. \quad \forall e \in S_U^M : E_e \triangleq \sum_{n=1}^{N} (e^t . a_n)^2 = \sum_{m=1}^{M} \mu_m^2 \sigma_m^2 \quad , \text{where } e = \sum_{m=1}^{M} \mu_m u_m \qquad (2\text{-}7c)$$

$$4. \quad \underset{S^m \subset S_U^M}{\text{Max}} \sum_{n=1}^{N} \| P_{S^m}(a_n) \|_2^2 = \sum_{j=1}^{m} \sigma_j^2 \quad , m = 1 \ldots M \qquad (2\text{-}7d)$$

$$5. \quad \underset{S^m \subset S_U^M}{\text{Min}} \sum_{n=1}^{N} \| P_{S^m}(a_n) \|_2^2 = \sum_{j=M-m+1}^{M} \sigma_j^2 \quad , m = 1 \ldots M \qquad (2\text{-}7e)$$

where: - P = Min(M,N)

- "Max" and "Min" denote operators, maximizing or minimizing over all m-dimensional subspaces S^m of the ambient range space S_U^M.

- $\sigma_m = 0$ when $m > P$

Instead of proving explicitely the above theorem -which is no more than a simple exercise on invariance of Euclidean norms under orthonormal transformations-, we rather comment in some more detail on its conceptual interpretation, as we will continuously use this interpretation in the subsequent chapters.

In colloquial terms: The TOTAL ENERGY in a vector sequence is equal to the energy in its singular spectrum (2-7a). This total energy is additively decomposed over M orthogonal left singular directions in the ambient M-dimensional range space (2-7b). If M>P, then M-P of these directions contain no energy at all. Equation (2-7c) shows that the energy, MEASURED IN AN ARBITRARY DIRECTION "e" can also be decomposed as a sum of "orthogonal" energies associated to the singular directions.

In terms of SVD, E_e is the sum of squared projections of the vectors $\sigma_p.u_p$ on the direction e (2-7c). For the example (1-1) we have radially plotted the oriented column energy along all unit vectors e in the ambient range space, and the oriented row energy along all unit vectors f in the in the ambient domain (Fig.2-3)

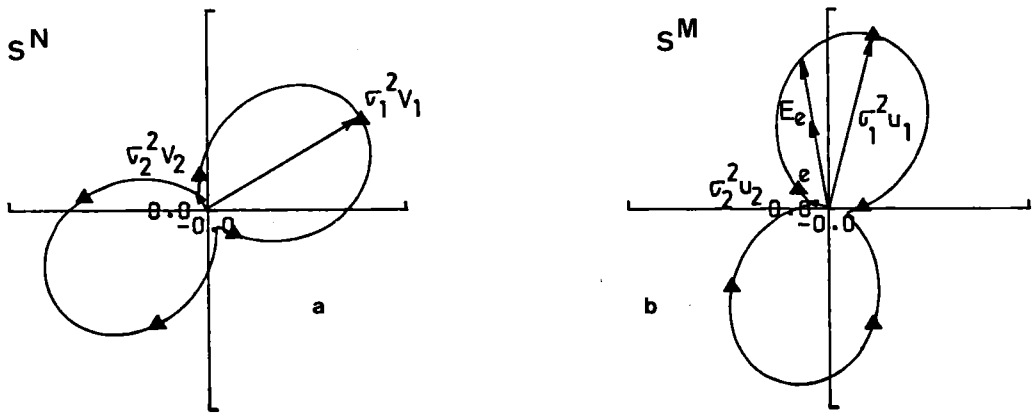


Fig. 2-3: Oriented row and column energy distribution for matrix A in ex(1-1).

Extrema in this "oriented energy distribution" occur at each singular direction. The first singular direction shows a maximum; the last singular direction (and possibly all the zero singular directions) show a minimum; the other singular directions (not present in the example) are saddle-points. Notice the typical narrowing along orientations close to smaller singular values, even for a mild ratio $\sigma_1 / \sigma_2 = 3$.

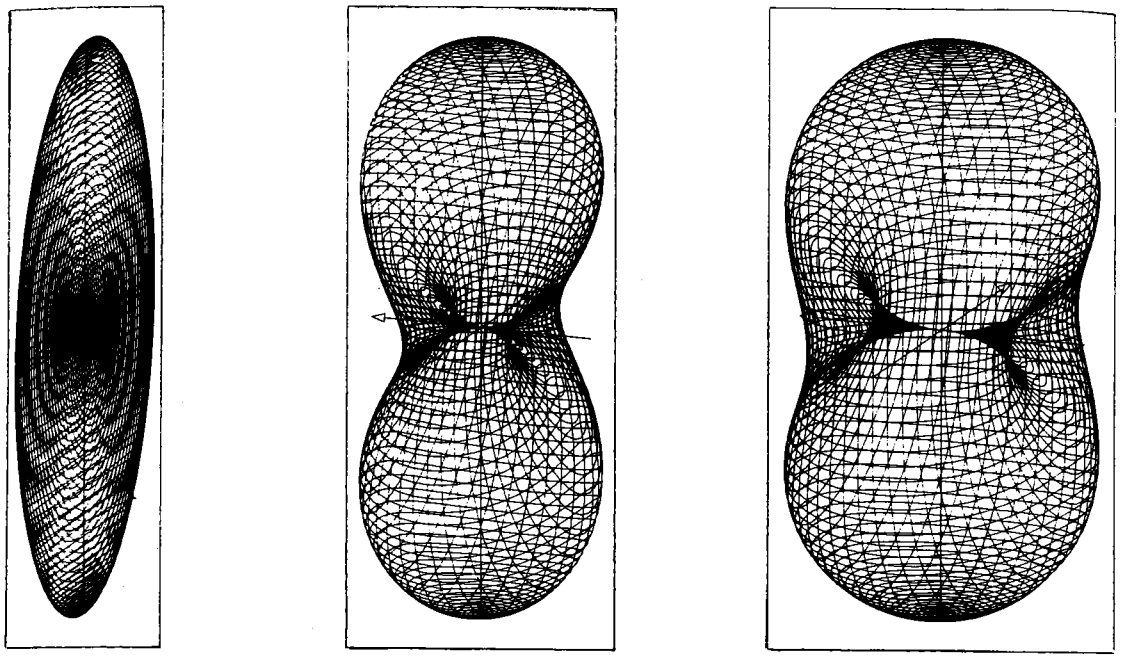


Fig. 2-4: The mapping ellipsoid, the oriented energy distribution and the distribution of its square root for a nearly singular 3x3 matrix.

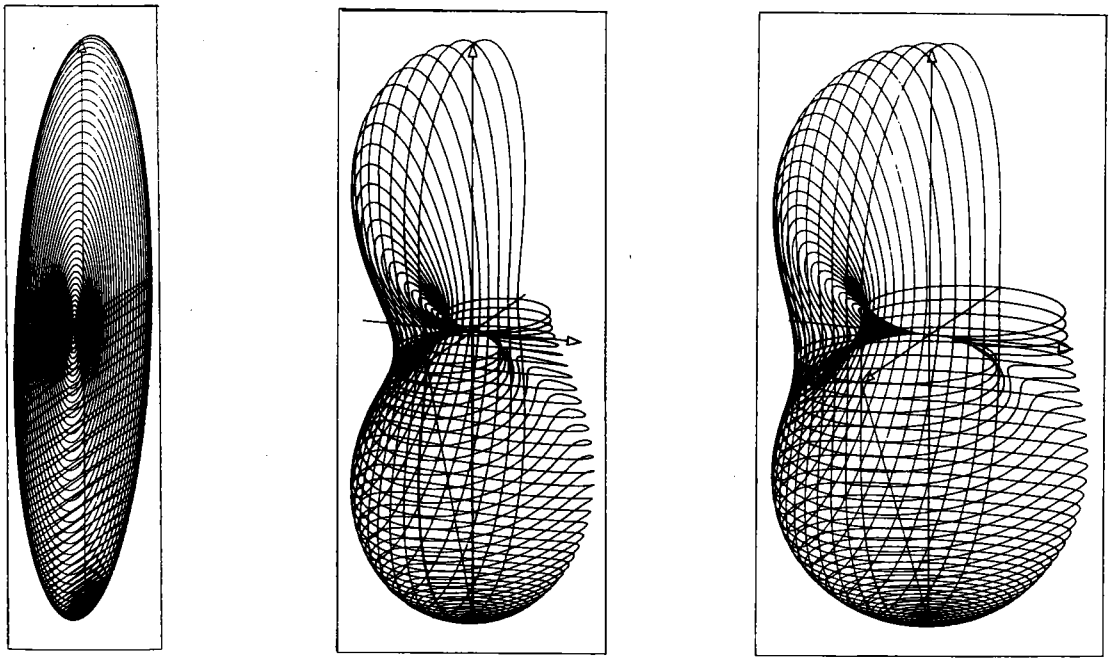


Fig. 2-5: The same distributions, of which a quarter has been cut out for representational purposes.

We want to stress here, that although the oriented energy distribution is everywhere differentiable (which implies that it is FLAT in a SUFFICIENTLY SMALL neighbourhood of the singular vector: a nice ALGEBRAIC property), it will grow very rapidly in a somewhat more distant neighbourhood. Moreover, reliable computations require to work directly with the square root of the energy (see next REMARK). In that case, the minima are sharp (and not differentiable). Practically this implies that if error blow-up is not kept within tight limits, minor inaccuracies on the data may lead to sizeable errors on the energy. Hence, accurate computations of the energy require very reliable computational methods, and this is precisely what Golubs algorithm (chapter 3) guarantees.

The equations (2-7d) and (2-7e) show that the orthogonal decomposition of the energy is canonical in the sense that it allows to find subspaces of dimension m where the projected sequence has minimal and maximal energy. This decomposition of the total space, as a direct sum of a space of maximal and minimal energy for a given vector sequence, leads to very interesting rank considerations based on an oriented signal to noise ratio (see further in this chapter), and which will prove to be extremely useful for realization purposes (chapter 9).

VERY IMPORTANT REMARK: Although, for ease of treatment, all the above has been stated in terms of (conceptually significant) ENERGIES -which implies dangerous summing of squares!-, the strength of the approach is that all computations can be performed WITHOUT EXPLICITELY USING THESE SQUARES. The singular value decomposition obtains the decomposition of the vector sequence as a sum of orthogonal dyadic terms, weighed with singular values that can be computed within full machine precision. A first illustration of the importance of this fact, that will be present throughout this thesis, will already appear from the next section.

Finally, to sharpen the geometric intuition on the above interpretations, Figs.2-4a...2-4c show the mapping-ellipsoid, the oriented energy distribution and the distribution of the square root of this energy for a 3-dimensional nearly singular ("strong" rank 2) matrix. Notice that the mapping ellipsoid is "thin" nearly everywhere, whereas the energy distribution is "fat" everywhere, except around a very narrow neighbourhood of the third singular vector. As we will see in chapter 3, rank determination is mainly a question of oriented energy measurement. Fig.2-4c shows in a suggestive way why rank determination is such a difficult numerical problem. In figs. 2-5a...2-5c a quarter of the surface has been cut out to show the typical (sharp!) narrowing in some more detail.

C. STOCHASTIC PROPERTIES OF COLUMNS OR ROWS IN A MATRIX.

It is interesting to establish the natural link that exists between the SVD of a matrix, and the stochastic properties of the sequence of vectors represented by its columns. To see this link, it is sufficient to analyse the meaning of the following expression for a sequence of N M-vectors a:

$$N.R_o \overset{\Delta}{=} N.\Omega(a\,.a^t) \underset{\text{ergodicity}}{\cong} \sum_{n=1}^{N} (a_n.a_n^t) = A.A^t = U\Sigma\Sigma^t U^t \qquad (2\text{-}8)$$

where $\Omega(.)$ stands for statistical expectation . Of course, strictly speaking, the above approximation is significant only for large N, and under the assumption of ergodicity. But in many circumstances, the relationship between the second-order joint moment matrix R_o of a vector stochastic process, and the finite summation (2-8) of external vector products of the associated sample sequence, is taken to be sufficiently satisfactory. Then from eq.(2-8), the following remarkable facts appear:

- Whatever the length N of the sample sequence may be, it can always be replaced by a sample sequence of $r \leqslant P=Min(M,N)$ vectors

$$U.\Sigma = [\, \sigma_1 u_1 \ \cdots \ \sigma_r u_r \,] \quad , \quad r = \text{rank}(A) \qquad (2\text{-}9)$$

  with identical second-order joint moment matrix.

- The number of entries, needed to store $U\Sigma$ is equal to that necessary for R_o, but for an equal precision on these data, the information stored in $U\Sigma$ is of superior quality compared to that in R_o. Indeed, U is associated directly to information in A, whereas R_o is proportional to the dangerously squared AA^t. This well known problem [8, 14] will be treated in chapter 3. Meanwhile, as an illustration consider the following sequence of three 2-vectors, put together in matrix A:

$$A = \begin{bmatrix} 1 & \mu & 0 \\ 1 & 0 & \mu \end{bmatrix} \qquad (2\text{-}10a)$$

  Now with absolute accuracy level ε , the product

$$U\Sigma \triangleq \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} \cdot \begin{bmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2}\,.\,\mu \end{bmatrix} \qquad (2\text{-}10b)$$

  can be computed and stored with virtually the same absolute accuracy. But in

$$R_o = \frac{1}{3} \begin{bmatrix} 1+\mu^2 & 1 \\ 1 & 1+\mu^2 \end{bmatrix} \qquad (2\text{-}11)$$

  the quantity $1+\mu^2$ can only be stored with absolute accuracy ε . This means that effects μ , smaller than $\sqrt{\varepsilon}$ will completely disappear in eq.(2-11), which is a severe loss of accuracy, compared to eq.(2-10a).

- The principal components of the correlation (the eigenvectors and eigenvalues of R_o) are directly available as the left singular vectors and the squared singular values of $A/\sqrt{N}$.

The point we want to stress now, is that from (2-8) it is clear that the covariance matrix in fact simply characterises the oriented energy distribution of a vector sequence. But if the original vector signal is available, it does this in a numerically ill way. The SVD of the available vector sequence (in fact $U\Sigma$ is sufficient) is a simple, efficient and by far more reliable way to perform and analyse this same measurement.

Though stochastic processes are not the main subject of this thesis, we nevertheless want to emphasize the importance of the above statement. The reason is that in numerical analysis this fact seems so well known, that algorithms to compute $U\Sigma$ from large, sequentially available data are described in basic textbooks [14] and are there mentioned as standard-to-use software. But in typical stochastic process environment, many theoretical derivations and implemented algorithms are still unnecessarily based on the covariance idea. Let us mention two examples:

- $U\Sigma$ is proportional to the orthogonal square root of R_o. Such a square root is often used as a "whitening transformation" in so called Minimum Variance, or Markov type estimations [38]. There is no reason -but it is frequently encountered in the litterature!-, to compute this square root from R_o, if R_o has to be obtained by an AA^t product from available measurements of a sample sequence of the analysed stochastic process.

- During the last few years, quite some effort has been spent on the Levinson algorithm [34]. This algorithm aims at obtaining the decomposition of a compound covariance matrix R:

$$R \triangleq \begin{bmatrix} R_0 & R_1 & \cdots & R_N \\ R_{-1} & R_0 & \cdots & R_{N-1} \\ \cdot\cdot & \cdot\cdot & \cdots & \cdot\cdot \\ R_{-N} & \cdot\cdot & \cdots & R_0 \end{bmatrix} \qquad (2\text{-}12)$$

in an efficient and reliable way. In analogy with (2-8), R is easily seen to be the variance matrix (i.e. the second-order joint moment matrix for zero-mean signals) of an augmented vector signal

$$\alpha \triangleq \begin{bmatrix} a_k \\ \cdots \\ a_{k-N} \end{bmatrix} \qquad (2\text{-}13)$$

directly related to the original vector signal. If eq.(2-12) is the result of COMPUTATIONS ON AVAILABLE DATA (2-13), then any effort to set up reliable algorithms for the decomposition of R is useless, because THE MERE FACT OF COMPUTING AND USING R RATHER THAN A OR $U\Sigma$ IS ALREADY A NUMERICAL DISASTER.

As said before, this thesis is not primarily concerned with stochastic problems. But the concept of oriented energy distribution seems to be so naturally suitable and intuitively powerful for use in statistical applications, that the author could not resist to briefly mention its significance in a stochastic setting. Moreover, some elementary stochastic SVD aspects are used for the analysis of oriented signal to noise ratios in the next section.

D. SIGNAL-TO-SIGNAL AND SIGNAL-TO-NOISE RATIOS.

Consider two vector signals $[a_n]$ and $[b_n]$, n=1 ... n=N. (2-14)

In a first attempt to compare the oriented energy distributions of [a] and [b], one might think of defining an "oriented signal-to-signal ratio" as the ratio of their oriented energies, measured in a certain direction e, or in a certain subspace S. The inconvenience of this definition is that the oriented energies are measured, based on orthogonal projections. This implies that [a] and [b] are measured in a certain metric, associated to the chosen basis for the description of the vector sequences. Though this is not inconvenient for energy measurements (a user of the energy-concept is responsible for the choice of the metric in which this energy is measured), it becomes quite embarrassing when it appears that the ratio of the energies associated to two vector sequences, but measured in one and the same direction, is dependent on the particular metric used. This unsatisfactory aspect can be explained, and a more significant definition can be introduced, based on the following properties:

Lemma 2-1:

Let $E_e[a]$ and $E_e[b]$ be oriented energies of vector sequences [a] and [b]

Apply a nonsingular linear transformation T on both [a] and [b].

$$\text{Then} \quad \forall T : \quad \frac{E_e[a]}{E_e[b]} = \frac{E_{(T^{-1})^t e}[Ta]}{E_{(T^{-1})^t e}[Tb]} \qquad (2\text{-}15)$$

□

The proof is a simple exercise based on the definitions. The message of lemma 2-1 is the following. Although the measurement of oriented energy-ratios IN A CERTAIN DIRECTION is dependent on the chosen metric (the chosen basis), the existence of A RATIO WITH A certain VALUE is INDEPENDENT from this metric. Stated otherwise: if a certain signal to signal ratio exists for some direction in a particular metric, it will appear in all metrics, linearly related to the original one. And thus in particular, if properly stated, the minimum of these ratios will be invariant to the chosen metric. We therefore define the MINIMAL SIGNAL TO SIGNAL RATIO (MSSR) R[a,b,S] of two vector sequences [a] and [b], measured in a subspace S as:

$$R[a,b,S] \triangleq \min_{e \in S} \frac{E_e[a]}{E_e[b]} \qquad (2\text{-}16)$$

Applying lemma (2-1) to this definition, we state the following:

Lemma 2-2:

$$\forall T,S : \quad R[a,b,S] = R[Ta,Tb,(T^{-1})^t S] \qquad (2\text{-}17)$$

□

In words: if for some metric, the minimal signal to signal ratio measured in a subspace S is R, then in all other metrics, there exists a subspace in which the measured signal to signal ratio is R.

Theorem 2-2:

The maximal MSSR, that can be obtained over all subspaces $S^m \subset S^M_U$, of a given dimension m, is invariant for nonsingular linear transformations T

$$\forall T: \quad \underset{S^m \subset S^M}{\text{Max}} \; R[a,b,S^m] = \underset{S^m \subset S^M}{\text{Max}} \; R[Ta,Tb,S^m] \qquad (2\text{-}18)$$

□

Hence no improvement in MSSR can be expected from a change of basis in S^M. On the other hand, there is no objection to apply such a transformation if a subspace of maximal MSSR is easier to compute in the transformed space.

Notice, that in the fundamental definition, the signals are NOT projected in the subspace S : the subspace S restricts the orientations, in which the measurements are allowed. For signals, projected in S, we have:

Theorem 2-3:

The MSSR in S^m of two sequences [a] and [b] is the same as the MSSR of their projections in S^m.

$$\forall S^m: \quad R[\,a,b,S^m\,] = R[\,P_{S^m}(a),P_{S^m}(b),S^m\,] \qquad (2\text{-}19)$$

□

Algorithm 2-1 (on computing m-dimensional approximations of maximal MSSR):

Consider two sequences [a] and [b] in an M-dimensional space S^M: compute the m-dimensional subspace S^m_a of maximal MSSR for these two sequences. Also compute two sequences $[a_2]$ and $[b_2]$ in S^m_a with the same MSSR as above and as close as possible (in two-norm sence) to the original sequences. The computation uses the SVD of the involved sequences:

$$A = U_a \Sigma_a V_a \qquad \text{and} \qquad B = U_b \Sigma_b V_b \qquad (2\text{-}20)$$

Step 1. Make the oriented energy measurement of [b] isotropical, by a change of basis with

$$T = \Sigma_b^+ U_b^t \qquad (2\text{-}21a)$$

In the new basis, $[a_1]$ and $[b_1]$ follow as:

$$A_1 = \Sigma_b^+ U_b^t A \quad , \quad \text{and} \quad B_1 = \Sigma_b^+ U_b^t B \qquad (2\text{-}21b)$$

Step 2: In the new basis, the oriented signal to signal ratio between [a] and [b] is equal to the oriented energy of [a]. From (2-7c), the m-dimensional subspace with maximal MSSR is then simply the m-th principal singular space of A_1. The associated MSSR is σ_m^2, and from (2-19), the m-dimensional closest approximate sequences are simply the projections of $[a_1]$ and $[b_1]$ in this principal singular subspace:

$$\text{From } A_1 = U_{A_1} \Sigma_{A_1} V_{A_1} \qquad (2\text{-}22a)$$

$$S^m = \text{Span}_{col}[\, u_1 \ldots u_m \,] \qquad (2\text{-}22b)$$

$$R[a,b,S^m] = \sigma_m^2 \qquad (2\text{-}22c)$$

□

The solution of this problem is the key to our reliable realization algorithm (chapter 8). A very interesting feature in this algorithm is, that from the SVD of A , the approximation error and the associated obtained MSSR are directly displayed, which makes a motivated choice of m possible ([107] & chapter 8).

A second important fact is, that the algorithm allows for a perfectly reliable implementation. Again all computations use SVD's and the associated orthogonal projections, and NEVER any A^tA-type squaring.

Finally, from section C it follows that vector stochastic processes are characterized by an equivalent vector signal $U\Sigma$, closely related to the covariance matrix. Hence, there is no substantial difference in speaking about signal to signal (MSSR), or signal to noise (MSNR) ratio, in which case the second vector sequence is not known by its actual time sequence, but as an equivalent sequence with identical stochastic properties. Moreover, if the reference noise is isotropic, the first step can be simply replaced by a trivial scaling with $1/E_n$, (the expected energy in any direction e, i.e. the standard deviation of the noise on one entry of the vector process). The rank-m approximation of highest minimal signal to noise ration can then further be obtained from one SVD [107].

For all the preceding derivations, the reader can easily generate dual properties for vectors put together as rows in a matrix.

E. SOME MORE PRACTICAL CONCEPTS, DIRECTLY RELATED TO SVD.

The analysis of the operator properties of a matrix A: $x \longrightarrow y=Ax$, and the analysis of the oriented energy distribution $U\Sigma$ of a column vector sequence, are very powerful conceptual tools in the analysis of (generalized) inversion, conditionning and rank of a set of linear equations, as will be illustrated now.

For all the subsequent applications, we partition the SVD as follows:

$$A = \underbrace{\begin{bmatrix} U_1 & U_2 \end{bmatrix}}_{R \quad M-R} \cdot \begin{matrix} \overbrace{\begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}}^{R \quad N-R} \end{matrix} \begin{matrix} R \\ M-R \end{matrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} \begin{matrix} R \\ N-R \end{matrix} = U_1 \cdot \Sigma_1 \cdot V_1 \qquad (2\text{-}23)$$

where $R \leq P = \text{Min}(M,N)$, and the σ_p, $p=R+1 \ldots p=P$ are assumed numerically zero, i.e. zero within the accuracy of the given data, or within the precision of the computations. From (2-23) one can directly obtain very pertinent information on A:

a. The four fundamental subspaces of a linear operator (Fig 2-6).

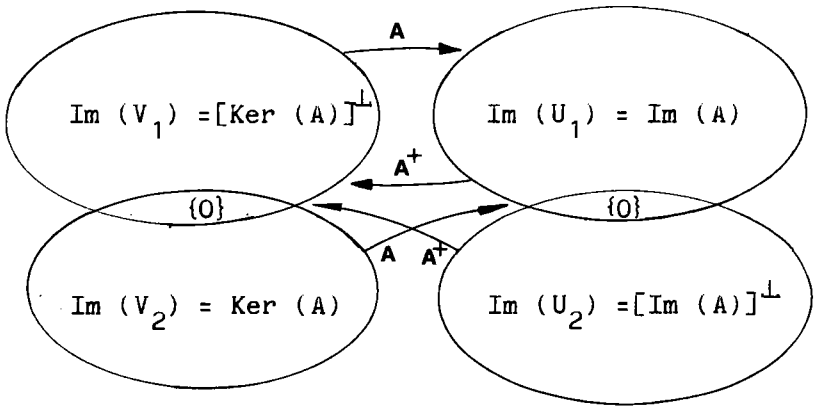


Fig. 2-6: The four fundamental subspaces of a matrix A.

The rows of V_2 form an orthonormal basis for the kernel of A, i.e.:

$$\text{Ker}(A) \triangleq \{ x : Ax = 0 , x \in S^N \} = \text{Span}_{\text{row}} [V_2] \qquad (2\text{-}24a)$$

The columns of U_1 form an orthonormal basis for the range of A i.e:

$$\text{Im}(A) \triangleq \{ y : y = Ax , x \in S^N \} = \text{Span}_{\text{col}} [U_1] \qquad (2\text{-}24b)$$

The columns of V_1 and U_2 span the complements of the above subspaces.

b.Genealized inverse.

We spend some time on this subject, because it is an always present key concept in reliable modelling and identification problems, and because there is a lot of confusion when it comes to its correct interpretation in practical applications. The idea is as follows [7]: restrict the operator A naturally as a mapping from the orthogonal complement of the kernel of A into the range of A. This operator is one-to-one and invertible:

$$x \longrightarrow y=Ax \quad : \quad U_1 \Sigma_1 V_1 x = y \qquad (2\text{-}25a)$$

$$A^+ y = x \longleftarrow y \quad : \quad x = V_1^t \Sigma_1^{-1} U_1^t y \qquad (2\text{-}25b)$$

which defines the generalized inverse in a very simple and natural way. Notice in this context the surprising property of commutativity of summation and generalized inversion for the dyadic decomposition (the DYADIC INVERSION PROPERTY):

$$A^+ \triangleq \left[\sum_{i=1}^{R} \sigma_i u_i v_i \right]^+ = \sum_{i=1}^{R} \left[\sigma_i u_i v_i \right]^+ \qquad (2\text{-}25c)$$

c.Norms of matrices:

The classical 2-norm for both A and its generalized inverse can be obtained directly from an SVD, as can be derived from Fig.2-1:

$$\|A\|_2 \triangleq \max_x \frac{\|Ax\|_2}{\|x\|_2} = \max_{y \in Im(A)} \frac{\|y\|_2}{\|A^+ y\|_2} = \sigma_1 \qquad (2\text{-}26a)$$

$$\|A^+\|_2 \triangleq \max_y \frac{\|A^+ y\|_2}{\|y\|_2} = \max_{x \in ker(A)^\perp} \frac{\|x\|_2}{\|Ax\|_2} = \sigma_R^{-1} \qquad (2\text{-}26b)$$

R = rank(A) and P = Min(M,N). The Frobeniusnorm is as easily obtained as:

$$\|A\|_F \triangleq \sum_{n=1}^{m=M} \sum_{m=1}^{n=N} a_{mn}^2 = \sum_{p=1}^{P} \sigma_p^2 = \text{trace } A^t A \qquad (2\text{-}27)$$

Clearly 2-norms are significant in contexts, where the matrix is seen as a linear operator, whereas the Frobeniusnorm is more meaningful for the energetic analysis of matrices, seen as sequences of vectors.

d. Numerical rank.

The rank of a matrix is known to be a delicate numerical problem. Historically, rank has been defined on the basis of linear dependence of rows or columns, zeroness of eigenvalues, zeroness of (sub-)determinants etc... But the extension of these concepts to the definition of a NUMERICALLY RELIABLE RANK has proved to be highly confusing and misleading. Let us illustrate this with a few examples.

Example 2-1 [54]: consider the very simple, uppertriangular matrix

$$A = \begin{bmatrix} 1 & 10000 \\ 0 & 1 \end{bmatrix} \qquad (2\text{-}28)$$

A is uppertriangular with nonzero diagonal elements, its determinant is 1, both eigenvalues are 1. So algebraically, this matrix is nonsingular. Yet a minor disturbance

$$A_d = \begin{bmatrix} 1 & 10000 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0.0001 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 10000 \\ 0.0001 & 1 \end{bmatrix} \qquad (2\text{-}29)$$

makes the determinant strictly equal to zero, and shifts the eigenvalues to 0 and 2. So numerically, this matrix is highly singular, and clearly neither the eigenvalues nor the determinant are reliable measures for singularity. Notice that one could here suspect singularity by comparing the order of magnitude of the entries in the matrix, the norms of the columns, or the angles between them. But methods, based on these concepts fail in other examples:

Example 2-2 [14]: For large N, consider the following

$$A = \begin{bmatrix} -1 & 1 & 1 & \cdots & 1 \\ 0 & -1 & 1 & \cdots & 1 \\ 0 & 0 & -1 & \cdots & 1 \\ \cdot & \cdot & \cdot & \cdots & \cdots \\ 0 & \cdots & & \cdots & 0 \quad -1 \end{bmatrix}_{NxN} \qquad (2\text{-}30)$$

All nonzero entries in A have a same (unit) value. Again the eigenvalues and the determinant are unit. And yet,

$$A = \begin{bmatrix} -1+\varepsilon & 1 & 1 & \cdots & 1 \\ 0+\varepsilon & -1 & 1 & \cdots & 1 \\ 0+\varepsilon & 0 & -1 & \cdots & 1 \\ \cdots & \cdot & \cdot & \cdots & \cdots \\ 0+\varepsilon & & \cdots & 0 & -1 \end{bmatrix}_{NxN} \qquad (2\text{-}31)$$

is singular for ε as small as $\varepsilon = 2^{-(N-1)}$.(Other, even somewhat smaller disturbances can make A singular too [14], but this one is particularly simple). Clearly, elements all of a same magnitude do not guarantee nice rank behaviour. Finally, consider:

Example 2-3 [55],[115]:

$$A = \frac{1}{\sqrt{N.(N-1)}} \begin{bmatrix} N-1 & -1 & \cdots & -1 & -1 \\ -1 & N-1 & \cdots & -1 & -1 \\ \cdot\cdot & \cdot\cdot & \cdots & \cdot\cdot & \cdot\cdot \\ -1 & -1 & \cdots & N-1 & -1 \\ -1 & -1 & \cdots & -1 & N-1 \end{bmatrix}_{NxN} \qquad (2\text{-}32)$$

A classical analysis of this matrix is really confusing:

First: A has columns, which are all of unit length, and which are ALL TWO BY TWO NEARLY ORTHOGONAL: the cosine between any two columns is

$$\cos\,(a_i,a_j) = \frac{1}{N-1} \qquad , \qquad i \neq j \tag{2-33}$$

which tends to zero for large N. Yet A is PERFECTLY SINGULAR. (The sum of all columns is the zero vector).

Second: consider the Nx(N-1) submatrix of A, obtained by dropping the last column. THE ANGLE BETWEEN ANY COLUMN AND THE HYPERPLANE SPANNING THE REMAINING N-2 COLUMNS is given by:

$$\cos\,[a_i, \mathrm{Span}_{col}(a_1, \ldots a_{i-1}, a_{i+1}, \ldots a_{n-1})] = -\sqrt{\frac{N-2}{2.(N-1)}} \tag{2-34}$$

So one might conclude [55] that such a submatrix is of full rank. This also is NOT correct. Indeed, with a minor change

$$dA = N^{-1/2}.(N-1)^{-3/2} \begin{bmatrix} -1 & -1 & \ldots & -1 \\ -1 & -1 & \ldots & -1 \\ \ldots & \ldots & \ldots & \ldots \\ -1 & -1 & \ldots & -1 \\ N-1 & N-1 & \ldots & N-1 \end{bmatrix}_{N,N-1} \tag{2-35a}$$

this matrix becomes strictly singular. Notice that for large N

$$\frac{\|dA\|_F}{\|A\|_F} = \frac{1}{N-1} \tag{2-35b}$$

tends to zero. Thus the approach, by which one measures singularity as the distance necessary to bring ONE COLUMN IN THE SPACE GENERATED BY THE OTHERS (residual value analysis [55]), is NOT reliable. One may need a lot less energy to bend ALL vectors into a (new) hyperplane of lower dimension [115]. We will again use this idea for our "global LLS approach to the solution of a set of linear equations" in chapter 4.

Third: based on the fact, that there exists no Nx(N-1) submatrix of numerically full rank, one could conclude that the rank of the original matrix A is at most (N-2). This AGAIN is a wrong conclusion. Indeed, no disturbance of norm less then 1 (the norm of each of the columns in A, independently of N), is capable to transform A into a singular matrix.

As one can see, this example is quite extravagant. Its properties, and the above rank study are described in some more detail in [115]. But it should clearly emerge from the above short treatment, that if the definition of rank is not chosen appropriately, one may run into severe numerical trouble whenever modelling procedures are based in a critical way on ranks of

matrices. This conclusion is well known in numerical analysis, in which reliable rank detection is always based on Singular Value Decomposition. Notice indeed, that in example (2-1), the SV's are approximately 10000 and 0.0001; in example (2-2) all SV's have about the same magnitude except for the smallest one, that is smaller than, but approximately 2^{2-N} [14]. In example 2-3 the singular spectra are as follows:

For the NxN matrix

$$\sigma_n = \sqrt{N/(N-1)}\ ,\ n = 1 \ldots N-1 \quad \text{and} \quad \sigma_N = 0 \qquad (2\text{-}37)$$

whereas for for the NxN-1 submatrix [55]:

$$\sigma_n = \sqrt{N/(N-1)}\ ,\ n = 1 \ldots N-2 \quad \text{and} \quad \sigma_{N-1} = \sqrt{1/(N-1)} \qquad (2\text{-}38)$$

So clearly, in each of the above special cases, the singular spectrum apparently detects the correct singularity. That this is general is one of the great discoveries in numerical analysis [4,5,7,10,11,14,21 and many others]. We will now introduce this SVD approach in a natural and conceptually suggestive way, using the energy properties of SVD, discussed in section B. Indeed, the oriented energy Ee measured in direction e, is precisely the sum of squares of the corrections to be added to the given matrix, in order to measure no energy at all in direction e. In the singular directions, this energy (the square of the associated singular value), is extremal, and the terms of the dyadic decomposition are precisely the corrections to be made in each of the singular directions. We formalize these facts in a theorem:

Theorem 2-5 (on oriented rank reduction):

For a vector e of unit length:

$$\text{a. } E_e\,[\,A - e.e^t A\,] = 0 \quad \text{and} \quad \left\| e.e^t A \right\|_F = E_e[A] \qquad (2\text{-}38)$$

$$\text{b. } \left\| A - e.a^t \right\|_F = \min_x \left\| A - e.x^t \right\|_F \quad \text{iff} \quad E_e[\,A - e.a^t] = 0 \qquad (2\text{-}39)$$

$$\text{c. } \text{Rank}\,(A - e.e^t A) = \text{Rank}(A) - 1 \quad \text{iff} \quad e \in \text{Span}_{col}(A) \qquad (2\text{-}40)$$

$$\text{d. } \min_{e \in \text{Span}_{col}(A)} [\ \left\| e.e^t A \right\|_F\ ] = \left\| u_R^t.u_R\, A \right\|_F = \sigma_R^2 \qquad (2\text{-}41)$$

□

The meaning of the above expressions is the following:

a. Eq.(2-38): The rank-one correction dA, that makes A energyless in the direction of a unit vector e, is $e.e^t.A$, the energy of which is $E_e[A]$.

b. Eq.(2-39): The matrix of lowest energy, obtainable with corrections on the columns of A, parallel to a direction e, is a matrix of zero energy in the direction e. In colloquial terms: the e-oriented energy is the maximum that can be extracted from matrix A with e-oriented corrections on the columns, and eq.(2-38) gives an explicit expression for these corrections.

c. Eq.(2-40): If e is in the column span of A, then annihilation of e-oriented energy and rank reduction are equivalent phenomena (and conversely). This is the key to the conceptual understanding of reliable (energy-based) rank reduction methods and their criticalness. The oriented energy distribution of a disturbed matrix describes the total (original + disturbation) oriented energy of this matrix. If the undisturbed column vector sequence is orthogonal to some direction, its oriented energy is EXACTLY zero in that direction. But then additive disturbances can not enlarge this energy more than with their own energy level. In Figs.(2-4b) and (2-4c) one sees such low-energy orientations. But one also sees that these orientation are very critical: small errors may cause disastrously wrong conclusions.

d. Eq.(2-41): An overall closest approximation of a rank R matrix by a matrix of rank R-1 is obtained by a rank-one correction with the dyadic term associated to the smallest singular value. This is nothing but the Eckart and Young approximation [35], which appears here to be a simple oriented energy-extraction, in the direction of minimal energy of the matrix. The oriented energy approach, picturally represented in the Figs. 2-4b/c and 2-5b/c gives a very suggestive interpretation of this operation.

With the above interpretation, theorem (2-5) is self-explanatory. We will therefore only give an outline of the formal proof.

- Eq.(2-38) is trivial using $\left\|e.e^t A\right\|_F^2 = \left\| e^t A\right\|_F^2 = E_e[A]$

- Eq.(2-39) follows from the "orthogonal" additive decomposition:

$$A + e.x^t = A_1 + A_2 = e.(e^t A + x^t) + (I - e.e^t).A \qquad (2\text{-}42)$$

for the terms of which the rule of Pythagoras is valid:

$$\left\|A_1 + A_2\right\|_F = \left\|A_1\right\|_F + \left\|A_2\right\|_F \qquad (2\text{-}43)$$

and hence $A + e.x^t$ will have minimal norm when $A_1 = 0$, i.e. when $x = -e^t A$.

- Both directions in eq.(2-40) easily follow from the following facts: $e \in \mathrm{Im}(A+ee^tA)$; linear combinations of columns of A and $A-ee^tA$ will differ only in a vector proportional to e, hence there is a rank difference only if $e \in \mathrm{Im}(A)$; If $e \in \mathrm{Im}(A)$ then $\mathrm{Im}(A+ee^tA) \subset \mathrm{Im}(A)$ and hence $\mathrm{rank}(A-ee^tA) \leqslant \mathrm{rank}(A)$; finally, ee^tA is a rank-one correction, and so it can change the rank of A maximally with one unit.
- Finally, Eq.(2-41) directly follows from expression (2-7c).

e. Condition numbers:

Essentially, condition numbers describe the sensitivity of a solution of linear sets of equations to possible inaccuracies in the available data. As this is the subject of a more detailed dicussion in chapter 3, we will only introduce some ideas, as an illustration of the conceptual power of the above sections. For this, consider the following two problems:

PROBLEM 1. Assume an exact matrix A. For this GIVEN MATRIX A, how much is the maximal increase in relative inaccuracy, that may occur when solving x from Ax=y, for the worst position of the RHS y, and the unluckiest position of the error dy.

This is a question to which the classical condition number gives an answer. In 2-norm, the answer is trivially obtained from Fig. 2-1:

$$K(A) \triangleq \underset{\substack{dy,y \\ Ax=y \\ Adx=dy}}{\text{Max}} \frac{(\|dx\|_2 / \|x\|_2)}{(\|dy\|_2 / \|y\|_2)} = \frac{\sigma_1}{\sigma_R} = \|A\|_2 \cdot \|A^+\|_2 \qquad (2\text{-}44)$$

PROBLEM 2. Again assume an exact matrix A. Now for a given SET OF LINEAR EQUATIONS Ax=y, how much is the maximal increase in relative inaccuracy, that may occur when solving x from Ax=y.

Notice, that in this PROBLEM 2, the RHS y is assumed fixed and known, and hence it can be used in the evaluation of the condition. Again, the SVD representation of Fig 2-1. allows for a direct answer of this question:

$$K_y(A) \triangleq \underset{\substack{dy \\ Ax=y \\ Adx=dy}}{\text{Max}} \frac{(\|dx\|_2 / \|x\|_2)}{(\|dy\|_2 / \|y\|_2)} = \frac{\|y\|_2}{\sigma_1 \cdot \|x\|_2} K(A) \qquad (2\text{-}45)$$

Similar expressions exist for sensitivity to inaccuracies in the entries of matrix A, and for global sensitivity of a set of linear equations (see chapter 3). What we want to emphasize here is mainly the strength of full conceptual understanding of SVD as presented in sections A and B. Indeed, from this interpretation, eq. 2-44 gives an upperbound for the increase in relative inaccuracy of the solution x, for the unluckiest orientation of dy ($dy // u_r$), and the worst orientation of y ($y // u_1$). So this upperbound

does NOT use information about the actual orientation of y, and hence gives information about the matrix A, rather than about the set of associated equations. In eq.2-45, the actual position of y is taken into account, which may drastically reduce the estimation of the solution sensitivity if y is nearly orthogonal to strong singular directions.

This geometric view on the problem is more attractive than a purely algebraic upperbound: it indicates whether an ill condition will GENERICALLY lead to highly degenerated results, or rather exceptionnally, and it gives an indication on the particular situations where difficulties occur. It is shown with an example in the next chapter that the generic condition strongly depends on the particular context, but that this context can be easily recognized from external and quite general information on the type of problem. This geometric approach has often been the guideline to our understanding and finally to the solution of the realization and identification problems treated in chapter 9-10. And although their solution is now more formalized for reasons of brevity and rigor, it still is the author's conviction that a real understanding, and even more the improvement of this algebra is not possible without geometrical interpretation of the type described in this chapter: a point of view, that we hope to motivate in the remainder of this thesis.

Chapter 3

SVD - PROPERTIES OF SENSITIVITY AND NUMERICAL STABILITY

For conceptual as well as for numerical reasons, it is often appreciated that the solution of a problem does not significantly change for insignificant variations on the given data. In this context, it is usual to make a clear distinction between the notions of sensitivity (conditioning) and of numerical stability.

A given PROBLEM STATEMENT is said to be WELL CONDITIONED when perturbations on the data lead to perturbations of a comparable level on the result. It is called ILL when the perturbation of the result can be much larger than the perturbation on the available data.

A given ALGORITHM is said to be NUMERICALLY STABLE when the results of this algorithm can always be kept within a bounded neighbourhood of the exact results, associated to the available data. These bounds depend on the computational precision and on the complexity of the problem.

Notice that we used the word RESULT rather than SOLUTION. We call the solution the set of data, required by the problem statement itself. Any construct, based on this set of data, and useful in the general problem context is called a result. For example:

- For a set of linear equations Ax=y, the solution is the set of entries in vector x, i.e. the representation of x in the chosen basis of the problem statement Ax=y. The result however may be the vector x, seen as a entity independent of the chosen basis, or the orientation only of vector x (forward results), or the product Ax seen as an approximation for the Right Hand Side y (backward result).

- For the SVD $A=U\Sigma V$ of a given matrix, the solution is the set of entries in the three factors U, Σ, and V. The result may be the singular spectrum Σ, the weighted left singular basis $U\Sigma$, the characterisation of the four fundamental spaces of A, the total set of singular triplets (forward results), or it may be the factored rank-r reconstruction $A_r = U_{M,r} \cdot \Sigma_{rr} \cdot V_{rN}$ of matrix A (backward result).

Notice also that we attribute conditioning to a PROBLEM STATEMENT, rather than to a PROBLEM. Indeed, apparently equivalent problem formulations for one and a same problem may show quite large variations in perturbation sensitivity ([110] and chapter 7). Moreover, for certain applications, it is possible to reformulate an originally ill problem statement into one that is well posed ([111] and chapter 8, [113]). This chapter however is restricted to the analysis of problem statements AS THEY ARE. We now state the main properties of sensitivity and/or numerical stability for the SVD and for sets of linear equations.

A. SENSITIVITY PROPERTIES OF THE SVD.

Theorem 3-1 (on the sensitivity of the singular spectrum):

$$\left| \sigma_p(A+dA) - \sigma_p(A) \right| \leq \left\| dA \right\|_2 \quad , \quad p=1 \ldots P \tag{3-1a}$$

$$\left\| d\Sigma \right\|_F \leq \left\| dA \right\|_F \tag{3-1b}$$

□

This classical theorem guarantees FORWARD insensitivity [Eq.(3-1a) for the individual singular values (bounded by the maximal oriented energy in dA ; eq.(3-1b) for the singular spectrum as a whole (bounded by the total energy in A)], and is based on eigenstructure properties of symmetrically extended matrices of the type (see [14], and [19],[56],[57]):

$$\tilde{\tilde{X}} \triangleq \begin{bmatrix} 0 & X \\ X^t & 0 \end{bmatrix} \tag{3-2}$$

Theorem 3-2 (on the sensitivity of the singular subspaces):

Consider the singular value decompositions of an original matrix A, and of a slight perturbation A+dA of this matrix. Partition both according to their p-th principal singular space:

$$A = U \Sigma V = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \cdot \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} \tag{3-3a}$$

$$A + dA = U^* \Sigma^* V^* = \begin{bmatrix} U_1^* & U_2^* \end{bmatrix} \cdot \begin{bmatrix} \Sigma_1^* & 0 \\ 0 & \Sigma_2^* \end{bmatrix} \cdot \begin{bmatrix} V_1^* \\ V_2^* \end{bmatrix} \tag{3-3b}$$

When the perturbation dA is small enough, i.e. when

$$0 \leq \alpha \triangleq \frac{\left\| U_2^{*t} . dA . V_1^{*t} \;\vdots\; V_2^* . dA^t . U_1^* \right\|_F}{\sigma_{p+1} - \sigma_p - \left\| U_1^{*t} . dA . V_1^{*t} \right\|_2 - \left\| U_2^{*t} . dA . V_2^{*t} \right\|_2} < \frac{1}{2} \tag{3-4}$$

then there exist matrices $P_{M-p,p}$ and $Q_{p,N-p}$, such that

$$\left\| P^t_{p,M-p} \;\vdots\; Q_{p,N-p} \right\|_F < 2\alpha \tag{3-5}$$

$$\text{Span}_{col} \left[U_1 + U_2 . P \right] = \text{Span}_{col} \left[U_1^* \right] \tag{3-6a}$$

$$\text{Span}_{row} \left[V_1 + Q . V_2 \right] = \text{Span}_{row} \left[V_1^* \right] \tag{3-6b}$$

□

Theorem 3-2 is a slightly modified version (restricted to principal singular subspaces, and for perturbations from A to A+dA) of theorem (6-4) in Stewarts remarkable paper [6]. It is a quite elaborated estimation of the perturbation bounds for the principal singular subspaces due to perturbations on the matrix entries. The interpretation is as follows:

- For dA sufficiently small compared to the separation between the involved singular values [eq.(3-4)], P and Q can be bounded [eq.(3-5)], and so can the perturbation of the considered principal singular subspace [eq.(3-6)]. Notice that the sensitivity grows when the separation between the concerned singular values narrows, which is equivalent to saying that the absolute orientation of a singular subbasis, associated to a cluster of close singular values, becomes vague within its span: an effect that is proportional to the closeness between the involved Singular Values. This is a numerically more meaningful equivalent for the algebraic nonuniqueness statement for singular bases (Theorem 1-2).

- The matrices P and Q are directly related to the canonical angles ϑ_i, separating the principal singular subspaces of the original and the perturbed matrix [6]:

$$\sigma_i(P) = \tan [\, \vartheta_i(S_U^i , S_{U*}^i) \,] \qquad (3\text{-}7a)$$

$$\sigma_i(Q) = \tan [\, \vartheta_i(S_V^i , S_{V*}^i) \,] \qquad (3\text{-}7b)$$

The canonical angles [16] are directly related to the so called "direct rotation" from one subspace into another: among all orthonormal transformations rotating one space into another, the direct rotation differs minimally (in Frobenius sence) from the unit transformation [58]. In particular, the largest canonical angle is related to the concept of "gap" γ [see 15] between any two subspaces S_1 and S_2:

$$\sin(\vartheta_1) = \gamma(S_1,S_2) = \sup_{\substack{x \in X \\ \|x\|=1}} [\, \inf_{y \in Y} \|x-y\|_2] = \sup_{\substack{y \in Y \\ \|y\|=1}} [\, \inf_{x \in X} \|x-y\|_2] \qquad (3\text{-}8)$$

(a restricted definition for the 2-norm, in which case γ is a metric).

As one sees, the above theorem is quite elaborated, but at the same time not so easy for intuitive interpretation. Therefore, the author likes a simpler but more suggestive approach: The idea is

1. to separate the effects related to the singular spectrum, and those related to the singular subspaces

2. to force these singular values and spaces away as far as possible while perturbing matrix A as little as possible.

To do so, we will use the experience, built up in chapter 2:

Theorem 3-3 (On rotations of maximal sensitivity):

For all $A = U \Sigma V$:

1. If $A+dA = U.(\Sigma + d\Sigma).V$ then $\|dA\|_F = \|d\Sigma\|_F$ (3-9)

2. If $A+dA = (U.G).\Sigma.(H.V)$ then $\|dA\|_F = \frac{\sigma_i - \sigma_j}{\sqrt{2}} . \sin(z)$ (3-10)

where i,j,G,H and z are related by:

$$G = \begin{array}{c} \begin{array}{cccccccccc} 1 & 2 & .. & \downarrow i & .. & \downarrow j & .. & M-1 & M & \end{array} \\ \begin{bmatrix} 1 & 0 & .. & 0 & .. & 0 & .. & 0 & 0 \\ 0 & 1 & .. & 0 & .. & 0 & .. & 0 & 0 \\ . & . & .. & . & .. & . & .. & . & . \\ 0 & 0 & .. & \cos(z) & & \sin(z) & .. & 0 & 0 \\ . & . & .. & . & .. & . & .. & . & . \\ 0 & 0 & .. & -\sin(z) & .. & \cos(z) & .. & . & . \\ . & . & .. & . & .. & . & .. & . & . \\ 0 & 0 & .. & 0 & .. & 0 & .. & 1 & 0 \\ 0 & 0 & .. & 0 & .. & 0 & .. & 0 & 1 \end{bmatrix} \end{array} \begin{array}{l} 1 \\ 2 \\ . \\ \leftarrow i \\ . \\ \leftarrow j \\ . \\ M-1 \\ M \end{array} \quad (3\text{-}11a)$$

$$H = \begin{array}{c} \begin{array}{cccccccccc} 1 & 2 & .. & \downarrow i & .. & \downarrow j & .. & N-1 & N & \end{array} \\ \begin{bmatrix} 1 & 0 & .. & 0 & .. & 0 & .. & 0 & 0 \\ 0 & 1 & .. & 0 & .. & 0 & .. & 0 & 0 \\ . & . & .. & . & .. & . & .. & . & . \\ 0 & 0 & .. & \cos(z) & & -\sin(z) & .. & 0 & 0 \\ . & . & .. & . & .. & . & .. & . & . \\ 0 & 0 & .. & \sin(z) & .. & \cos(z) & .. & . & . \\ . & . & .. & . & .. & . & .. & . & . \\ 0 & 0 & .. & 0 & .. & 0 & .. & 1 & 0 \\ 0 & 0 & .. & 0 & .. & 0 & .. & 0 & 1 \end{bmatrix} \end{array} \begin{array}{l} 1 \\ 2 \\ . \\ \leftarrow i \\ . \\ \leftarrow j \\ . \\ N-1 \\ N \end{array} \quad (3\text{-}11b)$$

□

The proof and interpretation are simple, elegant, and direct:

- Eq.(3-9) guarantees that for strictly spectrum related effects, variations in the singular spectrum and on matrix A have exactly the same level, hence have the same (ideal) unit sensitivity in both forward and backward direction. The proof of this classical result is based on the (Frobenius) norm-preserving property for orthonormal matrices.

- Eq.(3-10) evaluates the variation in A, WHEN TWO SINGULAR VECTORS ARE ROTATED SIMULTANEOUSLY OVER AN ANGLE z, WITHIN THEIR OWN SPAN. In fact, what we want to obtain is a maximal forward effect on the matrices U and V, for a minimal effect in A. Therefore, we use all the energy of dA to rotate away as far as possible one pair of singular vectors, without affecting the others. As the variation in $U\Sigma V$ is concentrated in the singular vectors of two of its triplets only, it follows from the dyadic decomposition that the proof merely requires the verification of the identity:

$$\left\| \begin{bmatrix} \cos(z) & \sin(z) \\ -\sin(z) & \cos(z) \end{bmatrix} \cdot \begin{bmatrix} \sigma_i & 0 \\ 0 & \sigma_j \end{bmatrix} \cdot \begin{bmatrix} \cos(z) & -\sin(z) \\ \sin(z) & \cos(z) \end{bmatrix} - \begin{bmatrix} \sigma_i & 0 \\ 0 & \sigma_j \end{bmatrix} \right\|_F$$

$$= \sqrt{2}.(\sigma_i - \sigma_j).\sin(z) \qquad (3\text{-}12)$$

Though closely related to Stewarts general theorem 3-2, theorem 3-3 is believed to be an important new contribution to the perturbation analysis of Singular Spaces in that it allows for a complete and detailed decomposition of the sensitivity problem into the effect on any singular triplet:

- The effort, necessary in matrix A, to modify one singular value is equal to the required variation of this singular value. To concentrate the variation in one singular value only is possible by adding to matrix A a term proportional to the associated dyadic term. The efforts in A, necessary to vary different singular values are orthogonal and hence quadratically additive.

- The effort, necessary in matrix A, to modify one singular vector, can be decomposed into elementary rotations in 2-dimensional singular planes, spanned by the considered singular vector and any remaining one. The effort, necessary to rotate a singular vector in such a plane, is inversely proportional to the separation between the two involved singular values. In other words: bending a singular vector in the direction of another singular vector is much easier when the involved singular values are close.

Notice that both (3-9) and (3-10) are exact evaluations (not upperbounds) for the effects associated to elementary operations (modifying a singular value, rotating a pair of singular vectors...), that clearly distinguish the different types of perturbation.

Notice also, that eq.(3-10) continuously follows the effect of a simultaneous rotation z in U and V. It is conjectured that this angle z is the largest angle over which this pair of singular vectors can rotate for a given variation dA. If this would be confirmed by a proof, (3-10) describes explicitely the "worst sensitivity" subspace perturbation.

B. NUMERICAL COMPUTATION OF THE SVD.

There is no doubt that Golubs method [5] leads to the most efficient and most reliable algorithm for the computation of a single Singular Value Decomposition of a general matrix A. Golubs main achievement was to avoid a dangerous numerical booby-trap, that had not been really recognized until the mid-sixties [4]. As this pitfall still frequently occurs in many well established theories, and as one of the purposes of this thesis is to eliminate some of them (see e.g. chapter 8), we briefly describe the problem in its purest setting:

The most obvious, and apparently innocent way to attack the SVD-computation problem is to use the very close relationship that exists between the singular structure of matrix A, and the eigenstructure of AA^t and A^tA.

Indeed, from Eqs.(1-1),(1-4) one derives:

$$A\,A^t = U\Sigma\Sigma^t U^t = U\,.\Sigma\Sigma^t.\,U^{-1} \qquad (3\text{-}13a)$$

$$A^tA = V^t\Sigma^t\Sigma\,V = V^{-1}.\Sigma^t\Sigma\,.\,V \qquad (3\text{-}13b)$$

So the left (resp., right) singular vectors are the eigenvectors of AA^t (resp., A^tA), and the singular values of A are the positive square roots of the eigenvalues, common to AA^t and A^tA. Notice that both AA^t and A^tA are symmetric, nonnegative definite matrices, which guarantees the existence of real nonnegative eigenvalues $\sigma(AA^t)$ and $\sigma(A^tA)$ (the squares of the singular values), and a complete set of orthonormal eigenvectors (the left or right singular vectors). This conceptually attractive (algebraic) relationship has been the origin of many computational disasters. It cannot be stressed enough that the SVD of a matrix should NEVER be computed using AA^t or A^tA. Neither should the generalized (Least Squares) solution x_g of an overdetermined set of linear equations be computed as:

$$x_g = (A^tA)^{-1}.\,A^ty \qquad (3\text{-}13c)$$

Example 1-2: The exact singular values of

$$A = \begin{bmatrix} 1 & 0.0001 & 0 \\ 1 & 0 & 0.0001 \end{bmatrix} \qquad (3\text{-}14a)$$

are $\sqrt{2.00000001}$ and 0.0001. But

$$AA^t = \begin{bmatrix} 1.00000001 & 1 \\ 1 & 1.00000001 \end{bmatrix} \qquad (3\text{-}14b)$$

Now in $\varepsilon = 10^{-7}$-precision this is represented as

$$AA^t = \begin{bmatrix} 1.0000000 & 1.0000000 \\ 1.0000000 & 1.0000000 \end{bmatrix} \qquad (3\text{-}14c)$$

which has eigenvalues 2 and 0. This means that in $\varepsilon = 10^{-7}$-precision the second singular value has completely vanished, which represents an error of 0.0001 on this singular value, and causes a reconstruction error $\|A-U\Sigma V\|_F$ of the same level. This is much more than the sensitivity of the problem allows from variations on the 7-th digit, due to finite data storage precision, and rounding error accumulation, due to the finite precision of the computations. The loss of accuracy occurs because in AA^t 0.0001x0.0001 is computed and added to 1x1 = 1. With Golubs algorithm, this dangerous squaring is avoided, and a decomposition assuring a reconstruction $U\Sigma V$ within a modest multiple of the sensitivity bounds is found.

ALGORITHM 3-1 (Golubs SVD algorithm for real matrices [5,14]):

The SVD of a real matrix A can be computed as follows:

1. With elementary Householder or Givens transformations on rows and columns, reduce A to upper bidiagonal form.

2. With QR iterations on this bidiagonal form, perform a further reduction to diagonal form (chasing).

3. The accumulation of orthonormal transformations (Householder and/or Givens operations) on the left and right of matrix A results in the U, resp., V matrices.

4. The resulting diagonal matrix is matrix Σ. If necessary, a negative sign is transferred to U or V, and the singular triplets are reordered according to decreasing magnitude of the SV's.

Algorithm (3-1) is EFFICIENT and BACKWARD NUMERICALLY STABLE, i.e.:

1. EFFICIENT: The SVD is performed in k.MNP "ax+b"-type elementary operations, where k varies from 3 to 20 for typical examples [10].

1. BACKWARD NUMERICALLY STABLE: For ALL matrices A, the factors U and V, obtained from the algorithm, are orthonormal within machine precision ε_m, and the product $U\Sigma V$ is equal to A within a modest multiple of ε_m [14].

This well known algorithm and its properties were originally described in [5], they are clearly treated in [14] for didactical purposes, whereas the FORTRAN code, fully tested and documented is available in [10]. So we are not going to repeat this here. Instead, we will concentrate on the conceptual significance of SVD as a numerical tool in linear modelling and its applications (chapters 4,7,8,9,10). Our only contribution to the numerical computation of SVD will be an algorithm ASVD for the efficient recursive computation of large sets of slowly varying, large, low-rank Hankel matrices (chapter 5), as it occured during the study of the realization problem (chapter 9) in adaptive SSR identification (chapter 10).

Simply remember for later comparison with ASVD that in the classical algorithm matrix A is destroyed during the computation, and that the first reduction step is deterministic and requires O(MNP) elementary operations.

C. SENSITIVITY PROPERTIES OF A SET OF LINEAR EQUATIONS.

The sensitivity problem for sets of linear equations makes sense in an appropriate rank context only. If a level α of inaccuracies on A is assumed, and

$$R \triangleq \min_{\|dA\|_F < \alpha} [\mathrm{Rank}(A+dA)] \qquad (3\text{-}15)$$

then if more than R terms of the dyadic decomposition effectively participate in the solution, no bound on the sensitivity can be guaranteed.

Indeed, in that case the components of the error on x

$$v_p^t \cdot \sigma_p^{-1} \cdot u_p^t . dy \quad , \quad p > R \tag{3-16}$$

can blow up without bound for arbitrarily small perturbations on dy. We will therefore assume that only R terms are taken into account (see chapter 4). Then the sensitivity properties of a set of linear equations are well established, and usually stated in terms of the following quantities:

Definitions [14]:

$$\alpha \triangleq \alpha_2 \triangleq \frac{\|dA\|_2}{\|A\|_2} \quad , \quad \text{and} \quad \alpha_F \triangleq \frac{\|dA\|_F}{\|A\|_F} \tag{3-17a}$$

$$\beta \triangleq \frac{\|dy\|_2}{\|y\|_2} \tag{3-17b}$$

$$\gamma \triangleq \frac{\|y\|_2}{\|x\|_2} \cdot \frac{1}{\|A\|_2} \leqslant \frac{\|y\|_2}{\|Ax\|_2} \tag{3-17c}$$

$$\rho \triangleq \frac{\|y - Ax\|_2}{\|A\|_2 \cdot \|x\|_2} \leqslant \frac{\|y - Ax\|_2}{\|Ax\|_2} \tag{3-17d}$$

$$K \triangleq \|A\|_2 \cdot \|A^+\|_2 = \frac{\sigma_1}{\sigma_R} \tag{3-17e}$$

$$\hat{K} \triangleq \frac{K}{1 - K.\alpha} = \frac{\|A\|_2 \cdot \|A^+\|_2}{1 - \|dA\|_2 \cdot \|A^+\|_2} \tag{3-17f}$$

$$dx \triangleq A^+ y - (A+dA)^+ . (y+dy) \tag{3-17g}$$

$$x \triangleq A^+ y \tag{3-17h}$$

where A^+ is the rank-R generalized inverse. Then the following relations are available [14]:

Theorem 3-4 (on the sensitivity of a set of linear equations):

$$\text{If} \quad \|dA\|_2 \cdot \|A^+\|_2 < 1 \tag{3-18}$$

$$\text{then} \quad \frac{\|dx\|_2}{\|x\|_2} \leq \hat{K}\alpha + \hat{K}\beta\gamma + \hat{K}K\rho\alpha + \hat{K}\alpha \qquad (3\text{-}19a)$$

$$\leq \hat{K}\,[(2+K\rho)\alpha + \gamma\beta] \qquad (3\text{-}19b)$$

for dA that do not alter the rank of A. For full-rank problems (R=P), the latter condition is always fulfilled, leading to further simplification:

$$\text{For } M > N = R: \quad \leq \hat{K}\,[(1+K\rho)\alpha + \gamma\beta] \qquad (3\text{-}20a)$$

$$\text{For } M = N = R: \quad \leq \hat{K}\,(\alpha+\beta\gamma) \leq \hat{K}(\alpha+\beta) \qquad (3\text{-}20b)$$

$$\text{For } N > M = R: \quad \leq \hat{K}\,(\alpha+\beta\gamma) + K\alpha \leq \hat{K}\,(2\alpha+\beta\gamma) \leq \hat{K}(2\alpha+\beta) \qquad (3\text{-}20c)$$

The above expressions are careful evaluations of forward sensitivity of the solution of a set of linear equations to small changes in A and y [14]. Most of them follow trivially from the conceptual interpretation in chapter 2. Let us for example explain 3-20b: clearly, the relative error on x is sensitive to variations α on A, and to variations β on y.

- Assume for a while a correct matrix A. Then for $dy // u_R$, and $y // u_1$, it directly follows from Fig.2-1 that [all in 2-norm; see also eq.(2-44)]:

$$\forall\, y,dy: \quad \frac{\|dx\|}{\|x\|} \leq K(A) \cdot \frac{\|dy\|}{\|y\|} \qquad (3\text{-}21a)$$

Analogously, one easily understands

$$\forall\, dy: \quad \frac{\|dx\|}{\|x\|} \leq K(A) \cdot \frac{\|y\|}{\sigma_1 \cdot \|x\|} \cdot \frac{\|dy\|}{\|y\|} \qquad (3\text{-}21b)$$

(3-21a) and (3-21b) are the same as (3-20b), for $\alpha = 0$.

- Now consider additional variation on A ($\alpha \neq 0$). The first effect is the influence of a change in A on the PROPAGATION OF THE RHS-ERROR. From the dyadic decomposition, it follows that dA may entirely influence A in its weakest singular value σ_R, and hence:

$$dA = \|dA\| \cdot u_R\, u_R^t \qquad (3\text{-}22)$$

which can change K maximally into $\hat{K}$:

$$\hat{K} = \frac{\sigma_1}{\sigma_R - \|dA\|_2} = \frac{\sigma_1 / \sigma_R}{1 - \|dA\|_2/\sigma_R} = \frac{K}{1 - K\alpha_2} \geq K \qquad (3\text{-}23)$$

This explains the replacement of K by $\hat{K}$ for the sensitivity to changes in the RHS. The second effect is purely due to the change of A, with respect to the UNCHANGED RHS y, and maximal for $y // u_R$:

$$\left\| dx_{(dA,y)} \right\| = \frac{\|y\|}{\sigma_R} - \frac{\|y\|}{\sigma_R - \|dA\|_2} = \alpha \cdot \hat{K} \cdot x \qquad (3\text{-}24)$$

and can be added directly to effect 1.

Example 3-1 (an illustrative example):

The purpose of this small example is to get a better feeling of what is conceptually behind all the above algebra. Consider

$$\begin{bmatrix} 1 & 10 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \qquad (3\text{-}25)$$

The SVD of the involved matrix is :

$$\begin{bmatrix} 1 & 10 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.9951 & -0.0985 \\ 0.0985 & 0.9951 \end{bmatrix} \cdot \begin{bmatrix} 10.099 & 0 \\ 0 & 0.0990 \end{bmatrix} \cdot \begin{bmatrix} 0.0985 & 0.9951 \\ -0.9951 & 0.0985 \end{bmatrix} \qquad (3\text{-}26)$$

Now define $m_x \triangleq \dfrac{\|x\|_2}{\|Ax\|_2}$ for $x = \begin{bmatrix} \cos(x,v_1) \\ \cos(x,v_2) \end{bmatrix} = \begin{bmatrix} \cos(\varphi) \\ \sin(\varphi) \end{bmatrix}$ (3-27a)

and $m_y \triangleq \dfrac{\|A^+ y\|_2}{\|y\|_2}$ for $y = \begin{bmatrix} \cos(y,u_1) \\ \cos(y,u_2) \end{bmatrix} = \begin{bmatrix} \cos(\psi) \\ \sin(\psi) \end{bmatrix}$ (3-27b)

Both Eqs.(3-27) measure the norm amplification when solving x from the set of equations Ax=y, but in eq.(3-27a) the orientation of vector x is taken as independent variable, whereas in 3-27b, it is vector y that can vary freely in the range space. Both functions can be simply evaluated:

$$m_x^2 = \left[\sigma_1^2 \cdot \cos^2(x,v_1) + \sigma_2^2 \cdot \cos^2(x,v_2) \right]^{-1} \qquad (3\text{-}28a)$$

$$m_y^2 = \sigma_1^{-2} \cdot \cos^2(y,u_1) + \sigma_2^{-2} \cdot \cos^2(y,u_2) \qquad (3\text{-}28b)$$

In BOTH cases σ_1^{-1} is the minimum, and $\sigma_R^{-1} = \sigma_2^{-1}$ is the maximum of the curves. But on the average, m_x is closer to its minimum, whereas m_y is closer to its maximum. In Fig. 3-1, $m_x(\varphi)$ and $m_y(\psi)$ are plotted, where φ and ψ denote the angles with respect to v_1 and u_1 respectively.

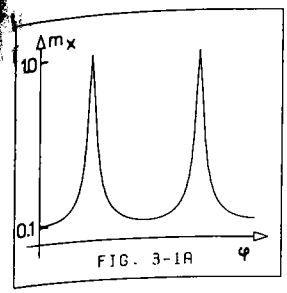


FIG. 3-1A

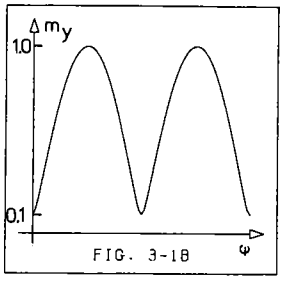


FIG. 3-1B

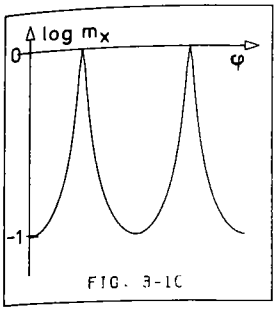


FIG. 3-1C

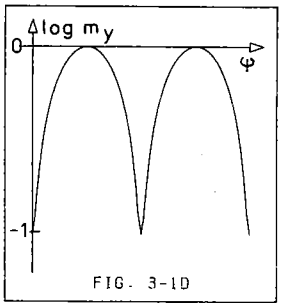


FIG. 3-1D

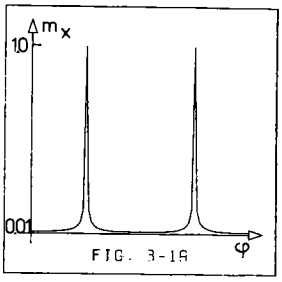


FIG. 3-1A

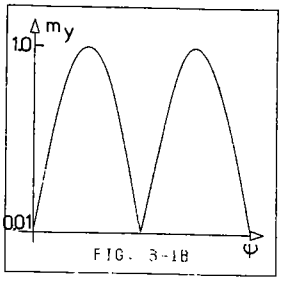


FIG. 3-1B

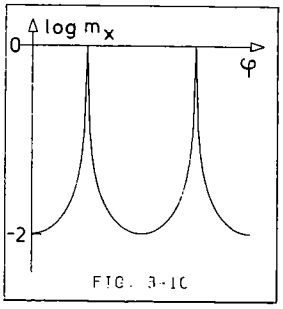


FIG. 3-1C

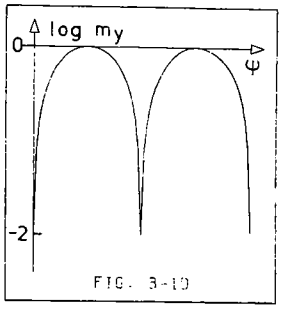


FIG. 3-1D

K(A) = 10 K(A) = 100

Fig. 3-1: The functions $m_x(\varphi)$ and $m_y(\psi)$ for K(A)=10 and K(A)=100

Notice that the property of "being usually close" to either the maximum or the minimum significantly increases with increasing condition number.

Notice also that the only difference between (3-28a) and (3-28b) is a different choice of the independent variables φ and ψ, which are related in a nonlinear fashion. But this distinction is extremely significant for modelling purposes:

- The orientation of dy is unpredictable with respect to the main u-orientations of A. Hence it is meaningful to assume their effect on x to be of a level close to $\sigma_R^{-1} \cdot \|dy\|_2$. This level should then be compared with the expected length of the exact part of y:

- If in the considered problem the x-orientation is not correlated with the v-orientations of A, then each φ interval (3-27a) is equally probable to occur, most vectors y will be oriented close to u_1, and it is most probable that x has a length close to $\sigma_1^{-1} \cdot \|y\|_2$ (Fig.3-1a).

- If on the other hand the orientation of y is not correlated with the main u-orientations of A, then most vectors x will be oriented close to v_R, and x will most probably have a length close to $\sigma_R^{-1} \cdot \|y\|_2$ (Fig.3-1b).

An example for the first situation is the identification of an impulse response H of a linear system from input-output measurements (u_i, y_i): the input signal u_i forms the Hankel matrix U, and the unknown impulse response is independent from U (see chapter 10). The RHS is the product

H.U, hence y will mainly be oriented in strong u-directions, and the effect on x, of variations dy will rather be of a level $K(A).\|dy\|_2$ (bad luck)

An example for the second situation is the problem of fitting an N-th degree polynomial through a set of N+1 arbitrary couples (α_m, y_m). Then powers of the abcissas form the Vandermonde matrix A with elements

$$a_{m,n} = \alpha_m^{n-1} \quad , \quad m=1...N+1 \; ; \; n=1...N+1 \qquad (3\text{-}29)$$

and the ordinates y_m form the RHS. Clearly in this case, the RHS-orientation is independent from the columns of A, hence x will be oriented mainly close to the strong v-directions of A, and the effect of variations dy on x will rather be of a level $K_y(A).\|dy\|_2$ (luck).

The distribution $m_e = \dfrac{\|dx\|_2 / \|x\|_2}{\|dy\|_2 / \|y\|_2}$ for inaccuracies on the RHS can now easily be obtained for all orientations of y and dy [Fig.(3-2)]:

- If $y \,//\, u_R$ and $dy \,//\, u_1$, then the solution x will be a factor K(A) MORE PRECISE than the given RHS (!); if $y \,//\, dy$, the presicion of both the RHS and the solution are THE SAME; and if $y \,//\, u_1$ and $dy \,//\, u_R$ then the solution x will be a factor K(A) LESS PRECISE than the given RHS.

- For a fixed y, m_e varies from $K_y(A)/K(A)$ to $K_y(A)$, depending on the orientation of dy. Of course, this orientation of dy is not known, and so one should assume the worst case for dy : $m_e = K_y(A)$.

- Over ALL possible orientations of y, $K_y(A)$ varies from 1 to K(A).

Fig.3-2 also gives a good idea about the genericity of large error magnifications:

- If dy has a uniform probability distribution in S^M, then m will usually be close to its upperbound $K_y(A)$ [Figs.3-2a and 3-2c]. In this (most common) situation $K_y(A)$ is a realistic measure for m_e. If dx has a uniform probability distribution, then m_e will usually be much smaller than $K_y(A)$.

- The distribution of y depends on the considered problem, and leads to a USUALLY dy-INSENSITIVE SOLUTION if y is independent from A [Figs.3-2a and 3-2b]; to a USUALLY dy-SENSITIVE SOLUTION if x is independent of A [Figs.(3-2c) and 3-2d)].

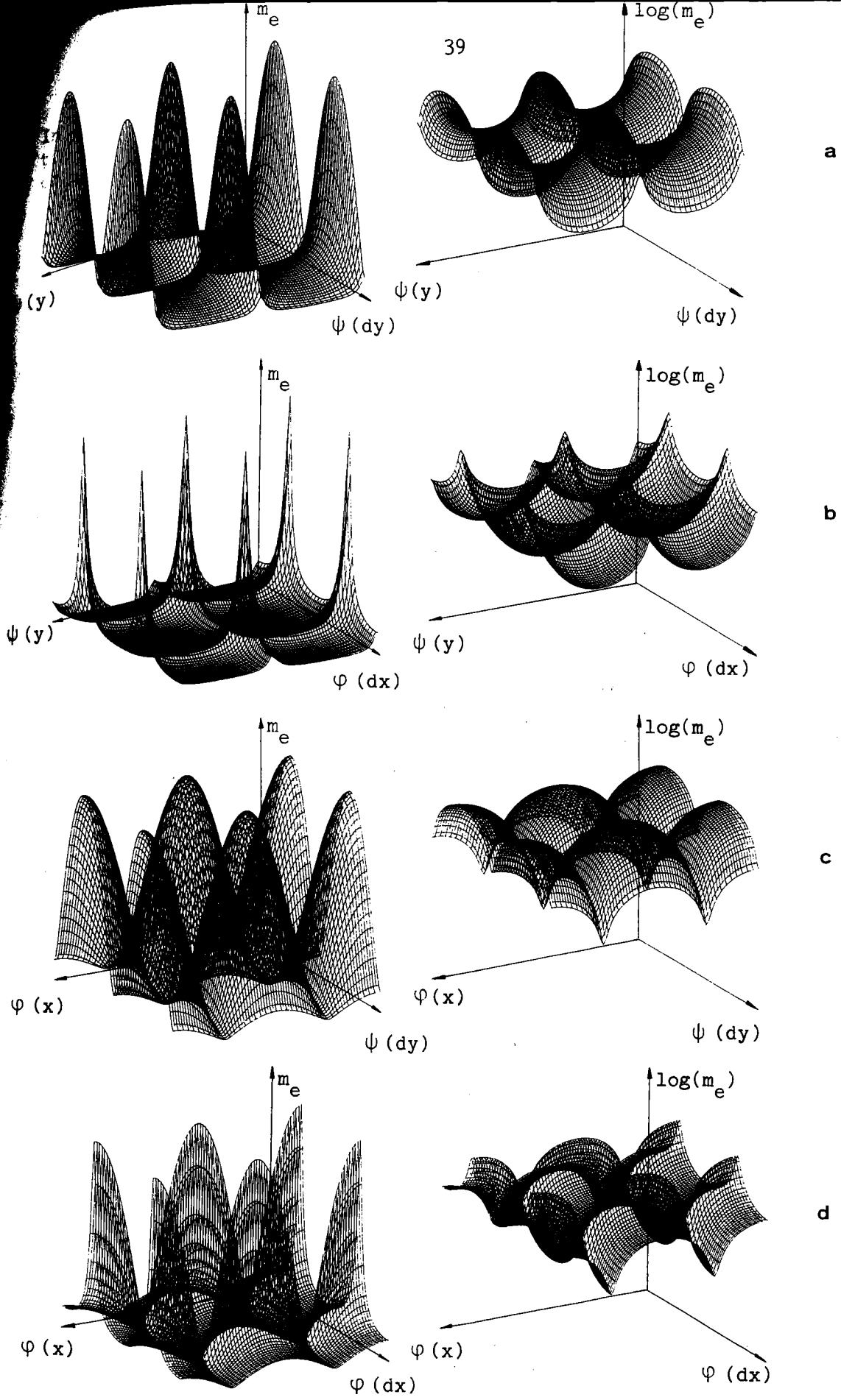


Fig.3-2: Error magnification distributions (linear and log. scale) for example 3-1, and for the 4 possible uniform distributions.

a. Uniform in y and dy
b. Uniform in y and dx
c. Uniform in x and dy
d. Uniform in x and dx

In many cases, the matrix A is much more precisely known than the RHS entries in y. Then the above analysis is applicable as such. If the entries of A may also vary over significant ranges, then this additional effect should be taken into account, which will (partially) reduce the importance of the above described effects.

D. NUMERICAL COMPUTATION OF THE SOLUTION OF SETS OF LINEAR EQUATIONS.

For completeness, we resume the error propagation bounds for algorithms, using orthogonal reduction techniques. Let us remind here, that if one is only interested in the solution of the given set of equations, it is not necessary to compute explicitely the SVD of the involved matrix A. The following results are mainly due to Wilkinson, but are extremely well described by Lawson and Hanson [14], to which we refer for more details.

a. Mono-precision arithmetic.

In case all computations are performed with the same machine precision, the error propagation can be limited within well defined bounds.

Theorem 3-5 (on the error propagation in mono-precision LLS algorithms):

Let A be an MxN matrix of pseudorank $R \leqslant P = \mathrm{Min}(M,N)$, and y be an M-vector. Then with ε precision arithmetic, a solution x can be obtained, that is the exact (minimum-length) LLS solution of a problem

$$(A+dA)(x+dx) = y+dy \qquad (3\text{-}30)$$

a. If M>N=R :

$$\|dA\|_F \leqslant (6M-3N+41).N\varepsilon.\|A\|_F + O(\varepsilon^2) \qquad (3\text{-}31a)$$

$$\|dy\|_2 \leqslant (6M-3N+40).N\varepsilon.\|y\|_2 + O(\varepsilon^2) \qquad (3\text{-}31b)$$

$$dx = 0 \qquad (3\text{-}31c)$$

b. If M=N=R :

$$\|dA\|_F \leqslant (3N^2+41N).\varepsilon.\|A\|_F + O(\varepsilon^2) \qquad (3\text{-}32a)$$

$$\|dy\|_2 \leqslant (3N^2+40N).\varepsilon.\|y\|_2 + O(\varepsilon^2) \qquad (3\text{-}32b)$$

$$dx = 0 \qquad (3\text{-}31c)$$

c. If R=M<N :

$$\|dA\|_F \leqslant (6N-3M+41).M\varepsilon.\|A\|_F + O(\varepsilon^2) \qquad (3\text{-}33a)$$

$$dy = 0 \qquad (3\text{-}33b)$$

$$\|dx\|_2 \leqslant (6N-3M+40).M\varepsilon.\|x\|_2 + O(\varepsilon^2) \qquad (3\text{-}33b)$$

d. If $R < P$: $\|dA\|_F \leq (6M+6N-6R-3P+84).R\varepsilon . \|A\|_F + Q + O(\varepsilon^2)$ (3-34a)

$\|dy\|_2 \leq (6M-3R+40).R . \|y\|_2 + O(\varepsilon^2)$ (3-34b)

$\|dx\|_2 \leq (6N-6R+43).R . \|x\|_2 + O(\varepsilon^2)$ (3-34b)

where R is determined by the algorithm and Q is the Frobeniusnorm of the submatrix, dropped during the computations due to reduced rank approximation.

b. Mixed precision arithmetic.

In case it is possible to perform the computations with a higher precision at selected points in the algorithm, then the error propagation can be limited within much narrower bounds, with virtually no additional storage nor computation cost.

Theorem 3-6 (on the error propagation in mixed precision LLS algorithms):

Let A be an MxN matrix of pseudorank R P = Min(M,N), and y be an M-vector. Then with ε precision arithmetic but extended precision facilities at certain selected points in the algorithm, a solution x can be obtained, that is the exact (minimum-length) LLS solution of a problem

$(A+dA)(x+dx) = y+dy$ (3-35)

a. If M>N=R : $\|dA\|_F \leq 30N.\varepsilon . \|A\|_F + O(\varepsilon^2)$ (3-36a)

$\|dy\|_2 \leq 29N.\varepsilon . \|y\|_2 + O(\varepsilon^2)$ (3-36b)

$dx = 0$ (3-36c)

b. If M=N=R : $\|dA\|_F \leq 30N.\varepsilon . \|A\|_F + O(\varepsilon^2)$ (3-37a)

$\|dy\|_2 \leq 29N.\varepsilon . \|y\|_2 + O(\varepsilon^2)$ (3-37b)

$dx = 0$ (3-37c)

c. If R=M<N : $\|dA\|_F \leq 30M.\varepsilon . \|A\|_F + O(\varepsilon^2)$ (3-38a)

$dy = 0$ (3-38b)

$\|dx\|_2 \leq 29M.\varepsilon . \|y\|_2 + O(\varepsilon^2)$ (3-38c)

d. If $R < P$:

$$\|dA\|_F < 59R.\varepsilon.\|A\|_F + Q + O(\varepsilon^2) \qquad (3\text{-}39a)$$

$$\|dy\|_2 < 29R.\varepsilon.\|y\|_2 + O(\varepsilon^2) \qquad (3\text{-}39b)$$

$$\|dx\|_2 < 29R.\varepsilon.\|x\|_2 + O(\varepsilon^2) \qquad (3\text{-}39c)$$

where R is determined by the algorithm and Q is the Frobeniusnorm of the submatrix, dropped during the computations due to reduced rank approximation.

Chapter 4

ON GLOBAL LINEAR LEAST SQUARES APPROXIMATION

This will be a short chapter. An interlude of rather phylosophical content included more for relaxing freshness than with the aspiration to question the two centuries of extensive investigation, that have lead to the nowadays well established and unassailable theory of Least Squares Techniques: exegit Gauss monumentum aere perennius, and who would DARE question the Master.

The idea that will be developped here is far from being fully analysed. But it is simple, fresh, and unsophisticated. It has improved the author's insight in the axiomatic background of classical Least Squares techniques for sets of linear equations. It is one of the purposes of this chapter to transmit this gain in understanding, the other being to propose a more general approach to solving sets of equations in LLS sense.

A. On minimizing $\|A\xi - y\|$.

To fix the ideas, assume the specific problem of a full rank overdetermined set of (conflicting) linear equations Ax=y. To avoid any discussion on preprocessing of the available data (scaling, coordinate transformation, whitening of the data..., which are NOT pertinent here), we assume that all measures of this kind have already been taken, or even more simply: that all the available data are EXACT, that NO STATISTICS WHATSOEVER on the expected A and or y are available, and that all computations are performed in EXACT arithmetic (which then implies that the given data satisfy some totally unknown (slightly) nonlinear relation from which one wants to extract a linearized-model solution.

The usual technique then is to extract a least squares solution x' from the following minimizing procedure [14]:

$$x' : \|Ax'-y\|_2 = \min_{\xi \in S^N} \|A\xi - y\|_2 \qquad (4\text{-}1)$$

What we want to discuss now is NOT how to find this x' in an efficient way, but rather wether the problem thus stated is really meaningful. Let us therefore analyse this classical definition in the conceptual context of chapter 2: Eq.(4-1) is equivalent to

$$x' : Ax' = y' \ , \ \|Ax'-y\|_2 = \min_{\eta \in Im(A)} \|\eta - y\|_2 \qquad (4\text{-}2)$$

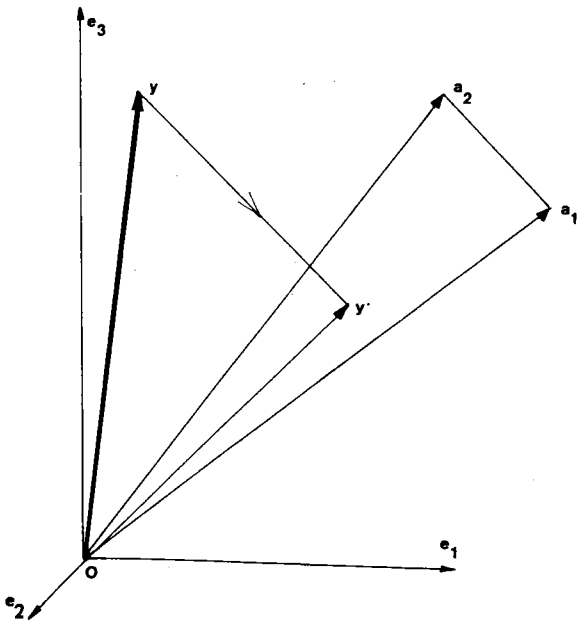


Fig.4-1: Illustration of the classical LLS technique.

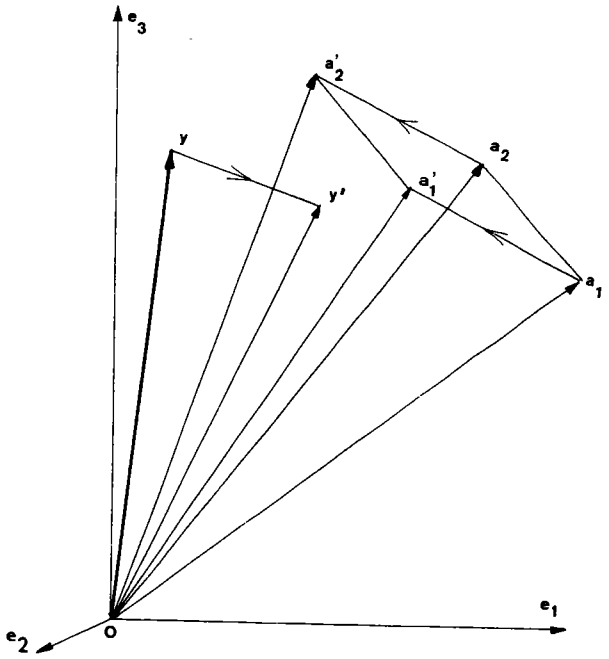


Fig.4-2: Illustration of the new GLS technique.

with the well known orthogonal projection interpretation [Fig.(4-1)], in which x' is the exact solution of Ax'=y', and y' is the projection of y into Im(A), i.e. the vector in Im(A), that is as close as possible to the given y. So what (4-1) and (4-2) in fact do, is to look for the solution of a different problem, in which the Right Hand Side y is altered (as little as possible) in order to reduce the original (conflicting) problem into one that has an exact solution. The point that we want to make now is, that this modification is COMPLETELY imposed on the Right Hand Side y, which seems rather unfair if the statement Ax=y is interpreted as the problem to reconstruct the RHS y, using the columns available in A:

$$x' : \left\| \sum_{n=1}^{N} a_n . x'_n - y \right\|_2 = \min_{\xi_1, \ldots, \xi_N} \left\| \sum_{n=1}^{N} a_n \xi_n - y \right\|_2 \qquad (4\text{-}3)$$

Indeed, when there is no reason to assume the entries in A to be more accurate than the entries in y, there is no motivation to keep the columns in A unchanged, and to impose all the modification on y. Except for the evident reason of justice, there is also a very practical argument to spread the modification effort on BOTH A and y, which is the following:

First: the total effort to modify both A and y into A' and y' corresponding to an exactly solvable set of equations is never larger, and usually smaller than the effort necessary to perform this modification using y only.

Second: when the modification energy can be spread over (N+1).M entries, the modification per entry can generically be kept smaller than when it has to be spread over M entries.

Finally: when the conflict in the overdetermined set of equations Ax=y

is due to inaccuracies in the acquisition or storage of the entries in A and y, it is always possible to guarantee a solution, that is the exact solution to an A AND y modified problem within the same perturbation level, but it is NOT necessarily possible to find an y-perturbed problem with an exact solution, within this neighbourhood.

We now describe an approach that allows to perturb "globally" the equation Ax=y into an exactly solvable set of equations.

B. THE GLOBAL LLS APPROACH.

Inspired by eq.(4-2), consider the following alternative to eq.(4-1):

$$x': A'x'=y', \left\| [A'\ y']-[A\ y] \right\|_2 = \min_{B,z:\mathrm{rank}[B\ z]=\mathrm{rank}[B]} \left\| [B\ z]-[A\ y] \right\|_2 \qquad (4\text{-}3)$$

In eq.(4-3), the RHS y that has to be constructed, and the columns of A that are used for this construction, are treated on a basis of equal importance. What (4-3) looks for, is a hyperplane close to BOTH y and the columns of A. By an orthogonal projection of BOTH y and the columns of A, the original (conflicting) set of equations can then be GLOBALLY modified into an exactly solvable one, spreading the perturbation over BOTH A and y Fig.(4-2).

The rank condition as minimization constraint watches that y' is really linearly dependent on the columns of A', thus avoiding the (nongeneric) situation, in which the columns of A' and y' form a linearly dependent set, in which y' however is an essentially independent base vector. In that case indeed, y'can still not be constructed as an exact linear combination of the columns of A! In fact this rank condition is the basis for the RANK DECISION in the case of rank deficient problems: the rank of a numerically rank deficient problem should be determined as follows:

- Determine the closest approximation [A" y"] of [A y] of smallest possible rank R within the uncertainty level on the given data.

- Then find the closest approximation [A' y'] of [A" y"] of largest rank $r \leqslant R$, that satisfies the condition

$$\mathrm{rank}\ [A'\ y'] = \mathrm{rank}\ [A'] \qquad (4\text{-}4)$$

Such a choice guarantees that x' can be exactly solved from A'x'=y', that no effects above accuracy level are unnecessarily neglected, and that no effect of inaccuracies can alter the rank-properties of the solution.

- If [A' y'] is within the accuracy level or expected modification distance, one can call x' an APPROXIMATE SOLUTION.

- If [A' y'] is beyond the accuracy level or expected modification distance, one better calls x' a LINEARIZED APPROXIMATION.

C. ON THE COMPUTATION OF GLS SOLUTIONS.

a. An algorithm: A general (explicit) procedure is as follows:

1. Compute $[A_{MN}\ y\ ] = U_{M,N+1}.\Sigma_{N+1,N+1}.V_{N+1,N+1}$ (4-5)

2. Compute $[A'_{MN}\ y'] = U_{M,r}\cdot\Sigma_{r,r}\ .V_{r,N+1}$ (4-6)

3. Find the exact solution of $A'_{MN}.x' = y'$ (4-7)

For the generic case of a full rank overdetermined set of equations, the rank r is directly r=R=P=Min(M,N)=N. In case of rank deficiency, the above described condition can be checked for ranks of [A' y'], decreasing from N downwards. If r turns out to be zero, then x'= 0.

Notice that as in the case of classical LS procedures, it is certainly not necessary to perform the SVD explicitely, and the solution should be obtainable with techniques that compute the solution directly, and are by far more time and storage efficient.

b. An extreme example: Example 4-1.

Consider the following overdetermined set of N linear equations in N-2 unknowns:

$$\begin{bmatrix} N-1 & -1 & \dots & -1 \\ -1 & N-1 & \dots & -1 \\ \dots & \dots & \dots & \dots \\ -1 & -1 & \dots & N-1 \\ -1 & -1 & \dots & -1 \\ -1 & -1 & \dots & -1 \end{bmatrix}_{N,N-2} \cdot \begin{bmatrix} x_1 \\ \dots \\ \dots \\ x_{N-3} \\ x_{N-2} \end{bmatrix}_{N-2,1} = \begin{bmatrix} -1 \\ -1 \\ \dots \\ -1 \\ N-1 \\ -1 \end{bmatrix}_{N,1} \qquad (4\text{-}6)$$

With the classical LS technique:

$$x' = A^{+}.y = -0.5\ [\ 1\ \dots\ 1\ \dots\ 1\ ]^t \qquad (4\text{-}7a)$$

$$y' = 0.5\ [\ -2\ \dots\ -2,\ N-2,\ N-2\ ]^t \qquad (4\text{-}7b)$$

$$dy = 0.5\ [\ 0\ \dots\ 0\ ,\ N\ ,\ -N\ ]^t \qquad (4\text{-}7c)$$

and hence $$\frac{\|dy\|_2}{\|y\|_2} = \sqrt{\frac{N}{2.(N-1)}} \xrightarrow[\text{large } N]{} 0.707 \qquad (4\text{-}7d)$$

th the GLS technique:

$$[A'y'] = \frac{N}{N-1}\begin{bmatrix} N-2 & -1 & \dots & -1 & -1 \\ -1 & N-2 & \dots & -1 & -1 \\ \dots & \dots & \dots & \dots & \dots \\ -1 & -1 & \dots & N-2 & -1 \\ -1 & -1 & \dots & -1 & N-2 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix} \qquad (4\text{-}8a)$$

$$d[A\ y\ ] = \frac{1}{N-1}\begin{bmatrix} -1 & -1 & \dots & -1 & -1 \\ -1 & -1 & \dots & -1 & -1 \\ \dots & \dots & \dots & \dots & \dots \\ -1 & -1 & \dots & -1 & -1 \\ -1 & -1 & \dots & -1 & -1 \\ N-1 & N-1 & \dots & N-1 & N-1 \end{bmatrix} \qquad (4\text{-}8b)$$

$$x' = A^{+}.y' = - \begin{bmatrix} 1 & \dots & 1 & \dots & 1 \end{bmatrix}^{t} \qquad (4\text{-}8c)$$

$$\text{and hence} \quad \frac{\|dy\|_2}{\|y\|_2} = \frac{\|dA\|_F}{\|A\|_2} = \frac{1}{N-1} \xrightarrow[\text{large } N]{} 0 \qquad (4\text{-}8d)$$

Hence for large N in this example, if only the RHS is allowed to vary (the classical approach), y has to be modified over approximately 70% . If the matrix A is allowed to vary also (global LS), a small change of $1/(N-1)$ is sufficient. If for instance eq.(4-6) has been obtained by measurement and both the entries of A and y have been measured with a same absolute precision ε , then the RHS and each of the columns of A are available with a relative precision $\varepsilon\sqrt{N-1}$, and a solution in the classical LS sense will be merely a meaningless, purely formal approximation, whereas in the global LS sense, it is still possible to find a quite accurate approximate solution in a well defined and meaningful sense.

D. CONCLUSION.

We have shown that a more general definition for Least Squares approximation may be meaningful for modelling purposes. We have called the resulting technique the Global Least Squares approach, and tried to motivate its introduction in the context of what precisely should be understood by "close linear fitting of a set of noisy and/or essentially nonlinearly related data". We have clarified the conceptual interpretation, outlined an SVD based algorithm for its computation, and illustrated its significance with a quite extravagant example. It would not be honest however, if we would not mention some difficulties, that should still be investigated and clarified for the sake of full understanding, reliable implementation and correct interpretation. The difficulties occur when the RHS has a size

comparable with, or larger than the columns in matrix A, and when this RHS is nearly orthogonal to most columns in A. This situation means that the equations in Ax=y are highly conflicting, and is easily detected as a large rank defect before the minimization condition in (4-3) is decently fulfilled. If the author would be confronted with such a situation, he would simply conclude that the data are not appropriate for linear modelling, and as such reject the problem as irrelevant from a linear modelling point of view. Nevertheless, it remains true that in that case the GLS solution becomes quite sensitive to small variations on the data (without significant reduction of the approximation closeness), and this is a phenomenon that one can not simply ignore. Simple examples of such situations are

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1+\varepsilon \end{bmatrix} \qquad (4\text{-}9)$$

and

$$\begin{bmatrix} 1 & \varepsilon \\ 0 & \varepsilon \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad (4\text{-}10)$$

the analysis (and associated headaches) of which I generously leave to the interested reader.

Chapter 5

ASVD : A NEW ALGORITHM FOR THE RECURSIVE COMPUTATION OF THE SINGULAR VALUE DECOMPOSITION OF LARGE LOW RANK MATRICES.

To spend an entire chapter on a "new" SVD algorithm is certainly not without danger. The numerical analyst will probably feel some irritation and refer to Golub's work [4-5]. And there is little doubt indeed that under the set of assumptions, of use in numerical analysis, his algorithm will forever remain the most reliable, and the most efficient. These usual assumptions are that one wants to compute a SINGLE, COMPLETE factorization $A=U\Sigma V$ of a GENERAL MxN matrix A on a SEQUENTIAL, DIGITAL machine. Often, this setting is considered so natural, that there seems to be no relevant investigation to be performed in this domain. Yet, many applications require the SVD of a much more restricted class of matrices in quite a different situation. In the specific context of recursive SSR identification for example (chapter 10), the SVD of LONG SEQUENCES of SLOWLY VARYING LARGE, (BLOCK) HANKEL matrices of LOW (NUMERICAL) RANK have to be computed. Under such different assumptions, procedures that are far more time and storage efficient can be conceived, which in this case is essential for realistic on-line application.

This chapter describes a technique that arose in such a special purpose environment. It should be clear from the above, that the presented method does not intend to challenge Golub's algorithm in its usual, general purpose context. On the other hand, our algorithm is far from being simply a unique-purpose gadget of no further interest than the specific application it was designed for:

First: what follows is NOT an adaptation of Golubs algorithm, but is based on a totally different approach. The basic idea is so simple, that its analysis improved the conceptual insight of the author in the related linear algebraic problem, an effect that is hoped to be transmitted to the reader. Essentially the algorithm solves the problem in a way, that is very closely related to the geometrical interpretation of a matrix, seen as an operator, rather than by the formal manipulation of the entries of this matrix, seen merely as a table of numbers.

Second: the algorithmic setting is essentially modular, iterative, and data-flow minded. As such, it is flexible enough to conceive its implementation on multi-processor, parallel processing, data-flow, or even analog machines. The latter aspects are not worked out in this thesis, but will be mentionned at the appropriate places as an illustration of the algorithms conceptual flexibility.

Finally, the practical significance of ASVD will clearly appear in a practical (SSR identification-) context (Chapter 10). Other applications have not been fully explored. The author however firmly believes that the new approach will prove useful in many other fields, one of which is the domain of sparse matrix techniques (circuit analysis). It is the author's hope and belief that when sufficient experience will have been acquired with ASVD, this algorithm will soon prove useful in a broad variety of concrete applications.

Let the future decide on the relevance of the here described algorithm.

A.GENERAL DESCRIPTION OF THE ALGORITHM.

A suggestive description of the algorithm is given in Fig. 5-1. For any MxN matrix A, this scheme computes the largest R singular triplets.

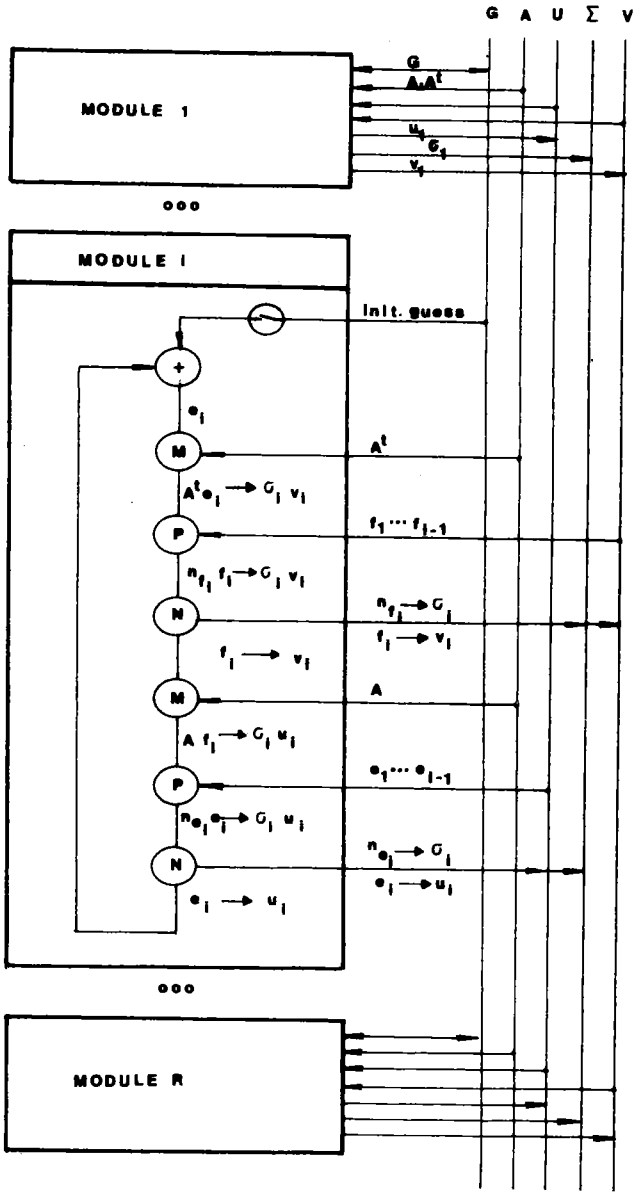


Fig. 5-1: General description of ASVD.

Each module M_i in Fig. 5-1 computes approximations e_i, s_i, f_i of increasing accuracy for the triplet u_i, σ_i, v_i. For this purpose it uses (besides some general information and possibly an initial guess) the UNALTERED matrix A and the available approximations e_j, s_j, f_j; j=1...i-1 of singular triplets, associated to larger singular values. To start up module M_i, a first

approximation e_i for u_i (if not known, a random vector) is temporarily fed to the module. The flow of data in each module is then as follows: vector e_i is multiplied with A^t, and the result is orthonormalized with respect to the available approximation

$$S_F^{i-1} = \mathrm{Span}_{col}[\, f_1 \ldots f_{i-1} \,] \qquad (5\text{-}1)$$

of the (i-1)-th principal singular space

$$S_V^{i-1} = \mathrm{Span}_{col}[\, v_1 \ldots v_{i-1} \,] \qquad (5\text{-}2)$$

The norm n_f resp., the normalized vector f_i are a new approximation for σ_i resp., v_i. This f_i is then multipied with A, orthonormalized with respect to S_E^{i-1}, and again normalized. This yields a new approximation for e_i, (again) for σ_i, and the loop is closed.

The given scheme is claimed to converge to the correct singular triplets in each module, at the places indicated. For infinite precision arithmetic this claim is easily verified, based on the following

Theorem 5-1: For all real MxN matrices A, and for p=1 ... P = Min(M,N):

$$\forall e \in S_U^M : \; tg\,(\, A^t e \,,\, S_V^p \,) \leqslant \left(\frac{\sigma_{p+1}}{\sigma_p}\right) . \; tg\,(\, e \,,\, S_U^p \,) \qquad (5\text{-}3a)$$

$$\forall f \in S_V^N : \; tg\,(\, A\, f \,,\, S_U^p \,) \leqslant \left(\frac{\sigma_{p+1}}{\sigma_p}\right) . \; tg\,(\, f \,,\, S_V^p \,) \qquad (5\text{-}3b)$$

$$\forall e \in (\, S_U^{p-1} \,)^{\perp} : \; A^t e \in (\, S_V^{p-1} \,)^{\perp} \qquad (5\text{-}4a)$$

$$\forall f \in (\, S_V^{p-1} \,)^{\perp} : \; A\, f \in (\, S_U^{p-1} \,)^{\perp} \qquad (5\text{-}4a)$$

$$\forall e \in S_U^M : tg\,[\, \mathrm{proj}_{(S_V^{p-1})^{\perp}} (A^t e) \,,\, v_p \,] \leqslant \left(\frac{\sigma_{p+1}}{\sigma_p}\right) . \; tg\,(e, u_p) \qquad (5\text{-}5a)$$

$$\forall f \in S_V^N : tg\,[\, \mathrm{proj}_{(S_U^{p-1})^{\perp}} (A\, f) \,,\, u_p \,] \leqslant \left(\frac{\sigma_{p+1}}{\sigma_p}\right) . \; tg\,(f, v_p) \qquad (5\text{-}5b)$$

$$\forall e \in \left(S_U^{p-1} \right)^{\perp} : tg\left(A^t e , v_p \right) \leq \left(\frac{\sigma_{p+1}}{\sigma_p} \right) . tg\left(e, u_p \right) \qquad (5\text{-}6a)$$

$$\forall f \in \left(S_V^{p-1} \right)^{\perp} : tg\left(A f , u_p \right) \leq \left(\frac{\sigma_{p+1}}{\sigma_p} \right) . tg\left(f, v_p \right) \qquad (5\text{-}6b)$$

$$\forall \| e \in (S_U^{p-1})^{\perp} \|_2 = 1 : \frac{\sigma_p^2 - \| A^t e \|_2^2}{\sigma_p^2} = \sum_{j=p+1}^{M} g_j^2 . \left(1 - \frac{\sigma_j^2}{\sigma_p^2} \right) \qquad (5\text{-}7a)$$

$$\forall \| f \in (S_V^{p-1})^{\perp} \|_2 = 1 : \frac{\sigma_p^2 - \| A f \|_2^2}{\sigma_p^2} = \sum_{j=p+1}^{N} h_j^2 . \left(1 - \frac{\sigma_j^2}{\sigma_p^2} \right) \qquad (5\text{-}7b)$$

In which the usual symbols and expressions are used (if necessary, see chapter 1, appendix 1, and/or the following pages).

In colloquial terms:

Eq.(5-3) states that almost any vector comes closer to each of the principal singular SPACES after multiplication with A or A^t. The (nongeneric) exceptions are the vectors, strictly orthogonal to the principal singular spaces (eq.5-4)

Eqs.(5-5) and (5-6) specify these results for closeness to singular VECTORS rather than to principal singular SPACES. Notice that in eq.(5-5) the vector is FIRST multiplied with A or A^t, and THEN projected, whereas in (5-6), the vector is FIRST projected, and THEN multiplied with A or A^t.

Eq.(5-4) guarantees that a vector, orthogonal to a p-th singular space, does not leave this space after multiplication with AA^t or A^tA, i.e.: the principal singular spaces and their complements are invariant spaces with respect to AA^t and A^tA.

Eq.(5-7) finally states that the magnification of a vector, close to one of the singular vectors, converges nicely to the singular value associated to this singular vector.

B. ILLUSTRATION AND PROOF OF THEOREM (5-1).

The entire idea can be easily explained and proved by considering the vectors e, described in the left singular basis, and the vectors f, described in the right singular basis. Then, with P = Min(M,N) $\geqslant$ Rank(A), multiplication with A and A^t can be very simply described by (chapter 2):

$$\text{If} \quad e = \sum_{m=1}^{M} g_m u_m \quad \text{then} \quad A^t e = \sum_{p=1}^{P} (\sigma_p . g_p) v_p \qquad (5\text{-}8a)$$

$$\text{and if} \quad f = \sum_{n=1}^{N} h_n v_n \quad \text{then} \quad A f = \sum_{p=1}^{P} (\sigma_p . h_p) u_p \qquad (5\text{-}8b)$$

which amounts to a simple scaling of the coordinates g_p in the left, or the coordinates h_p in the right singular basis:

$e \in S_U^M \longrightarrow A^t e \in S_V^P$		$f \in S_V^N \longrightarrow Af \in S_U^P$	
g_1	$\sigma_1 g_1$	h_1	$\sigma_1 h_1$
g_2	$\sigma_2 g_2$	h_2	$\sigma_2 h_2$
.. ...	.....	.. ...	.....
g_P	$\sigma_P g_P$	h_P	$\sigma_P h_P$
g_{P+1}	0	h_{P+1}	0
....	.....	....	.....
g_M	0	h_N	0

(5-9)

Clearly, if abstraction is made of the particular basis U or V one is working in, the operations A on vector f, and A^t on vector e are COMPLETELY IDENTICAL. Hence there is no objection to describe both e and f in a sort of common basis in which the singular vectors u_p and v_p are identical for p=1 ... P, and $u_p = v_p$ by definition for p=P+1 ... Max(M,N). This is shown in Fig.(5-2), for the case M=2,N=3.

In this setting it is immediately clear that after multiplication with A^t the p-th component has grown more than the components p+1 ... Max(M,N), and hence vector e will rotate AWAY from u_{p+1} and TOWARDS u_p. This is the simple essence of Eqs (5-3),(5-5),(5-6), and is illustrated for M=N=2, σ_1=3, σ_2 =2 in Fig. (5-3).

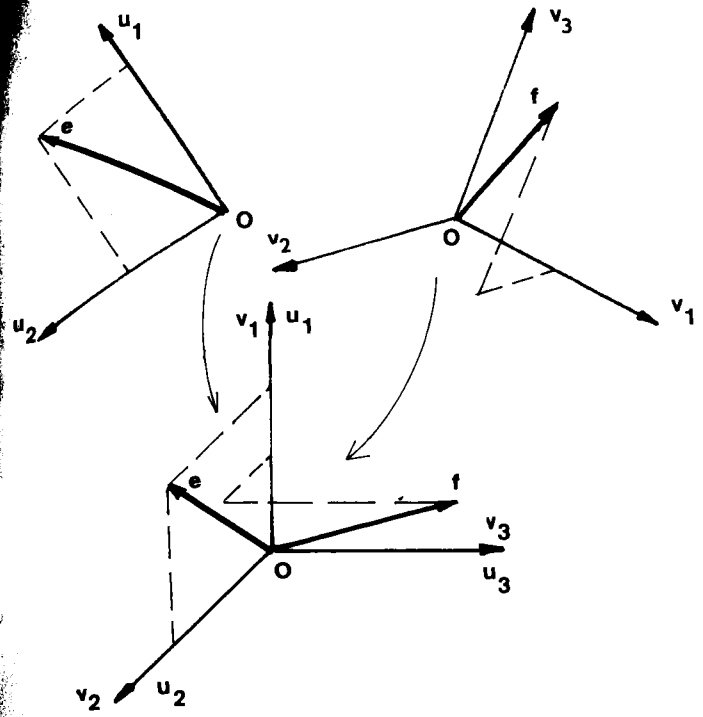


Fig.(5-2): Illustration of a common basis for U and V when M=2, N=3.

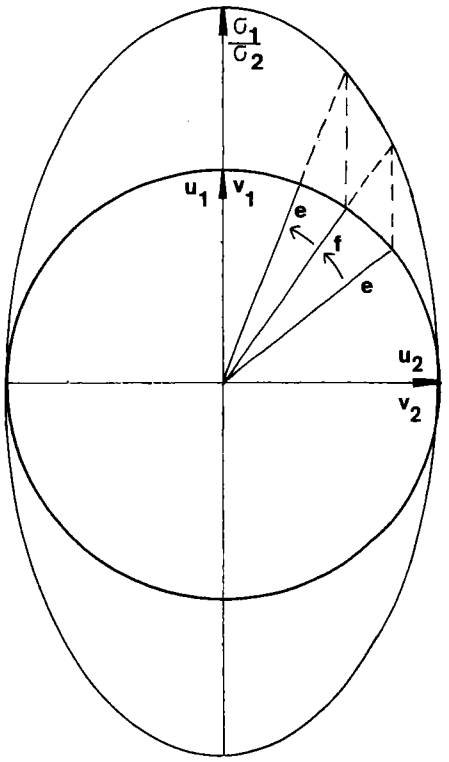


Fig.(5-3): Geometric illustration of an "$e \longrightarrow A^t e$" step in ASVD.

Once this is recognized, the proofs are nearly trivial. Eq.(5-3a), for instance, is proved as follows:

$$tg^2(A^t e, S_V^p) = \frac{\sum_{j=p+1}^{P} \sigma_j^2 . g_j^2}{\sum_{j=1}^{p} \sigma_j^2 . g_j^2} \leqslant \frac{\sigma_{p+1}^2 \sum_{j=p+1}^{M} g_j^2}{\sigma_p^2 \sum_{j=1}^{p} g_j^2} = \frac{\sigma_{p+1}^2}{\sigma_p^2} tg^2(e, S_U^p) \qquad (5\text{-}10)$$

□

The proof for (5-3b) is totally analogous. For (5-4a), one sees that:

$$\text{if } g_j = 0,\ j=1...p-1 \text{ , then also } \sigma_j . g_j = 0,\ j=1...p-1 \text{ .} \qquad (5\text{-}11)$$

□

For the Eqs. (5-5) and (5-6), in all cases, the components g_j, j=1 ... p-1 are zeroed out by the projection orthonormal to the (p-1)-th principal singular space. And so, for instance for Eq. (5-6a), one can verify that

$$tg^2(A^t e, v_p) = \frac{\sum_{j=p+1}^{P} \sigma_j^2 \cdot g_j^2}{\sigma_p^2 \cdot g_p^2} \leqslant \frac{\sigma_{p+1}^2 \sum_{j=p+1}^{M} g_j^2}{\sigma_p^2 \cdot g_p^2} = \frac{\sigma_{p+1}^2}{\sigma_p^2} tg^2(e, u_p) \qquad (5\text{-}12)$$

□

whereas for Eq.(5-5a):

$$tg^2\left(\text{Proj}_{\left(S_V^{p-1}\right)^{\perp}}(A^t e),\ v_p \right) = \frac{\sum_{j=p+1}^{P} \sigma_j^2 \cdot g_j^2}{\sigma_p^2 \cdot g_p^2}$$

$$\leqslant \frac{\sigma_{p+1}^2 \cdot \sum_{j=p+1}^{M} g_j^2 + \sigma_{p+1}^2 \cdot \sum_{j=1}^{p-1} g_j^2}{\sigma_p^2 \cdot g_p^2} = \frac{\sigma_{p+1}^2}{\sigma_p^2} tg^2(e, u_p) \qquad (5\text{-}13)$$

□

The proof of (5-7) requires some more manipulation:

$$\sigma_p^2 - \left\| A^t e \right\|_2^2 = \sigma_p^2 - \sum_{j=p}^{P} \sigma_j^2 \cdot g_j^2 = \sigma_p^2 \cdot (1 - g_p^2) - \sum_{j=p+1}^{P} \sigma_j^2 \cdot g_j^2$$

$$= \sigma_p^2 \cdot \sum_{j=p+1}^{M} g_j^2 - \sum_{j=p+1}^{M} \sigma_j^2 \cdot g_j^2$$

$$= \sum_{j=p+1}^{M} g_j^2 \cdot (\sigma_p^2 - \sigma_j^2) \qquad (5\text{-}14)$$

□

Which proves (5-7a). The proof for (5-7b) is analogous. Notice that (5-13) is valid at each instant during the convergence of the algorithm, as long as machine precision does not interfere with the computations. The modification for this expressions AT CONVERGENCE is given in the section on accuracy and convergence.

Notice also that when $\sigma_p \cong \left\| A\, e_p \right\|_2$ the convergence of the square of the SV is equivalent to the convergence of the SV itself:

$$\frac{\sigma_p - \left\| A^t e_p \right\|_2}{\sigma_p} \cong \frac{1}{2} \cdot \frac{\sigma_p^2 - \left\| A^t e_p \right\|_2^2}{\sigma_p^2} \qquad (5\text{-}15)$$

A simple but interesting estimate for the absolute accuracy of a singular value at a certain instant during convergence follows from (5-15) by the following approximation:

$$\frac{\sigma_p^2 - \left\| A^t e \right\|_2^2}{\sigma_p^2} \leq \sum_{j=p+1}^{M} g_j^2 = 1 - g_p^2 = \sin^2(e,u_p) \qquad (5\text{-}16)$$

To conclude this section, let us mention that a very nice view at the algorithm is obtained when one looks at the trajectories of the sequence of normalized vectors e,f,e,...(Fig.5-4). These trajectories are quite similar to typical linear system trajectories. This is no coincidence: indeed, if the permanent normalization of $A^t e$ and Af were not performed (and this in fact is not essential: only necessary to avoid machine overflow), then this sequence of vectors would simply behave as the state vector of an autonomous linear system with the P singular values as (independent and positive) eigenvalues, and with initial guess as initial state. The scaling only performs a continuous radial projection on a sphere (and directly after the multiplication with A or A^t on the singular ellipsoid). The projections for p>1 are necessary only to keep the vectors in the hyperplane orthogonal to the (p-1)-st principal singular space, so that the vector converges to u_p, the largest singular vector in this hyperplane, rather than to u_1, the largest singular vector in the complete space S_U.

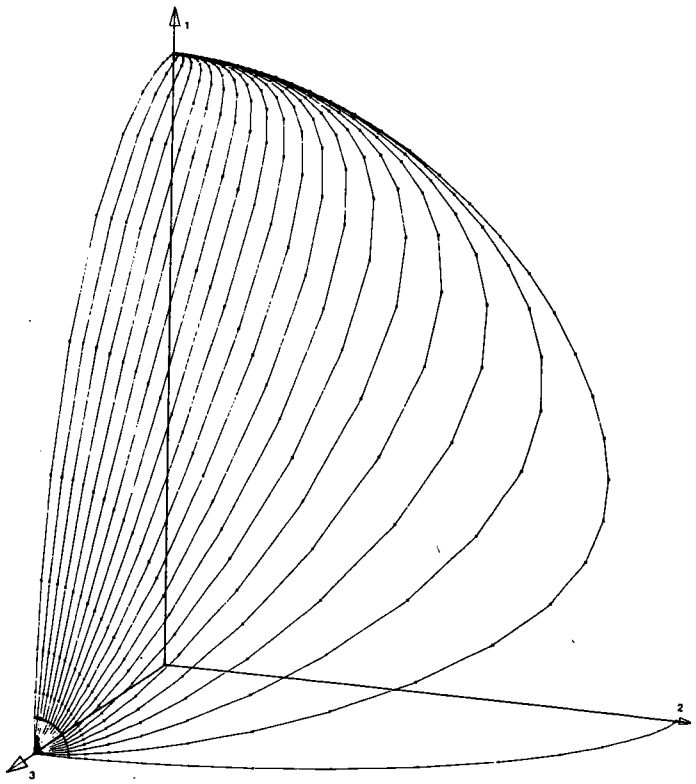


Fig.(5-4) Typical trajectories for the solution of an SVD with ASVD.

C. SOME ADDITIONAL COMMENTS.

1. From the preliminary analysis of the previous section, it is clear that the initial guess for any left p-th singular vector may be any vector that does not belong to the orthogonal complement of the p-th singular vector, which is generically the case.

* If no approximation is known to start up the iteration, a random vector can be taken, but it is appropriate to disturb this vector with a small random vector during the first iterations, to virtually annihilate the bad-luck probability of having chosen this vector in the forbidden orthogonal complement.

* If a good approximation of the p-th singular triplet is externally available, then the only computation to be performed is for converging from the approximation to the correct singular triplet. This is particularly important in the SSR identification algorithm in chapter 10.

* If no good approximation for the p-th singular triplet is externally available, then a good internal first guess is

$$e_{p-1}^{(k)} - e_{p-1}^{(k-1)} \qquad (5\text{-}17)$$

picked out at an appropriate moment k, during the convergence process of the previous triplet (p-1). This "appropriate" moment is somewhere in the middle of the convergence process, where the difference between two successive steps in the iteration is still significant (to avoid too large round-off error), but where the contribution of the singular triplets p+1... P have died out sufficiently.

2. An R-module algorithm will lead to a rank-R singular value decomposition, dropping the smallest P-R singular triplets in the decomposition.

* Notice that matrix A is USED, NOT ALTERED. So if A is highly structured (sparse, or Hankel, Toepliz, Jordan, companion...), there is no need for the storage of its complete MxN expansion. When R is small compared to P=Min(M,N), then a very high storage efficiency can be attained with ASVD. Again this is one of the key arguments in the recursive SSR identification scheme of chapter 10.

* Notice also, that if the main concern is to compute a correct GLOBAL ORIENTATION OF THE r-TH PRINCIPAL SINGULAR SPACE, (and not so much the orientation of the associated r-th principal basis within this principal space), then the only convergence rate of concern is

$$\sigma_{r+1} / \sigma_r$$

which is intimately related with the oriented signal to noise ratio for this matrix, with respect to the dropped singular triplets (see chapter 9 on realization).

Finally, let us recall the inviting characteristics of the algorithm with respect to dedicated computing structures. Though some of these aspects

have been briefly studied by the author, their analysis is far from being complete. Some of them will be mentionned occasionally, and just for the sake of (informal) information.

D.ACCURACY AND CONVERGENCE OF THE ALGORITHM.

It was proved in the first section, that ASVD is a CORRECT algorithm, i.e. that it converges to the correct solution, provided the computations are performed with infinite precision arithmetic. Moreover, it follows from the totally iterative nature of the algorithm that the accumulation of errors from one iteration loop to the other has no effect on the final result. The only sources of error that remain are:

a.Accumulation of errors within one iteration.

When the usual techniques such as double precision accumulation for scalar products are applied, the accumulation of errors within one iteration will be negligeable. Notice however that for practical application, this additional computational effort is superfluous, because the machine precision is far superior to the data accuracy. Finally one can appreciate that:

- at convergence, the set of e and f vectors become orthonormal in nature, so that the orthonormalization is quite safe.

- at convergence, singular triplets associated to small SV will suffer from errors, due to the projective and multiplicative operations. These may be relatively large (the ABSOLUTE error being approximately of the same level as the ABSOLUTE accuracy on the largest singular triplet). This is still backward stable, as the associated singular value will inversely limit the contribution to the reconstruction of the matrix A (via the dyadic decomposition).

b.Premature termination of the algorithm due to limit-cycle behaviour.

To understand this behaviour, it is useful to consider the general convergence behaviour in infinite precision, represented in Figs.5-5:

The convergence behaviour of the algorithm as a function of the error angle a_p, separating e_p and u_p, is shown in Fig.(5-5a) and (5-5b) for the singular vectors, and in Fig.(5-5c) and (5-5d) for the singular values. Clearly, the algorithm converges slowly for ratios σ_{p+1} / σ_p close to unity, and fast otherwise. Also, the iteration stops when the improvement angle $a_{p+1}-a_p$ is smaller than a certain threshold ε, i.e. when

$$\frac{g_{p+1}}{g_p} - \frac{g_{p+1} \cdot \sigma_{p+1}}{g_p \cdot \sigma_p} \leqslant \varepsilon \quad , \quad \text{or when} \qquad (5\text{-}18)$$

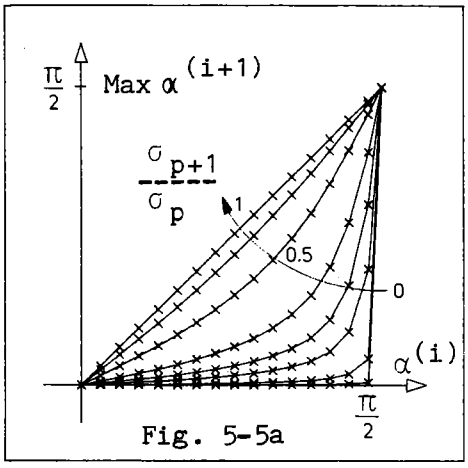


Fig. 5-5a

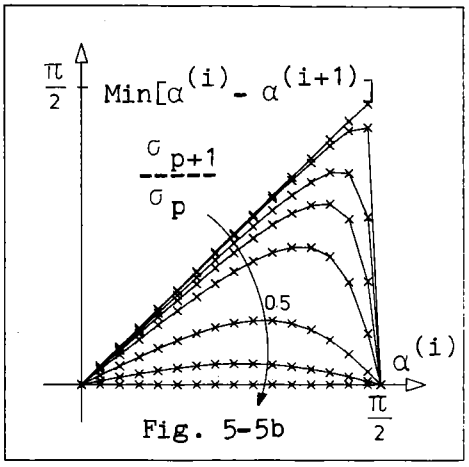


Fig. 5-5b

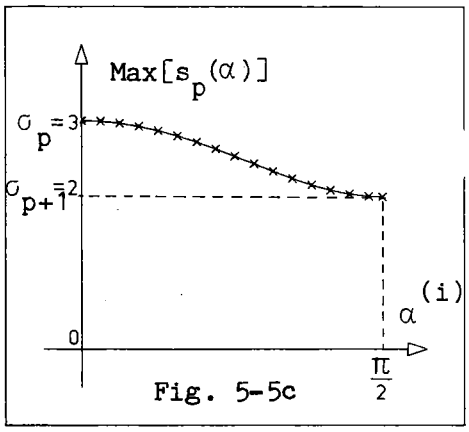


Fig. 5-5c

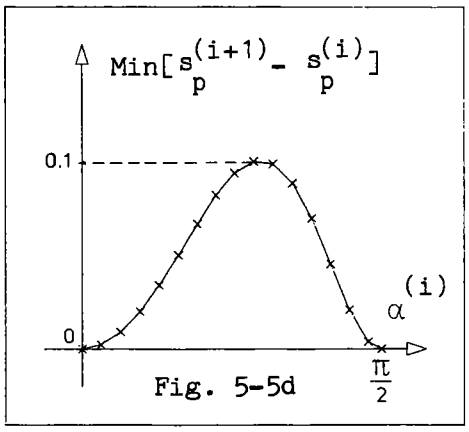


Fig. 5-5d

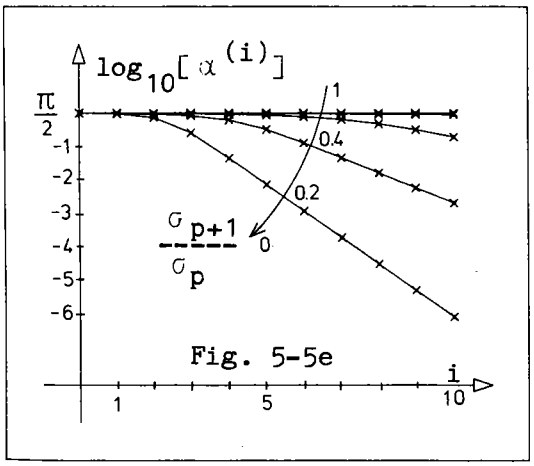


Fig. 5-5e

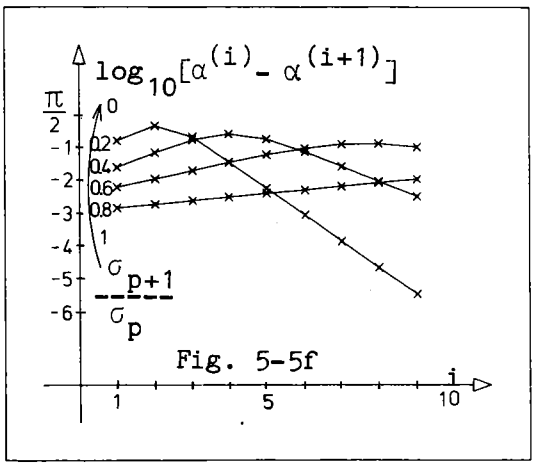


Fig. 5-5f

Fig.5-5: Aspects of the convergence behaviour of ASVD.

$$a_p \leq \frac{\sigma_p}{\sigma_p - \sigma_{p+1}} \cdot \varepsilon \qquad (5\text{-}19)$$

When two SV's are not extremely close, then the observed convergence rate can be used in eq.5-19 to determine ε from the required absolute precision a_p on the orientation of the involved singular vector.

When two SV's are very close, then ε will be very small. If ε is close to machine precision, then the expected improvement angle (i.e. the improvement angle obtained with exact arithmetic)in eq.5-18 will also be of the order of the machine precision. In that case, the accumulation of errors within the iteration procedure might be larger than ε . Further iterations will always tend to the correct solution, but the improvement angle might never converge below the ε threshold. As the accumulation of errors within one iteration is bounded, and as the number of results within a bounded neighbourhood of the correct solution is finite in finite precision, limit cycle behaviour may occur.

Eq.5-19 is an important result because it guarantees BACKWARD numerical stability for the algorithm. (Compare e.g. this result with the sensitivity property of eq.3-10 in chapter 3). Eq.(5-19) also shows that the SV will converge faster than the singular vectors, and that the convergence test should be carried out on the singular vectors rather than on the singular values. In fact, ε should be estimated from (5-19), with however a lower bound related to the accumulation of errors within one iterarion. As we did not need this refinement for our purposes in chapters 9 and 10, the exact procedure for this fail-safe termination is not yet available.

Figs. (5-5e) and (5-5f) show the convergence as a function of the number of iterations, with the concerned singular value ratio as a parameter. For small angles a, the convergence is exponential with this ratio as an exponent, which means that it is very poor when this ratio is close to 1, and the number of iterations becomes prohibitively large when one starts with an error angle very close to $\pi/2$. Considerable noise injection at the beginning of the first iteration, and a good initial guess will largely avoid the closeness to $\pi/2$. The poor convergence rate at singular ratios, close to one, require some additional comments.

First: if only a basis for the principal singular space of rank r is required, the internal convergence of their basis is irrelevant: the only relevant convergence rate is then determined by the smallest singular value among the "strong singular triplets", and the largest one among the "weak" or "virtually zero" ones.

Second: from the above, it is clear that the distance between two successive iteration steps is smaller for two close singular values than when these are well separated. When this distance becomes of the order of machine precision and/or the absolute error on previous triplets, then the converging action is overruled by the error propagation, and the above mentionned limit cycle behaviour is observed, which has been experimentally verified.

Third: the poor convergence is not unavoidable. With motivated extrapolation, it is possible to accelerate the algorithm considerably. In this context, it is worthwile to remember, that the evolution of the singular vectors behaves closely as a system of degree P. So it is an autoregressive recursion, and Wynn's so called epsilon algorithm [58] should perform very well. This aspect has not been studied yet. Instead, we give an extrapolation method that is EXACT, if the effects of the singular triplets beyond p+2 have converged out of the behaviour of the set of the triplets, with SV's σ_p and σ_{p+1}.

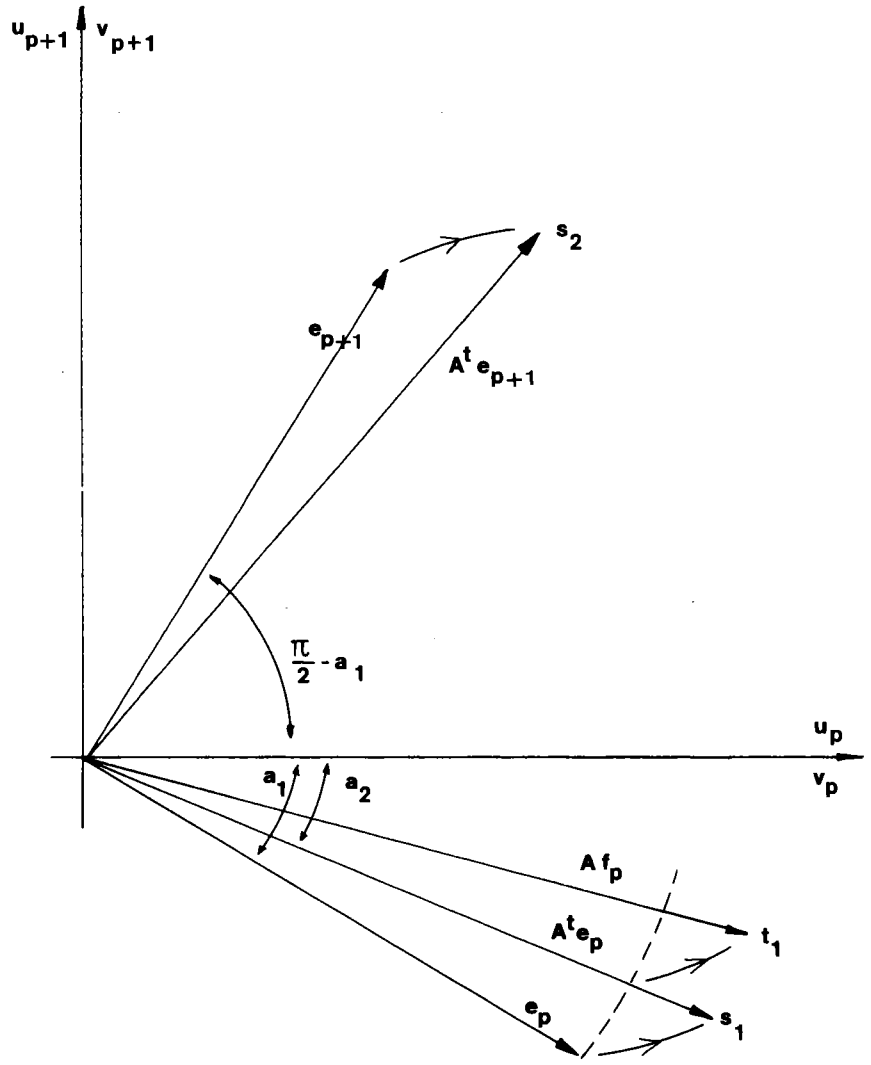


Fig. 5-6 : Vectors and angles involved in the acceleration.

E. AN ACCELERATION METHOD.

Fig.5-6 shows a two-dimensional situation, in which e_p, f_p and e_{p+1}, f_{p+1} are converging towards the singular vectors u_p, v_p resp. u_{p+1}, v_{p+1}. This is the usual situation after some iteration on two singular triplets that are close: if the next SV is sufficiently small, its effect has converged out of the behaviour of its two preceding triplets. This situation can be detected during the convergence process, during which a distinction between e_p (closest to u_p) and e_{p+1} (closest to u_{p+1}) can also be established by comparison of $A^t e_p$ and $A^t e_{p+1}$.

Now one can perform two successive iterations on e_p, one on e_{p+1}, and compute the norms of the obtained vectors. Notice that these norms can be obtained with nearly no accumulation of errors using 2 double precision accumulation scalars.

Then, according to Fig.(5-6), the following equations are valid:

$$s_1^2 \triangleq \sigma_p^2 . \cos^2(a_1) + \sigma_{p+1}^2 . \sin^2(a_1) \qquad (5\text{-}20a)$$

$$t_1^2 \triangleq \sigma_p^2 . \cos^2(a_2) + \sigma_{p+1}^2 . \sin^2(a_2) \qquad (5\text{-}20b)$$

$$s_2^2 \triangleq \sigma_p^2 . \sin^2(a_2) + \sigma_{p+1}^2 . \cos^2(a_2) \qquad (5\text{-}20c)$$

which form a set of 3 nonlinear equations in 4 unknowns. The closing equation relates the angles a_1 and a_2:

$$\mathrm{Tan}(a_2) = \frac{\sigma_{p+1}}{\sigma_p} \tan(a_1) \qquad (5\text{-}21)$$

It is a happy fact, that these equations can be "triangularised" (by simple elimination), to the following 4 explicit equations:

$$\sigma_{p+1}^2 = s_1^2 + s_2^2 - \sigma_p^2 \qquad (5\text{-}22)$$

$$\sigma_p^2 = (s_1^2 + s_2^2) \cdot \frac{\sin^2(a_1) \cdot \cos^2(a_2)}{\sin^2(a_2).\cos^2(a_1) + \sin^2(a_1).\cos^2(a_2)} \qquad (5\text{-}23)$$

$$\sin^2(a_2) = \frac{s_2^2 \sin^2(a_1) - s_1^2 \cos^2(a_1) + t_1^2 \quad [2\cos^2(a_1) - 1]}{2s_2^2 \sin^2(a_1) - 2s_1^2 \cos^2(a_1) + (s_1^2 + s_2^2).[2\cos^2(a_1) - 1]} \qquad (5\text{-}24)$$

and the equation in the only unknown a_1:

$$f = A.(B + t_1^2 A).[S \sin^2(a_1)-B] - [2B+SA].[B+SA].\sin^2(a_1) = 0 \qquad (5\text{-}25)$$

where

$$S = s_1^2 + s_2^2 \qquad (5\text{-}26)$$

$$A = 2 \cos^2(a_1) - 1 \qquad (5\text{-}27)$$

$$B = s_2^2 . \sin^2(a_1) - s_1^2 . \cos^2(a_1) = s_2^2 - S \cos^2(a_1) \qquad (5\text{-}28)$$

If e_p and e_{p+1} are ordered in such a way, that $s_1 > s_2$, then from $t_1 \geqslant s_1$, it readily follows that

$$f(a_1 = 0) = (t_1^2 - s_1^2) s_1^2 \geqslant 0 \qquad (5\text{-}27)$$

$$f(a_1 = \pi/2) = - \frac{1}{2} (s_2^2 - s_1^2)^2 \leqslant 0 \qquad (5\text{-}28)$$

which guarantees a solution between 0 and $\pi/2$. This solution can be quickly obtained with a "bisection" type algorithm. This requires $\log_2(\pi/4.\varepsilon)$ function evaluations. One such function evaluation of (5-25) can be obtained (grouping the operations) with one cosine evaluation and 9 multiplications.

Notice that the equations are EXACT under the two-dimensional assumption, and so a_1 is EXACT from their solution. A simple rotation of both vectors e_p an e_{p+1} then leads directly to complete convergence. This can be verified for possible inaccuracy [due to the solution of (5-25)] by an additional iteration. In the case of usual convergence where the σ_{p+2} effect can not be completely converged out of the triplets p and p+1, one obtains only an approximation for a_1. Usually this works well, if σ_{p+2}/σ_p is sufficiently smaller than (σ_{p+1}/σ_p). In the opposite case, more complex acceleration procedures should be set up.

F. STORAGE REQUIREMENTS AND OPERATION COUNT.

a.Storage.

As mentioned before, the storage requirements can be limited to a strict minimum. Except for a few scalars, the only storage required is that for the UNALTERED matrix A, and for the R singular triplets. If pushed to the extremes, even the sequences of e,f,e, ... vectors in process can be alternatively equivalenced with their final memory allocation in U and V. The significance of this property appears clearly from the following table (5-2) for Hankel matrices of low maximal rank R (see chapter 9 on SSR-realization for the motivation of this example):

MATRIX A		STORAGE REQUIREMENT	
Dimension M x N	Rank R	Classical SVD MN+MR+R+RN	ASVD M+N-1+MR+R+RN
16 x 16	2	322	97 (!)
32 x 32	2	1154	193 (!)
32 x 32	4	1284	323
64 x 64	2	4354	385 (!)
64 x 64	4	4612	643
64 x 64	6	4870	901
64 x 64	8	5128	1159
128 x 128	2	16898	769 (!)
128 x 128	4	17412	1283
128 x 128	6	17926	1797
128 x 128	8	18440	2311
128 x 128	10	17954	2825

TABLE 5-2: Comparison of required storage for low-rank Hankel matrices.

The advantage is more prominent, the lower the rank, and the larger the matrix. The 64x64 example of rank 4 can be quoted as usual for the algorithm AIDENT in a practical identification context (see chapter 10), and results in an improvement of a factor 8 in storage requirement.

b.Operation count.

The operation count is dependent on the particular circumstances one is working in and on the used strategy.

- Projection and normalization steps may be skipped a number of times for singular triplets that are close to the largest singular triplet.

- The convergence will significantly vary from one triplet to the other (according to σ_{p+1}/σ_p), so that the iterations may be restricted to a few of the triplets only.

- The convergence may be accelerated significantly by extrapolation.

- The number of iterations is proportional to the required number of accurate digits.

- Special techniques for specially structured matrices (for instance : FFT techniques for Hankel matrices) may significantly reduce the operation count.

- Dedicated computing units may improve the computation speed. The modular structure of the algorithms makes it particularly suitable to parallel and data flow implementation, which makes the comparison even more difficult.

- If a good external approximation of the decomposition is already available, the only required work is the convergence from the approximation to the correct decomposition.

- In normal convergence situations, a quite accurate and partially converged approximation for the next triplet can be picked out during the convergence of the previous triplet, which considerably reduces the amount of computation.

- Finally, if the computations are performed in an analog fashion, the complete SVD can be performed completely in parallel and in continuous time, the only time required being the time constants of the executing circuit. Though not very realistic for very large matrices, and not in the line of the recent tendency of digitizing the computations, it remains an interesting issue, if only for its conceptual power.

Let us finally notice that in a practical application, the main objective may not at all be the minimization of the overall computation count, nor the overall computation time, but the overall computing-unit-plus-maintenance cost, which may push the design significantly towards the most simple solution.

Due to all the above, it is not trivial to give a realistic expression of the "mean operation count". Therefore, we will consider the simplest, hence far-from-optimal strategy, for which we simply give a rough estimate. In this strategy, all operations are performed sequentially, and in each iteration cycle, in a loop over all R required singular triplets, a multiplication with A, one with A^t, and all the associated possible projections and normalizations are fully executed.

1.Multiplication.

In the case of a general MxN matrix, the matrix-vector multiplications $A^t e$ and Af take approximately 2RMN elementary "ax+b" operations per iteration. Using parallel processing, the computation SPEED can be increased significantly: each module i behaves independently (hence with R parallel processors it is possible to increase the speed with a factor R), and for each matrix vector product, each entry of the result can be computed separately [a possible factor (M+N)/2 on the average].

particular attention should be paid to sparse or highly structured matrices. For sparse matrices (with fill factor $\varphi \ll 1$), the number of operations will drop to $2.R.\varphi.M.N$ per iteration step. In case of a Hankel matrix, the matrix vector operation amounts to a convolution. According to [40, p202] this convolution can be performed most advantageously with FFT techniques, from dimensions N=32 on. This threshold will still be lower for our application as in this case the convolution is to be performed always with the same function, namely the entries of A. Finally, special purpose integrated circuits with hardware implemented FFT algorithms, perform the convolution in a negligeable amount of time.

2. Orthogonalization.

We assume the set of vectors $e_1 \ldots e_{p-1}$ orthonormal. For p=1, this assumption is superfluous; for p=2, it is trivially fulfilled. Then the simplest orthonormalization uses the Gramm-Schmidt (GS) expresssion:

$$\mathrm{Proj}_{(S_E^{p-1})^{\perp}} (e_p) = e_p - \sum_{j=1}^{p-1} (e_j^t . e_p) \cdot e_j \qquad (5\text{-}30)$$

Although in general GS orthogonalization is not numerically stable and should be replaced by the more stable Modified Gramm-Schmidt (MGS) orthogonalization, on convergence of ASVD the set of vectors $A^t e$ already form a nearly orthonormal basis, for which the GS method offers no more danger for numerically unreliable results.

For a complete orthogonalization of both the set of R A^t e-, and the set of R Af-vectors, the operation count is

$$R.(R-1).(M+N) \qquad (5\text{-}31)$$

One directly understands that this is one of the very weak points of the algorithm: if a complete SVD is required, then one complete iteration requires P.(P-1).(M+N) operations Though this also depends on the particular matrix, Golub's algorithm needs about 10.P.M.N operations for COMPLETE convergence. Therefore, in operation count on a sequential machine, the algorithm ASVD is competitive only if $R \ll P$.

Notice again that the orthonormalization of the different triplets can be performed in parallel. With R processors (one per term in the summation (5-30), the computation time per complete iteration can be reduced to

$$2.(R-1).(M+N) \qquad (5\text{-}32)$$

C. Normalization.

Normalization is not a very time consuming part of the algorithm. Per total iteration, it takes 2.(M+N).R elementary "ax+b" operations.

G.SOME REMARKS ABOUT DATA-FLOW IMPLEMENTATION.

A subject in itself - and far from being solved ! - , we are not going to give a detailed report on the data flow possibilities of the algorithm. We restrict this comment to noticing that the flow chart in itself suggests elegant data flow implementability.

CONCLUSIONS.

The main advantages of this algorithm are:

Its natural and direct link with a basic property of SVD, and hence its transparency for the potential user.

Its simplicity, the only building blocks being the matrix-vector product, orthogonalization, and normalization.

Its complete recursivity. Any earlier approximation cuts down the remaining computation cost.

Its modularity, and hence its flexibility for non-conventional implementations.

The fact that matrix A is not altered during the algorithm, and hence the storage efficiency in the case of highly structured or sparse matrices.

Its numerical reliability (backward stable algorithm).

Its main weak points are:

The algorithm is not competitive with Golubs algorithm for a single complete high precision SVD of a full rank matrix on a conventional machine.

Convergence may be VERY slow for two triplets associated to very close SVs, if appropriate acceleration rules are not executed. Such full-proof acceleration rules are not yet available for clusters of more than two close singular values.

Chapter 6

ASOLVE: A NEW ALGORITHM FOR THE ADAPTIVE LS SOLUTION OF A SET OF LINEAR EQUATIONS.

This chapter is to be read in a same state of mind as the preceding one. That is to say: the proposed algorithm has a same reliability as classical general purpose SVD based LS algorithms. But it has been designed for the special purpose of solving a slowly changing sequence of sets of linear equations with a Hankel matrix, in a recursive identification scheme (see chapter 10). For this special application, it is by far superior in time and storage efficiency, compared to the classical algorithms. Again, for the general case and on a sequential machine, no superior behaviour is (yet) claimed. On the other hand, the algorithm is flexible enough to be promising for advantageous application in other situations (sparse matrix computations !), and on other computing units (parallel processing, data flow). It is the author's belief that, if properly further developped, this algorithm will have a nice future.

A.GENERAL DESCRIPTION OF THE ALGORITHM.

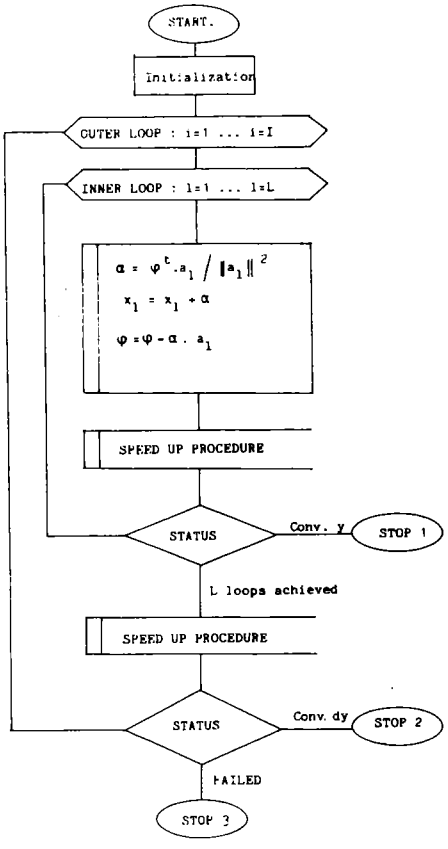


Fig. 6-1: General scheme of the algorithm ASOLVE.

Fig. 6-1 gives a general scheme of the algorithm in its most elementary form. Let us for a while forget the speed-up procedures and status evaluation for both the outer and inner loops. Then the algorithm performs IxL iterations towards the solution. One such iteration, carried out in the i-th iteration loop, and processing the l-th column of the KxL matrix A is illustrated in fig 6-2:

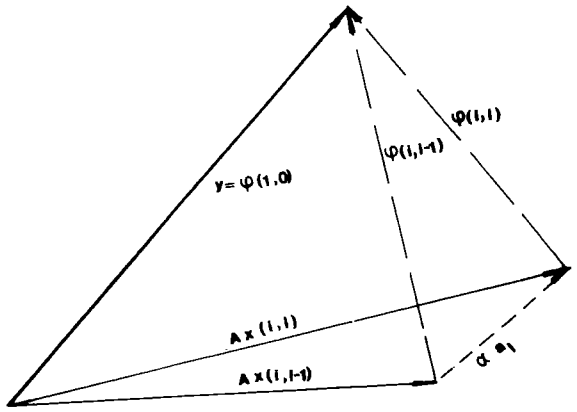


Fig. 6-2 : A geometric interpretation of iteration step (i,l)

After l-1 inner loops in the i-th outer iteration loop, an approximation x(i,l-1) of the solution x, and the remaining equation error $\varphi(i,l-1)$ are available. Now $\alpha.a_l$ is the orthogonal projection of the error vector on the l-th column of A. Consequently, a closer approximation x(i,l) of the solution x is obtained by simply adjusting the l-th component of x(i,l-1) with $+\alpha$, and the equation error $\varphi(i,l-1)$ with $-\alpha.a_l$.

The initialization of x and φ may be any good approximation x(0,0) of x, and the associated $\varphi(0,0)=y-Ax(0,0)$. If no approximation of x is available, x(0,0)=0 and $\varphi=y$ is chosen.

This (very!) simple form of the algorithm is not its most reliable nor its most efficient version. Refinements will be discussed in a later section of this chapter.

B.A FIRST EVALUATION.

For a first evaluation, we assume that no speed-up, nor status evaluation procedures are performed; that all loops are completely executed; and that vector φ is equivalenced with y (and hence y is destroyed after the computations).

a.Operation count.

A complete execution of I iterations in the elementary algorithm, including the initialization, requires roughly Z=IL(2K+2)+KL elementary "ax+b"-operations. For a square matrix, with N=L=K:

$$Z = N\,[N+2I(N+1)] \qquad (6\text{-}1)$$

The essential difference with classical algorithms (e.g. GAUSSIAN elimination or Singular Value Decomposition) is that the solution is iterative, hence for a given set of equations the factor I depends on the required accuracy rather than on the dimensions K or L of the set of equations. As a consequence, when large sets of equations have to be solved with a relatively low accuracy, one can expect that $I << K,L$ will be sufficient, and the operation count will be more favourable for the iterative algorithm. Also, if a good approximation of the solution is known, the only work left is the convergence from this approximation towards a solution with sufficient accuracy. The reader immediately feels that this may advantageously be used in recursive algorithms, as will be confirmed in Chapter 10.

b.Convergence.

The algorithm always converges to a solution of minimal equation error E, and the convergence is exponential. This important result is stated more formally in the following

Theorem 6-1 (on the convergence of ASOLVE):

For a given set of equations Ax=y, where A is a KxL matrix, the $\varphi(i,1)$ and x(i,1) of the algorithm satisfy the following LS and convergence properties for each column l=1 ... L:

$$a.\ \exists\varphi:\ \lim_{i\to\infty} \|\varphi(i,1)-\varphi\|_2 = 0 \qquad (6\text{-}2a)$$

$$b.\ \forall\, l:\ \varphi^t \cdot a_l = 0 \qquad (6\text{-}2b)$$

$$c.\ \|\varphi(i,1)-\varphi\|_2 << \|\varphi(0,1)-\varphi\|_2 \cdot \cos^i(\Omega_o / L) \qquad (6\text{-}2c)$$

where

$$\sin(\Omega_o) = \min_{\substack{\|z\|=1 \\ z \in Im(A)}} \max_{l} \frac{z^t \cdot a_l}{\|a_l\|_2} \longrightarrow 0 \qquad (6\text{-}3)$$

Proof: Trivial.

Remark: One should not be impressed too much by this proof of guaranteed exponential convergence, as the exponential

$$\cos^i(\Omega_o / L) \qquad (6\text{-}4)$$

may still be very slowly decreasing. The minimal rate of convergence, in fact, very strongly depends on the normalized condition of matrix A. (i.e. on the condition of a properly balanced A).

However, this convergence can be improved significantly with some refinement of the algorithm. Much work is still to be done in order to find an algorithm with optimal convergence (i.e. maximal convergence per elementary operation), but some preliminary results are available.

c.Nearly orthogonal matrices.

If matrix A is orthogonal, ($A^tA = AA^t = D$, where D is positive and diagonal), then the algorithm finds the solution in I=1 iteration. It can be verified that in that case, the algorithm in fact computes

$$x = D^{-1}.A^t.y \qquad (6\text{-}5)$$

which indeed only requires $O(N^2)$ operations. Sets of equations, with nearly orthogonal matrix A, will therefore very quickly converge to accurate solutions. As one easily understands, this preferential efficiency for nearly orthogonal matrices is related to the use of scalar products with UNIT weighing matrix as a metric:

$$x^t.\ y \quad = x^t.\ I\ .\ y \qquad (6\text{-}6)$$

It is not yet known wether other matrices can be exploited with a more favourable convergence per elementary operation.

d.Storage efficiency.

The basic algorithm is nearly storage optimal. Except for some scalars, the only required storage is for A, x, and y. In its most elementary form, where φ and y are equivalenced, the data in A are preserved (!), but the data in y are destroyed. So if complete data preservation is required, an additional auxiliary vector for φ is necessary.

A most important property is that the data in matrix A are NOT changed by the algorithm. This makes the algorithm particularly suitable for sparse matrix techniques (no fill-ins at all!), and structured matrix problems. The author is not aware of any other linear least squares approach to the solution of a linear set of equations, that allows to exploit matrix structure in such an efficient way.

The storage efficiency of this algorithm is one of the key ingredients of the AIDENT recursive SSR identification routine described in chapter 10.

C.REFINEMENT OF THE BASIC ALGORITHM.

A lot of refinements can improve the accuracy and speed of the algorithm:

For ACCURACY purposes, one might regularly compute [y-Ax(i,1)] as a new start value for $\varphi(i-1,1)$. In this way, one eliminates the error, accumulated during all previous updatings of $\varphi(i,1)$. If additionnally all scalar products are computed with accumulation in double precision, then we claim at least the same numerical stability as for the classical SVD based algorithm [14].

For SPEED purposes, many refinements can be thought of. As an example, consider the following idea: Store two auxiliary vectors φ'_i and x'

$$\varphi'_i = \sum_{l=1}^{L} \alpha(i,l).a_l \qquad \text{and} \qquad x'(l) = \alpha(i,l) \qquad (6\text{-}7)$$

Then after a complete iteration loop i over L columns of A, φ' can be used in the same way as any column of A, but now leading to the update $x \rightarrow x + \alpha.x'$. In this way, one can hope to globalize the effect of a complete loop over L columns into one update step, requiring O(n) additional elementary operations only. One can easily see that for K=L=2, this refinement leads to the exact solution in I...=1.5 steps (Fig. 6-3). For this reason, we propose to call this this refinement a "dim-2" refinement. The idea should be extendable to "dim-k" refinements.

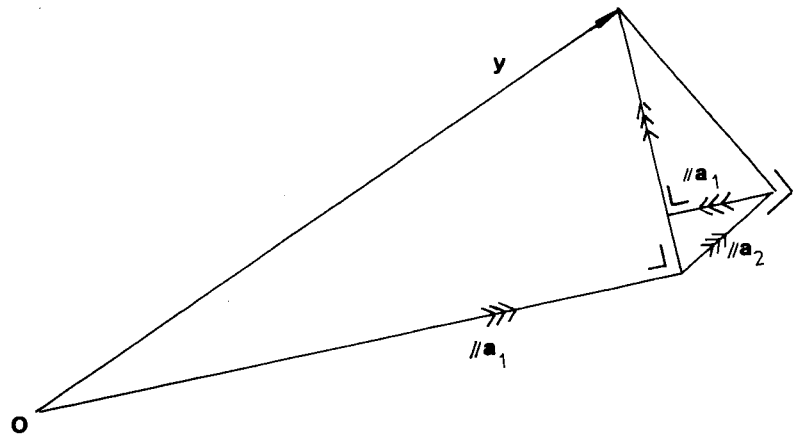


Fig. 6-3: Illustration of the dim-2 acceleration for K=L=2.

Other speed up procedures include grouping of columns, ordered selection of columns, partial iterations etc.

A concrete application of ASOLVE is found in chapter 10, where this algorithm is one of the keys to the success of AIDENT.

CONCLUSION.

The main advantages of this algorithm are:

Its simplicity, the only building blocks being the scalar product, and addition of vectors and/or scalars.

Its complete recursivity. Any earlier approximation cuts down the remaining computation cost.

The fact that matrix A is not altered during the algorithm, and hence the storage efficiency in the case of highly structured or sparse matrices.

Its numerical reliability (backward stable algorithm).

Its main weak point is:

The algorithm is not (yet) competitive with classical general purpose algorithms for a single, complete, high precision, solution of one far from orthogonal set of linear equations on a conventional machine. In this case, the convergence may be VERY slow if appropriate acceleration is not performed. The full analysis of all these effects has not been achieved yet, except in a recursive LS identification context (see chapter 10), where it has proved of excellent use.

Chapter 7

ON THE NUMERICAL IMPLICATIONS OF THE CHOICE OF A REFERENCE NODE IN THE NODAL ANALYSIS OF LARGE CIRCUITS

A.INTRODUCTION.

The existence and algebraic properties of singular matrix representations for N-terminal circuits have been described during nearly 60 years. Surprisingly, almost no attention was focussed on their numerical properties. The results presented in this chapter show that a decent circuit analysis should take into account these properties when choosing the type of circuit description to be used [110].

The subsequent statements apply to admittance, impedance or hybrid matrix representations, but to fix the ideas the restricted case of an admittance matrix representation of an N- terminal circuit will be presented [61].

$$Y^*.[v_1^* \ldots \quad v_t^* \quad \ldots v_N^*]^t = [i_1^* \ldots \quad i_s^* \quad \ldots i_N^*]^t \text{ , or } Y^*v^* = i^* \qquad (7\text{-}1a)$$

$$Y .[v_1 \ldots v_{t-1} \quad v_{t+1} \ldots v_N]^t = [i_1 \ldots i_{s-1} \quad i_{s+1} \ldots i_N]^t \text{ , or } Y\, v = i \qquad (7\text{-}1b)$$

Eq. (7-1a) is the indefinite representation, in which all terminal potentials v_j and currents i_j occur. Eq (7-1b) is the definite representation for which node voltage v_t has been chosen as a reference voltage, and the KCL (Kirchoff Current Law) equation at node s has been dropped. Usually one chooses s=t. By simply deleting row s and column t in Y*, one obtains Y. The reconstruction of Y* from Y is achieved via the so called zero row sum and zero column sum properties of Y* [61]:

$$\forall i : \sum_{j=1}^{N} y_{ij}^* = 0 \quad , \quad \text{and} \quad \forall j : \sum_{i=1}^{N} y_{ij}^* = 0 \qquad (7\text{-}2)$$

This trivial algebraic relationship can be invoked to qualify the descriptions (7-1a) and (7-1b) as equivalent. The additional equation and dependent and independent variables are called redundant. However, numerically these statements are incorrect. It turns out that even for surprisingly simple circuits, one of the matrices in (7-1a),(7-1b) may be ill conditionned, whereas the other is well conditionned. This is stated more formally in the subsequent section.

B.MAIN RESULTS.

Theorem 7-1 (on the condition of a linearly extented matrix):

For any two pairs of matrices A and A* = [A | Aa], where "a" is an arbitrary vector:

$$(1 + \| a \|_2^2)^{-1/2} \leqslant K(A) / K(A^*) \leqslant (1 + \| a \|_2^2)^{1/2} \qquad (7\text{-}3)$$

Proof:

Let σ_j, u_j and v_j (resp., σ_j^*, u_j^*, and v_j^*) denote the j-th singular value and the associated left and right singular vectors of A (resp., A*). The σ_j (resp., σ_j^*) are organized in decreasing order. From their definition, A and A* have the same row rank say n. Now observing the simple properties

$$\forall j : \quad \| u_j^t A \|_2 = \| u_j^t U \Sigma V \|_2 = \| \sigma_j . v_j^t \|_2 = \sigma_j \qquad (7\text{-}4a)$$

$$\forall u = \frac{x}{\|x\|_2} : \quad \| u_1^t A \|_2 \geqslant \| u^t A \|_2 \geqslant \| u_n^t A \|_2 \qquad (7\text{-}4b)$$

and

$$\forall x : \quad \| x^t A^* \|_2^2 = \| x^t A \|_2^2 + \| x^t A a \|_2^2$$

$$\leqslant \| x^t A \|_2^2 . (1 + \| a \|_2^2) \qquad (7\text{-}4c)$$

and combining this with the fact that a vector becomes smaller by deleting a component, we can derive four inequalities:

$$\sigma_1^* = \| u_1^{*t} A^* \|_2 \geqslant \| u_1^t A^* \|_2 \geqslant \| u_1^t A \|_2 = \sigma_1 \qquad (7\text{-}5a)$$

$$\sigma_n = \| u_n^t A \|_2 \leqslant \| u_n^{*t} A \|_2 \leqslant \| u_n^{*t} A^* \|_2 = \sigma_n^* \qquad (7\text{-}5b)$$

$$\sigma_1^{*2} = \| u_1^{*t} A^* \|_2^2 \leqslant \| u_1^{*t} A \|_2^2 . q \leqslant \| u_1^t A \|_2^2 . q = \sigma_1^2 q \qquad (7\text{-}5c)$$

$$\sigma_n^{*2} = \| u_n^{*t} A^* \|_2^2 \leqslant \| u_n^t A^* \|_2^2 \leqslant \| u_n^t A \|_2^2 . q = \sigma_n^2 q \qquad (7\text{-}5d)$$

where q denotes $(1 + \| a \|_2^2)$. The theorem then follows from (7-5) and

$$K(A) / K(A^*) = \sigma_1 \sigma_n^* / \sigma_1^* \sigma_n \qquad (7\text{-}6)$$

□

Observe that an analogous theorem and proof can be given for the addition or deletion of a linearly dependent ROW. By applying the theorem to add a column with $a=[-1,-1,\ldots-1]^t$ and afterwards to add a row with $a=[-1,-1,\ldots-1]$ to the matrix Y, one obtains a useful property:

Corollary 7-1:

The definite and indefinite admittance matrices Y and Y* satisfy:

$$N^{-1} \leq K(Y) / K(Y^*) \leq N \qquad (7\text{-}7)$$

□

The main theorem and its corollary only generate bounds on the variation of the condition number by adding or deleting a dependent row or column. These bounds (7-3) and (7-5) are however tight as the examples in the next section show.

C.EXAMPLES.

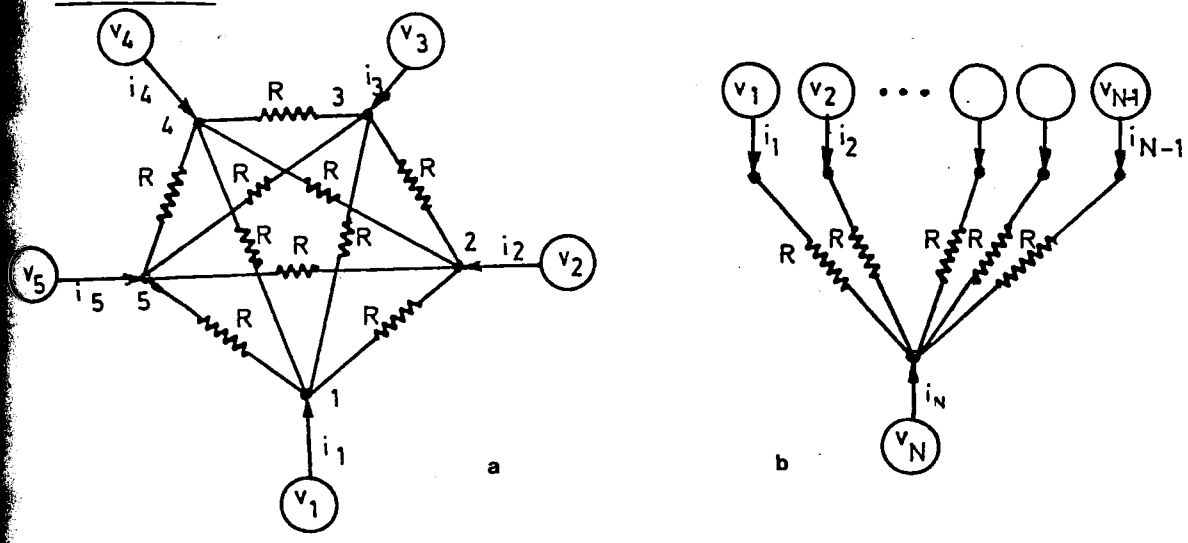


Fig. 7-1: Resistive circuits as extreme examples for corollary 7-1.

Example 7-1:

Fig. (7-1a) shows a simple resistive network with a complete graph, with N nodes (N>2). All resistances are equal to R. Then the indefinite admittance description is given by:

$$\frac{1}{R}\begin{bmatrix} N-1 & -1 & \ldots & -1 & -1 \\ -1 & N-1 & \ldots & -1 & -1 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & N-1 & -1 \\ -1 & -1 & \ldots & -1 & N-1 \end{bmatrix} \begin{bmatrix} v_1^* \\ \ldots \\ \ldots \\ \ldots \\ v_N^* \end{bmatrix} = \begin{bmatrix} i_1^* \\ \ldots \\ \ldots \\ \ldots \\ i_N^* \end{bmatrix} \qquad (7\text{-}8)$$

The condition number of Y* in(7-8) is one. Hence the description is perfectly well conditionned. Now in order to obtain a definite description, one might choose v_k as a reference voltage, and delete the associated k-th equation. The matrix Y which is the matrix Y* of (7-8) with ANY row k and column k deleted, has a condition number N. Hence the upperbound of the corollary is reached, which can be concretely experienced as follows (R=1 for simplicity):

Definite description: for an exact right hand side

$$i = \begin{bmatrix} 1 \\ -1 \\ 0 \\ \dots \\ 0 \end{bmatrix}_{N-1,1} \qquad \text{the solution is} \qquad v = \frac{1}{N} \begin{bmatrix} 1 \\ -1 \\ 0 \\ \dots \\ 0 \end{bmatrix}_{N-1,1} \tag{7-9a}$$

whereas an error

$$di = \begin{bmatrix} 1 \\ 1 \\ \dots \\ 1 \\ 1 \end{bmatrix}_{N-1,1} \qquad \text{generates an error} \qquad dv = \begin{bmatrix} 1 \\ 1 \\ \dots \\ 1 \\ 1 \end{bmatrix}_{N-1,1} \tag{7-9b}$$

from which it is clear that for this particular situation the relative error has increased with a factor K(Y) = N.

Indefinite description: the exact portion i* of the RHS satisfies

$$\sum_{k=1}^{N} i_k^* = 0 \tag{7-10}$$

Hence i* is orthogonal to $[1\ 1\ \dots\ 1]^t$ and the corresponding v* is N times smaller. Now any error di* can be decomposed as di'*+ di"*, where

$$di'^* \perp \begin{bmatrix} 1 \\ 1 \\ \dots \\ 1 \\ 1 \end{bmatrix}_{N,1} \qquad \text{and} \qquad di''^* \parallel \begin{bmatrix} 1 \\ 1 \\ \dots \\ 1 \\ 1 \end{bmatrix}_{N,1} \tag{7-11}$$

The corresponding vector dv* will then decompose as dv* = dv'* + dv"*.

dv'* will also be N times smaller than di'*, but dv"* is zero because the approximate solution is computed with generalized inverses. Hence

$$\frac{\|dv^*\|_2}{\|v^*\|_2} \leqslant \frac{\|di^*\|_2}{\|i^*\|_2} \tag{7-12}$$

and the upperbound (equality instead of inequality) is reached for any $di \perp [1\ 1\ \ldots\ 1]^t$. So with the indefinite description, the relative error will maximally increase with a factor $K(Y^*) = 1$.

Example 2: Fig. 7-1b shows a circuit with indefinite description:

$$\frac{1}{R}\begin{bmatrix} 1 & 0 & \ldots & 0 & -1 \\ 0 & 1 & \ldots & 0 & -1 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & 1 & -1 \\ -1 & -1 & \ldots & -1 & N-1 \end{bmatrix} \cdot \begin{bmatrix} v_1^* \\ \ldots \\ \ldots \\ \ldots \\ v_N^* \end{bmatrix} = \begin{bmatrix} i_1^* \\ \ldots \\ \ldots \\ \ldots \\ i_N^* \end{bmatrix}, \quad N>2 \qquad (7\text{-}13)$$

The condition number of Y^* in (7-13) is N. Hence in this case, the indefinite description is ill conditionned for large N. Obviously, by choosing v_N as the reference voltage, and dropping the n-th equation, the resulting definite matrix Y is perfectly well conditionned (unit matrix!). By the corollary the decrease in condition number can be at best a factor N. This example shows that the lower bound of corollary (7-1) is tight.

CONCLUSION.

From a practical point of view the above results show that certain descriptions may be better suited when the accuracy of the computations, the tolerances on the components, or the precision on the variables is pushed to extreme values. Some interesting results have been obtained with low precision arithmetic (single precision arithmetic) and the usual LU decomposition [62]. Our results constitute a further step towards a characterization of the conditioning of the different descriptions of electrical networks.

Chapter 8

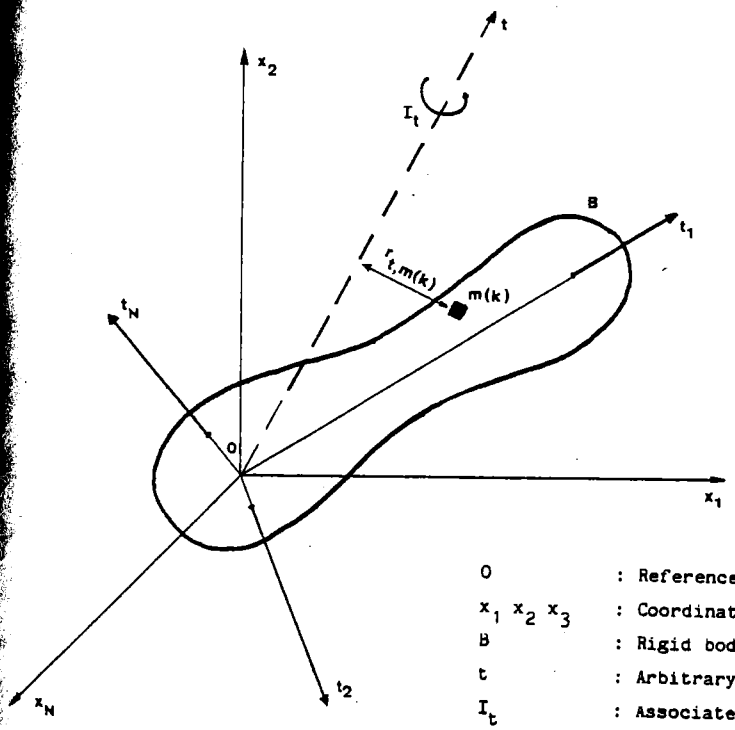
ON THE RELATIONSHIP BETWEEN SINGULAR VALUE DECOMPOSITION, AND THE PRINCIPAL AXES AND MOMENTS OF INERTIA OF A RIGID BODY

This chapter establishes the close connection between the concept of Singular Value Decomposition (SVD), and the concept of Principal Axes and Moments of Inertia (PAMI) of a rigid body. Many conceptual, algorithmic, and computational benefits can be derived from this relationship.

The two most significant achievements in this chapter are believed to be the setup of a numerically reliable algorithm for the computation of the PAMI, and the introduction of the numerical tool SVD in an intuitively appealing, mechanical PAMI-context.

A. INTRODUCTION.

The computation of the Principal Axes and Moments of Inertia (PAMI) is a problem which is considered to be solved [63]. Let us briefly refresh the subject (Fig. 8-1):



0	: Reference (Often centre of gravity)
$x_1\ x_2\ x_3$	: Coordinate system
B	: Rigid body
t	: Arbitrary axis
I_t	: Associated moment of rotational inertia.
$t_1\ t_2\ t_3$	: Principal axes of inertia with respect to 0.
$I_{t1} I_{t2} I_{t2}$	: Associated principal moments of inertia.

Fig.8-1: Notation and statement of the PAMI problem.

We will only consider the discretized case. With respect to a certain reference O (often the centre of gravity), one defines the moment of (rotational) inertia I_t around an axis t as:

$$I_t = \sum_{k=1}^{K} r_{t,m(k)}^2 \cdot m(k) \qquad (8\text{-}1)$$

I_t is a positive function of the orientation t, with one minimum I_{t1} at t_1, one maximum I_{tN} at t_N, and (for $N>2$) N-2 saddle points $I_{t2} \cdots I_{t(N-1)}$ at $t_2 \cdots t_{N-1}$. The orientations $t_1 \cdots t_N$ of extremal rotational inertia form a set of orthogonal directions in S^N and are called the principal axes (PA) of the rigid body, with respect to O. Accordingly, the associated moments of inertia are called the principal moments of intertia (MI), with respect to O. The entire structure of PA and MI is briefly indicated as PAMI.

Classically, the PAMI are obtained as the right eigenstructure of a so called inertia tensor T, i.e. the PA are the (right) eigenvectors, and the MI are the eigenvalues of T, where [63]:

$$\underline{\underline{T}} \triangleq \begin{bmatrix} t_{11} & -t_{12} & \cdots & -t_{1N} \\ -t_{21} & t_{22} & \cdots & -t_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ -t_{N1} & -t_{N2} & \cdots & t_{NN} \end{bmatrix} \qquad (8\text{-}2)$$

$$\text{with} \quad t_{ii} = \sum_{k=1}^{K} m(k) \left[\sum_{\substack{n=1 \\ n\neq i}}^{N} x_n^2(k) \right] \quad , i = 1 \ldots N \qquad (8\text{-}3a)$$

$$t_{ij} = \sum_{k=1}^{K} m(k)\, x_i(k) \cdot x_j(k) \quad , \; i,j = 1 \ldots N, \; i \neq j \qquad (8\text{-}3b)$$

Depending on the context, the eigenstructure of the inertia tensor T is really computed, or tables and composition rules are used. T is easy to obtain from the mass distribution of the considered body, described in the chosen basis. Moreover, it is a symmetric, semidefinite positive matrix, which guarantees that the computation of its eigenstructure is numerically stable [19,32]. And yet this technique requires ε^2 precision computations in order to handle properly ε-precision data. This apparently paradoxal statement is illustrated in the following simple example:

Example 8-1 (Fig. 8-2):

A rigid body with two point masses m(1)=1 at $[x_1(1),x_2(1)] = [1/\sqrt{2}\,, 1/\sqrt{2}\,]$ and m(2)=1 at $[x_1(2),x_2(2)] = [\mu/\sqrt{2}\,, -\mu/\sqrt{2}\,]$ is considered. The associated

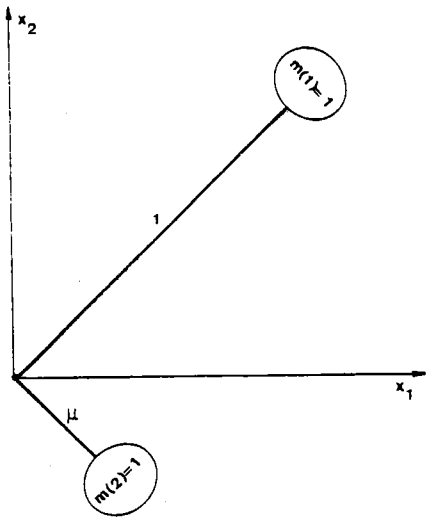


Fig. 8-2 : An example of a numerically critical PAMI problem.

inertia tensor T and its eigenstructure are [63]:

$$T = \sum_{k=1}^{2} m(k) \begin{bmatrix} x_2^2(k) & -x_1(k) \cdot x_2(k) \\ -x_2(k) \cdot x_1(k) & x_1^2(k) \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1+\mu^2 & -1+\mu^2 \\ -1+\mu^2 & 1+\mu^2 \end{bmatrix}$$

$$= \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & \mu^2 \end{bmatrix} \cdot \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$$

Hence the PAMI are :

Principal axes	$\begin{bmatrix} 1 \\ -1 \end{bmatrix}$	and	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$	(8-4a)
Associated Moments	1	and	μ^2	(8-4b)

Now let us see how this classical approach behaves numerically, i.e. when only finite precision computation is possible and/or only finite precision measurements are available.

First consider the case with exact measurements, but finite precision computations. The intertia tensor T is then represented by a matrix in which m(2) does not even appear, if the machine imprecision ε is larger than μ^2:

$$T = \begin{bmatrix} 1+\mu^2 & -1+\mu^2 \\ -1+\mu^2 & 1+\mu^2 \end{bmatrix} \underset{\varepsilon > \mu^2}{=} \begin{bmatrix} 1+\varepsilon_{11} & -1+\varepsilon_{12} \\ -1+\varepsilon_{21} & 1+\varepsilon_{22} \end{bmatrix} \qquad (8\text{-}5)$$

which means that one will find the PAMI struture of a body, in which the mass m(2) is not present anymore. To give two examples: unit masses, orthogonal to, and at a relative distance of 0.0001 of the strongest prin-

cipal axis, will disappear in 7-digit precision computations; and 8 bit precision computations will account for effects within a 4-bit range only.

Second consider inaccuracies η_i on the measurements (x,y) with an absolute level η (i.e. all measurements are performed with the same absolute precision). From an inspection of Fig. 8-2, it is clear that with infinite precision computations, a good algorithm should be able to yield reasonably accurate values for the PAMI when the measurements satisfy $1 \gg \mu \gg \eta \gg \mu^2$. But such measurement errors η_i will affect the inertia tensor T more than the effect μ^2 of mass m(2):

$$T = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \begin{bmatrix} \mu^2 & \mu^2 \\ \mu^2 & \mu^2 \end{bmatrix} + \begin{bmatrix} 2.\eta_2 & -\eta_1-\eta_2 \\ -\eta_1-\eta_2 & 2.\eta_1 \end{bmatrix}$$

$$\underset{O(\eta \gg \mu^2)}{=} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \begin{bmatrix} 2.\eta_2 & -\eta_1-\eta_2 \\ -\eta_1-\eta_2 & 2.\eta_1 \end{bmatrix} \qquad (8\text{-}6)$$

In the general case, both sources of inaccuracies ε_i and η_i will combine, and affect the results of the computations in a severe way. The observed degeneration of precision is NOT ESSENTIAL TO THE PAMI PROBLEM ITSELF, but to its FORMULATION AS AN EIGENSTRUCTURE PROBLEM OF T. Notice especially that the difficulties are NOT due to the eigenstructure decomposition of T, which is itself numerically stable because T is symmetric and positive semidefinite. Rather, the loss of accuracy is caused by the mere fact of USING tensor T in the computations: in this tensor, the data associated to different main directions in the space are not kept separated in a proper way. As these data appear in a squared form, ε - or η-errors on large contributions in one direction will mix up with small contributions of μ^2-level in another direction, thus leading to the observed fatal loss of accuracy. We will soon show that the PAMI problem can be solved in a numerically reliable way, if appropriate numerical methods are used for the computations: In the next section, we describe a procedure that guarantees $\varepsilon+\eta$ accurate results with ε precision computations on η accurate data. Hereto we establish a link with the Singular Value Decomposition (SVD).
The close connection of the SVD and PAMI concepts are believed to be important new element as well for the numerical analysis of mechanical problems, as for the conceptual understanding of numerical analysis methods.
The organization of the chapter is as follows: first, we establish the relationship between the numerical concept of Singular Value Decomposition (SVD), and the mechanical concept of Principal Moments and Axes of Inertia (PAMI) of a rigid body. From the established relationship, we will derive an intuitively appealing PAMI interpretation for SVD, and we will derive a reliable SVD based algorithm for the computation of PAMI. Finally, we will focus attention on some aspects of computational implementation.

A practical conclusion will be that the formulation of the PAMI problem as an SVD type problem guarantees numerically reliable results, that can be obtained with slight modifications of the present tables and/or computer-

programs, and with nearly the same computational effort as necessary for the eigenstructure decomposition of the inertia tensor.

B. THE BASIC THEOREM AND SOME EXAMPLES.

Theorem 8-1 (On the relation between SVD and PAMI):

Consider a rigid body B, composed of K discrete point masses m(k), located at $x(k) = [\, x_1(k), \ldots x_N(k)]^t$ in an N-dimensional coordinate system with reference O (Usually O is the center of gravity of B, and N=2 or N=3). For such a rigid body, we construct a matrix M_B as follows:

$$M_B = \begin{bmatrix} \sqrt{m(1)} \cdot x_1(1) & \sqrt{m(2)} \cdot x_1(2) & \ldots & \sqrt{m(K)} \cdot x_1(K) \\ \ldots & \ldots & \ldots & \ldots \\ \sqrt{m(1)} \cdot x_N(1) & \sqrt{m(2)} \cdot x_N(2) & \ldots & \sqrt{m(K)} \cdot x_N(K) \end{bmatrix} \qquad (8\text{-}7a)$$

then a rigid body B' with identical PAMI can be constructed with N unit pointmasses, located at

$$\sigma_i \cdot u_i \qquad , i=1 \ldots N \qquad (8\text{-}7b)$$

Where σ_i and u_i are the i-th singular value and i-th left singular vector in the Singular Value Decomposition $M_B = U \Sigma V$ of M_B. We will call this unit-mass structure with identical PAMI properties with respect to O, the O-equivalent body B' of B.

□

From theorem 8-1, one readily derives the following corollaries:

Corollary 8-1 (On the direct computation of the PAMI):

1. The principal axes of inertia $t_1 \ldots t_2$ of a rigid body B are the left singular vectors $u_1 \ldots u_N$ of the associated M_B-matrix.

2. The principal moments of inertia $I_{t1} \ldots I_{tN}$ are obtained from the SVD of M_B as:

$$I_{ti} = \sum_{\substack{n=1 \\ n \neq i}}^{N} \sigma_n^2 \qquad (8\text{-}8)$$

Corollary 8-2 (On composite objects):

If a body B is composed of two subbodies B_1 and B_2 with B-equivalent structures $U_1.\Sigma_1$ resp., $U_2.\Sigma_2$, then an M-matrix for B is obtained as

$$M_B = [\ U_1.\Sigma_1 \quad U_2.\Sigma_2\ ] \tag{8-9}$$

and its associated equivalent structure $U.\Sigma$ from its SVD $M_B = U\Sigma V$.

Proofs:

It is easily verified that the M_B and T matrices of any rigid body B are related by

$$M_B.M_B^t = \|M_B\|_F^2 \cdot I_{NN} - T \tag{8-10}$$

By construction, $M_{B'} = U_B\Sigma_B$ is the M-matrix of B'. But

$$M_{B'}.M_{B'}^t = M_B.M_B^t \quad , \text{ and} \tag{8-11a}$$

$$\|M_{B'}\|_F = \|M_B\|_F \tag{8-11b}$$

So B and B' have exactly the same inertia tensor T, hence identical PAMI, which proves the theorem.

Corollary 8-1 follows readily from theorem 8-1 and the orthogonal structure of the O-equivalent structure. Corollary 8-2 follows from the construction of the M-matrix, together with the fact that

$$M_{B1'} = U_{B1}.\Sigma_{B1} \quad \text{and} \quad M_{B2'} = U_{B2}.\Sigma_{B2}$$

are the M-matrices of the equivalent structures B1' and B2'.

Example 8-2:

To illustrate the above, consider the following discrete rigid body, composed of 8 point masses in a 3 dimensional space (Fig 8-3a, N=3, K=8):

k	1	2	3	4	5	6	7	8
m (k)	1	4	4	9	1	1	1	9
$x_1(k)$	0	1	3	3	4	2	4	1
$x_2(k)$	4	2	2	0	2	2	1	3
$x_3(k)$	1	3	4	2	0	1	2	0

(8-12)

Then one can directly compute the PAMI from:

$$M_B \triangleq \begin{bmatrix} 0 & 2 & 6 & 9 & 4 & 2 & 4 & 3 \\ 4 & 4 & 4 & 0 & 2 & 2 & 1 & 9 \\ 1 & 6 & 8 & 6 & 0 & 1 & 2 & 0 \end{bmatrix} \qquad (8\text{-}13)$$

$$= \begin{bmatrix} 0.687 & -0.536 & 0.490 \\ 0.519 & 0.834 & 0.186 \\ 0.509 & -0.127 & -0.852 \end{bmatrix} \cdot \begin{bmatrix} 9.9663 & 0 & 0 \\ 0 & 4.6162 & 0 \\ 0 & 0 & 3.5161 \end{bmatrix} \cdot V$$

and hence $I_{t1} = 33.67$; $I_{t2} = 111.69$; $I_{t3} = 120.64$ (8-14a)

$$t_1 = \begin{bmatrix} 0.687 \\ 0.518 \\ 0.509 \end{bmatrix} ; \quad t_2 = \begin{bmatrix} -0.536 \\ 0.834 \\ -0.127 \end{bmatrix} ; \quad t_3 = \begin{bmatrix} 0.490 \\ 0.186 \\ -0.852 \end{bmatrix} \qquad (8\text{-}14b)$$

which is represented in Fig 8-3b. The composition property is then illustrated in Figs. 8-3c to 8-3f, corresponding to the following computations:

$$M_{B1} \triangleq \begin{bmatrix} 0 & 2 & 6 & 9 & 4 \\ 4 & 4 & 4 & 0 & 2 \\ 1 & 6 & 8 & 6 & 0 \end{bmatrix} \qquad (8\text{-}15)$$

$$= \begin{bmatrix} 0.624 & 0.687 & 0.373 \\ 0.514 & -0.720 & 0.466 \\ 0.588 & -0.099 & -0.803 \end{bmatrix} \cdot \begin{bmatrix} 8.246 & 0 & 0 \\ 0 & 3.900 & 0 \\ 0 & 0 & 3.130 \end{bmatrix} \cdot V$$

$$M_{B2} \triangleq \begin{bmatrix} 2 & 4 & 3 \\ 2 & 1 & 9 \\ 1 & 2 & 0 \end{bmatrix} \qquad (8\text{-}16)$$

$$= \begin{bmatrix} 0.776 & -0.408 & 0.482 \\ 0.524 & 0.841 & -0.132 \\ 0.352 & -0.355 & -0.866 \end{bmatrix} \cdot \begin{bmatrix} 5.7421 & 0 & 0 \\ 0 & 2.6437 & 0 \\ 0 & 0 & 0.1976 \end{bmatrix} \cdot V$$

and hence

$$[M_{B1}\ M_{B2}] \triangleq \begin{bmatrix} 0.624 & 0.687 & 0.373 & 0.776 & -0.408 & 0.482 \\ 0.514 & -0.720 & 0.466 & 0.524 & 0.841 & -0.132 \\ 0.588 & -0.099 & -0.803 & 0.352 & -0.355 & -0.866 \end{bmatrix} \qquad (8\text{-}17)$$

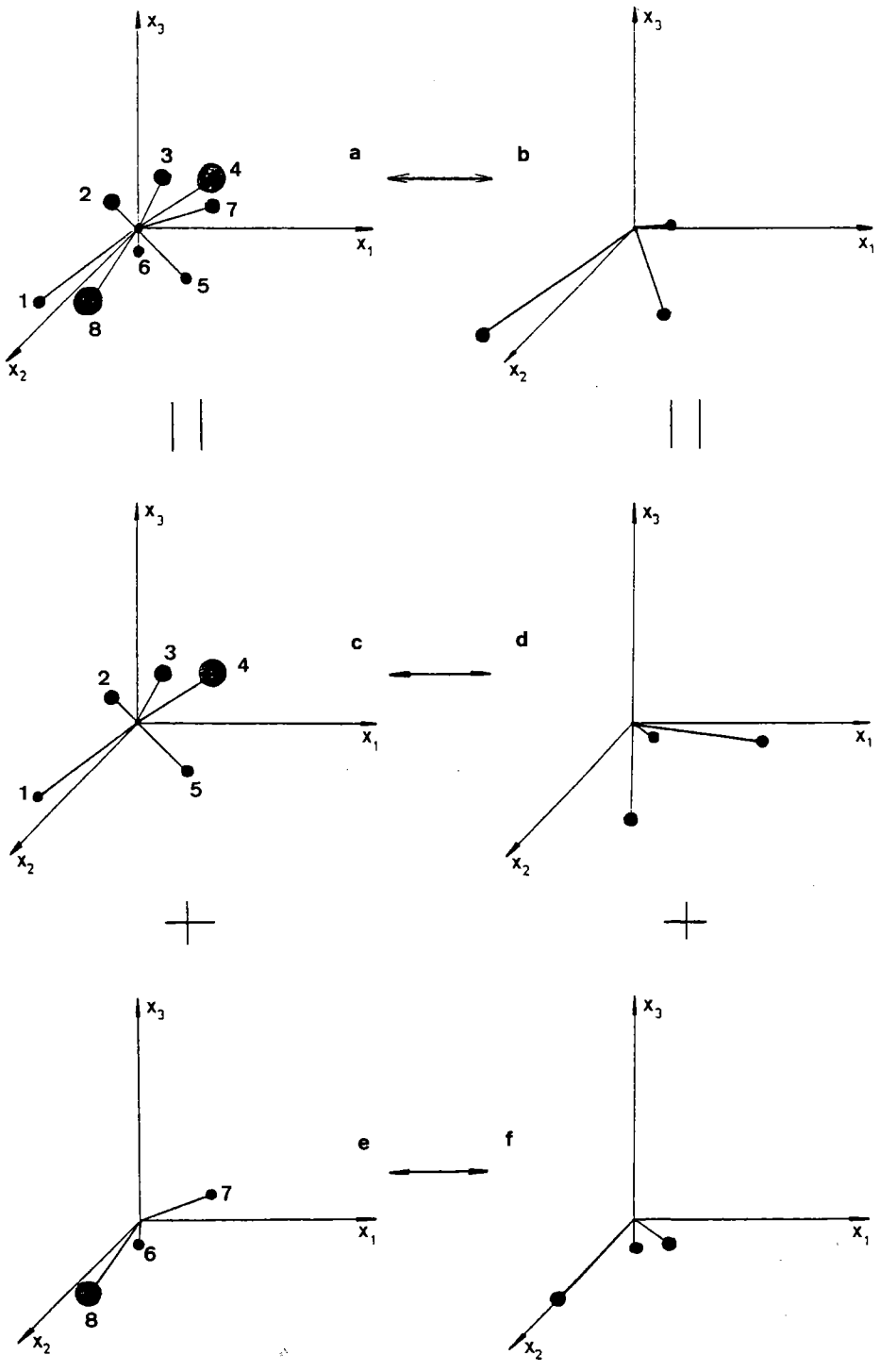


Fig 8-3 : An illustrative example for theorem 8-1 and its 2 corollaries.

$$= \begin{bmatrix} 0.687 & -0.536 & 0.490 \\ 0.519 & 0.834 & 0.186 \\ 0.509 & -0.127 & -0.852 \end{bmatrix} \cdot \begin{bmatrix} 9.966 & 0 & 0 \\ 0 & 4.616 & 0 \\ 0 & 0 & 3.516 \end{bmatrix} \cdot V$$

which yields indeed the same PAMI structure as the direct computation.

Important remarks.

1. In the light of chapters 2 and 3 remember that the equivalence between the eigenstructure of $M_B M_B^t$ and the singular structure of M_B is useful for conceptual purposes (here for proving the theorem), but is of no further use for reliable computational purposes: if M_B is itself available, one should NOT compute the eigenstructure of $M_B M_B^t$ (this would be totally equivalent to computing the eigenstructure of T) but one computes the SVD of M_B directly! This is the essential difference between the classical PAMI definition, and the one that follows from our theorem 8-1. For the example (8-1) of the introduction, one should compute the SVD

$$M_B = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & \mu \\ 1 & -\mu \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & \mu \end{bmatrix} \cdot V \qquad (8\text{-}18)$$

in which the effect μ appears clearly in the singular value $\sigma_2 = \mu$ that can be obtained with a precision ε / μ.

2. Clearly, theorem 8-1 can be used, at least in principle, for any mechanical system, provided it is cut in sufficiently small particles, which can be regarded as point masses within the required accuracy. In this case the M matrix will usually be a very long matrix, and for practical implementation one better partitions this matrix into smaller matrices, the results of which are then combined by an overall SVD. This was shown for a simple case in example 8-3. In many practical applications however, the considered system is a composition of subsystems with known PAMI, that may be analytically available from tables (when the subsystems have a simple geometry), or that may have been computed at a previous instant. To find the PAMI of such a composite body, one uses corollary 2.

Example 8-3 (Fig. 8-4)

The principal axes of a solid cilinder, with respect to its centre of gravity, are its axes of symmetry. The principal moments are given in eq. 8-19 together with the singular values of its O-equivalent body.

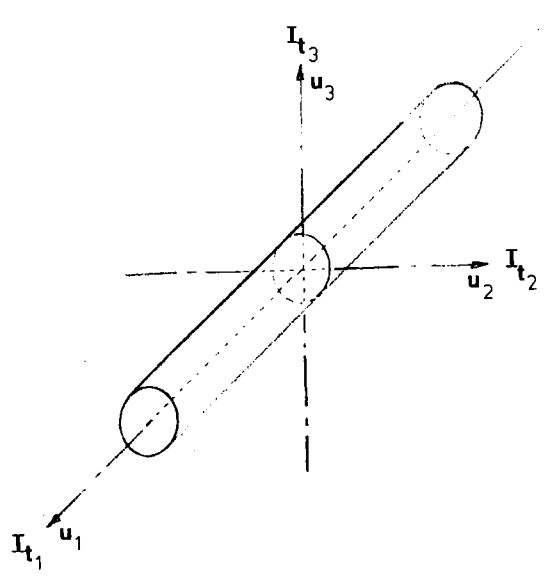


Fig. 8-4: PAMI and O-equivalent description of a solid cilinder.

$$\sigma_1 = \sqrt{\frac{ML^2}{12}} \quad , \quad t_1 = \frac{M R^2}{2} = \sigma_2^2 + \sigma_3^2 \qquad (8\text{-}19a)$$

$$\sigma_2 = \sqrt{\frac{MR^2}{4}} \quad , \quad t_2 = \frac{M L^2}{12} + \frac{M R^2}{4} = \sigma_1^2 + \sigma_3^2 \qquad (8\text{-}19b)$$

$$\sigma_3 = \sqrt{\frac{MR^2}{4}} \quad , \quad t_3 = \frac{M L^2}{12} + \frac{M R^2}{4} = \sigma_1^2 + \sigma_2^2 \qquad (8\text{-}19c)$$

The effect of adding a small mass-m notch at place v on the cilinder can now be simply accounted for by computing the SVD of the composite matrix

$$M_B = \left[\sqrt{\frac{ML^2}{12}}\ u_1 \ , \sqrt{\frac{MR^2}{4}}\ u_2 \ , \sqrt{\frac{MR^2}{4}}\ u_3 \ , \sqrt{m}\ v \right] \qquad (8\text{-}20)$$

Notice again the nice separation, in the SVD description, of effects due to the cilinder thickness, from those due to its length. In the direct PAMI description, these data are squared, and all mixed up.

Notice also that when the O-equivalent description of the structure is not directly available, it could be obtained from the moments of inertia as

$$\sigma_i^2 = \left[\sum_{j=1}^{N} I_{t_j} - (N-1).I_{t_i} \right] / (N-1) \qquad (8\text{-}21)$$

but this substraction is numerically dangerous. Therefore, if the data on

B_1 are available from previous computations, it is preferable to store the singular values σ_i rather than the principal moments I_{ti}. In fact, for intermediate results, it is ALWAYS numerically better to store the σ_i rather than the I_{ti}. Hence, it would be more appropriate to set up tables with O-equivalent data, rather than the classical tables with moments of inertia.

As is clear from the above example, it requires only a minor effort to reconvert the tables available today, but it makes the composition of structures easier, and what is more: it makes the whole difference between unstable and stable numerical computation.

3. The reader may wonder whether the (opposite) operation of DELETING a subbody can be performed as well. It appears, that if $M_B = [M_{B1}\ M_{B2}]$ is not known explicitely, it is impossible to delete a subsystem B_2 from a global subsystem B, without using the product $M_B.M_B^t$. This is numerically unstable. So adding a subsystem allows a stable computation, deleting does not.

C. COMPUTATIONAL ASPECTS.

Except for some common sense organization, the SVD based algorithm doesn't require much more than a reliable and efficient SVD module, and its computational complexity compares reasonably well with the more simple classical algorithms. In fact, if computation time is not really a critical factor, there is no reason for not using an SVD based algorithm.

The SVD based method requires roughly K square roots, plus an SVD. This compares reasonably with the computational complexity of the classical algorithm, in which the computation of the inertia tensor and its eigenstructure decomposition require roughly [32]

$$3\ N.(N-1).K + (3+2s)\ N^3 \qquad (8\text{-}22)$$

elementary "ax+b"-operations (Here s is the number of iterations in the eigenstructure decomposition).

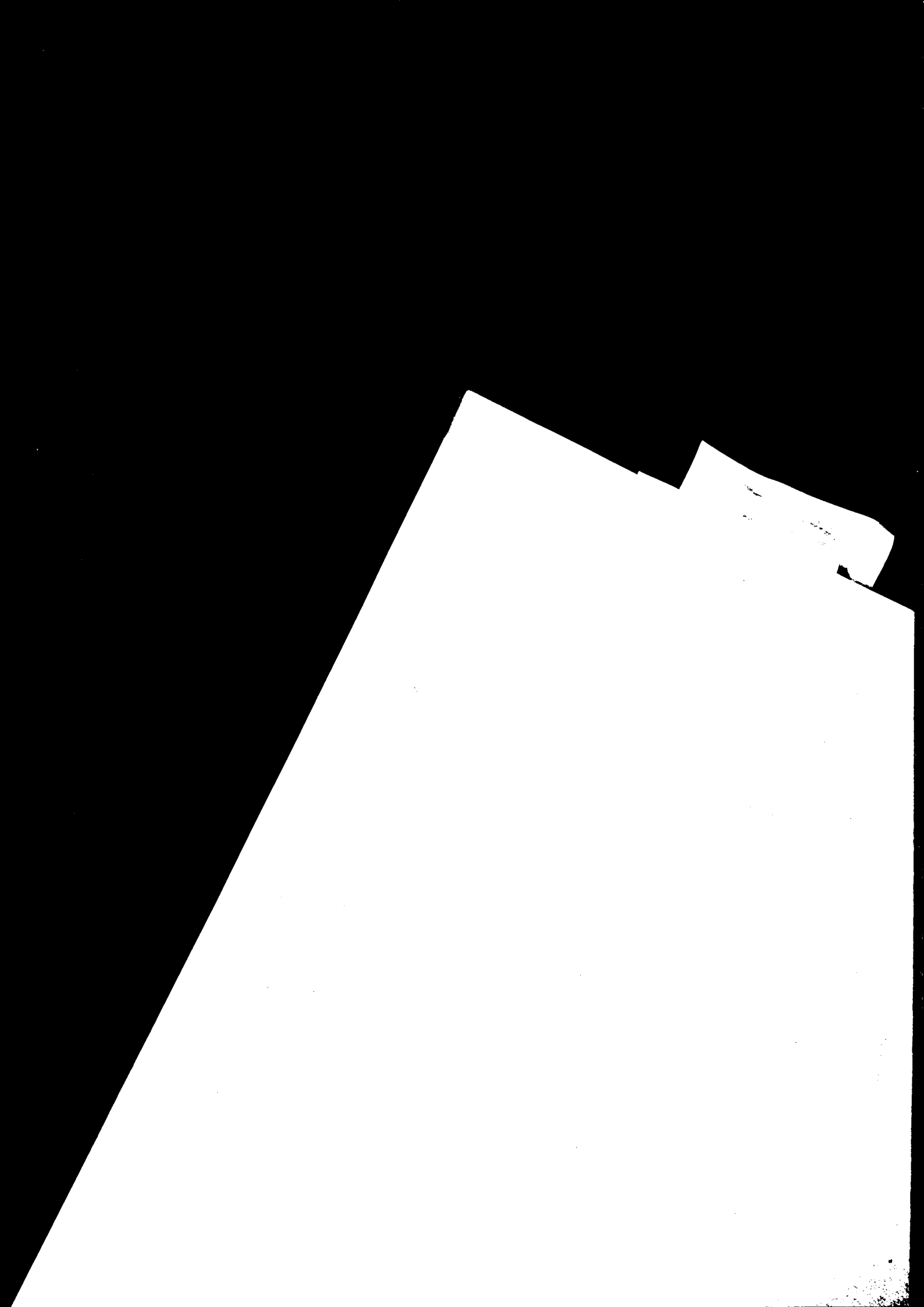
In fact, it is not so simple to compare rigorously both operation counts because of the N square roots, the rather variable number of iterations in both eigenstructure and SVD algorithms, and possible shortcuts that can be followed using the specific structure properties of the M or T matrices. This explains the term "roughly" in the above evaluation. But it is also clear from the above, that for large K the number of elementary operations for both algorithms is proportional to KN^2. So when square roots do not consume all the computation time, there is no reason not to apply the so much more reliable SVD algorithm.

If a very large number of mass particles have to be accounted for, they can be grouped into subbodies of KI particles, that can each be replaced by three singular vectors. In this way, the subbodies can be computed

separately, and later combined (parallel processing); or they can be built up gradually, and the result can be permanently combined with the latest "compressed" subbody (sequential processing). Both methods are numerically stable, if the partial results are stored under their singular form, and if the computation of the singular structure into PAMI is performed on the final result only. Notice finally, that special purpose routines exist, that are specially designed for sequential SVD processing of very long matrices [14].

D.CONCLUSION.

In this chapter we have shown that it is possible to introduce the numerical tool SVD in a very mechanically-minded way. We have shown its conceptual relationship with the notions of Principal Axes and Moments of Inertia of a rigid body, and we have discussed the benefits of recognizing this relationship. We have illustrated the claim of numerical superiority of the SVD approach over the inertia tensor approach with simple examples. The reliability of computation can be improved significantly when the computations are performed directly as well as when tables and composition rules are used. Finally, we have discussed the computational complexity of the proposed method. The general conclusion is, that if precision and reliability are important, and if the computation time is not really critical, it is a good choice to use the SVD based approach.



Chapter 9

ON NUMERICALLY RELIABLE REALIZATION

INTRODUCTION.

The realization of an Impulse Response (IR) sequence into a State Space Representation (SSR) is a standard topic in linear system theory. Ho and Kalman (1965) were the first to state the problem in a Markov Block Hankel (MBH) formalism [26]. Their basic algorithm has been reconsidered by many authors, in order to increase speed [28,29,50,100] or reliability [30,31, 101,108]; or to gain insight in its properties and structure [27,103]. The subject is quite fundamental, the literature on the matter is very extensive, and the main ideas will be rather familiar to most readers that have worked in the domain of linear systems. Therefore this chapter does not aim at ambitious (and superfluous) completeness. Rather, it wants to point out how the geometric interpretation of Singular Value Decomposition can help to set up reliable implementation schemes, in which the decision on the degree can be based on motivations that are at the same time physically meaningful, and numerically reliable. Following [103,106,107], we state the transfer realization problem as a simple (linear) matching problem of the (matrix) operators A, B, C to a State Impulse Response (SIR), that is directly related to the given data. This problem statement generalizes the MBH statement of Staar [103] considerably, but the geometric interpretation is essentially the same. It allows to situate and evaluate known or new approaches to the solution of this problem in a general framework, and to handle the problem of approximation in a transparent manner. The here presented solution strategies include direct generalizations of the Ho-Kalman approach [26], the Silverman algorithm [27], and they also situate the solutions presented by Zeiger & Mc Ewen [30], Moore [64], Kung [31]. Moreover a strategy for improving the matching of the B-Matrix and a strategy for slowly responding systems are presented in order to illustrate the power and the versatility of the general problem formulation and its geometric interpretation. Most of the reviewed, and all the proposed new solutions are presented using SVD, which in the concrete realization context is the key to reliable implementation. It will also appear why we emphasised so much the algorithm ASVD in chapter 6: with this algorithm, the structure of MBH matrices can be completely exploited to reduce the required storage capacity, necessary for reliable realization, and this without destroying the reliability of the solution. The practical significance of the presented realization scheme will then fully emerge in chapter 10, where it will be incorporated in a recursive SSR identification procedure.

A. REALIZATION OF EXACTLY REALIZABLE TRUNCATED IMPULSE RESPONSES.

The purpose of this section is to derive a formulation for the problem of EXACT Transfer Impulse Response realization, that can lead to a meaningful extension for the problem of NOISY impulse response approximation. The definitions and notations used in this section are standard in linear system

heory (See e.g. [24] or [36]). For completeness, an abbreviated overview f the used background material is given in appendix.

a. Initial problem statement.

We start our derivation by considering three closely related types of realization problems on "exactly realizable impulse responses" [106]:

Transfer Impulse response Realization (TIR):

Given a sequence of K_1 (1xm)-matrices H(k), find a triple of matrices A_1, B_1, C_1, such that:

$$C_1 A_1^{k-1} B_1 = H(k) \quad , \quad k=1 \ldots K_1 \tag{9-1}$$

Controllable State Impulse response Realization (CSIR):

Given a sequence of K_2 (Nxm)-matrices X(k), find a couple of matrices A_2, B_2, such that:

$$A_2^{k-1} B_2 = X(k) \quad , \quad k=1 \ldots K_2 \tag{9-2}$$

Observable State Impulse response Realization (OSIR):

Given a sequence of K_3 (1xN)-matrices $\Xi(k)$, find a couple of matrices C_3, A_3, such that:

$$C_3 A_3^{k-1} = \Xi(k) \quad , \quad k=1 \ldots K_3 \tag{9-3}$$

It is standard in exact realization theory to assume that the given data H(k), X(k), and/or $\Xi(k)$ originate from some triple A,B,C, and are available without any approximation or inaccuracy. It is also standard to assume that the truncation numbers K_1, K_2, K_3 are large enough, so that the finite sequences in the eqs. (9-1)-(9-3) characterize completely the infinite sequences, associated to A, B, and C. More specifically, the truncation numbers are assumed to be large enough compared to the controllability-, and observability indices of the considered class of systems [106]:

$$K_1 \geq i_c + i_o \tag{9-4a}$$

$$K_2 \geq i_c + 1 \tag{9-4b}$$

$$K_3 \geq 1 + i_o \tag{9-4c}$$

Under these assumptions, the TIR, CSIR and OSIR problems have at least one exact solution, that satisfies the infinite sequence, generated by A,B,C.

b. Transformation of the TIR problem into a CSIR or OSIR problem.

Normally, one is interested in the solution of the TIR problem. However, a direct solution of eq.(9-1) requires the solution of a set of nonlinear matrix equations in C, A, and B. For this and other reasons, that will become clear when the approximation problem will be considered, we transform the TIR problem into a CSIR or OSIR problem, as follows:

TIR ⟶ CSIR: Choose C and define X(k)

$$C_2 = I_{l,l.i_2} \cdot P^{-1} \qquad (9\text{-}5a)$$

$$X(k) = P \cdot [H(k) \ldots H(k+i_2-1)]^T \quad , \quad k=1 \ldots K_2 \qquad (9\text{-}5b)$$

$$i_2 \geqslant i_o \quad , \quad K_1 - i_2 + 1 \stackrel{\Delta}{=} K_2 \geqslant i_c + 1 \qquad (9\text{-}5c)$$

TIR ⟶ OSIR: Choose B and define $\Xi(k)$

$$B_3 = Q \cdot I_{m.i_3,\, m} \qquad (9\text{-}6a)$$

$$\Xi(k) = [H(k) \ldots H(k+i_3-1)] \cdot Q^{-1} \quad , \quad k=1 \ldots K_3 \qquad (9\text{-}6b)$$

$$i_3 \geqslant i_c \quad , \quad K_1 - i_3 + 1 \stackrel{\Delta}{=} K_3 \geqslant i_o + 1 \qquad (9\text{-}6c)$$

In the above expressions, P and Q are nonsingular matrices of appropriate dimensions, that can be freely chosen. I_{ab} is a pseudo-unit matrix: a top left (axb) submatrix of a sufficiently large unit matrix. Superscripts t and T denote the transpose resp., blocktranspose. The transformations of TIR into CSIR or OSIR essentially imply a choice of the parameters i_2 and i_3 which are safe overestimates for the observability and controllability indices.

The recognition that a finite segment of the Transfer Impulse Response is the representation of an associated state impulse response in some basis, seems to be so widespread that the author doesn't know its precise provenance. An elegant constructive way to prove the validity of this statement follows from the properties of a so called "trivial representation" [103]:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \left[\begin{array}{cccccc|c} 0 & I & 0 & 0 & \dots & 0 & H(1) \\ 0 & 0 & I & 0 & \dots & 0 & H(2) \\ \dots & \dots & \dots & \dots & \dots & 0 & \dots \\ 0 & 0 & 0 & 0 & \dots & I & \dots \\ -a_0 I & -a_1 I & -a_2 I & \dots & \dots & -a_{r-1} I & H(r) \\ \hline I & 0 & 0 & 0 & \dots & 0 & H(0) \end{array}\right] \qquad (9\text{-}7a)$$

Where I denotes the (1x1) unit matrix, r is an overestimation of the controllability index of the system, and the a_i are the autoregressive (AR) coefficients of the system, i.e.:

$$\forall\, k>0 : H(k+r) = \sum_{i=0}^{r-1} - a_i . H(k+i) \qquad (9\text{-}7b)$$

It is then easily verified that the Transfer Impulse Response samples of eq.(9-7) form the given sequence H(k), and that for $i_2 = r$ the CSIR choice for X(k), made in (9-5), corresponds to the State Impulse Response of the above system. This justifies the "Controllable State Impulse Response" terminology, and in a completely analogous fashion, the OSIR formulation can be justified. Once the X(h) and $\Xi(k)$ in eqs.(9-5b),(9-6b) are recognized as true controllable resp., observable State Impulse Response (block) vectors, it is readily understood that the AR order r is an unnecessarily high (common) lower bound for i_2 and i_3. Instead, from the definition of the controllability and observability indices, and from an argument similar to the proof of [36:p.181], it follows that the eqs.(9-4) are the correct (i.e. necessary and sufficient) lower bounds for the involved data quantities. A very important observation here is, that the above trivial realization not only is overdimensioned, but also is far from being unique for the chosen dimension. The particular companion form SSR of (9-7a) is numerically quite sensitive, and should therefore be used only when it can help to understand the meaning of the involved background algebra (as in this case); NEVER in a numerical context. We will see further, that there exist much more reliable Least Squares representations for the same H(k) sequence, and for the dimension r.

c. Reformulation of CSIR, OSIR and TIR as linear problems.

From eqs.(9-2) and (9-3) it follows that

$$X(k+j) = A_2^j . X(k) \qquad (9\text{-}8a)$$

$$\Xi(k+j) = \Xi(k) . A_3^j \qquad (9\text{-}8b)$$

Hence the sets of CSIR and OSIR equations are equivalent to the following sets of equations:

$$\underline{\text{CSIR}} : A_2^j \,[X(1) \ \ldots \ X(K_2-j)] \ = \ [X(j+1) \ \ldots \ X(K_2)] \qquad (9\text{-}9a)$$

$$[I \ \ A_2 \ \ldots \ \ A_2^{J_2}]^T . \ B_2 \ = \ [X(1) \ \ldots \ X(J_2+1)]^T \qquad (9\text{-}9b)$$

$$\text{where} \quad j = 1 \ \ldots \ J_2 \ \leqslant \ K_2 - 1 \qquad (9\text{-}9c)$$

$$\underline{\text{OSIR}} : [\Xi(1) \ \ldots \ \Xi(K_3-j)]^T . A_3^j \ = \ [\Xi(j+1) \ \ldots \ \Xi(K_3)]^T \qquad (9\text{-}10a)$$

$$C_3 \,[I \ \ A_3 \ \ldots \ \ A_3^{J_3}] \ = \ [\Xi(1) \ \ldots \ \Xi(J_3+1)] \qquad (9\text{-}10b)$$

$$\text{where} \quad j = 1 \ \ldots \ J_3 \ \leqslant \ K_3 - 1 \qquad (9\text{-}10c)$$

Using (9-5b) and (9-6b), the TIR equivalents of (9-9a),(9-10a) are then:

$$\underline{\text{TIR} \longrightarrow \text{CSIR}} : \ A_2^j . \ H_{i_2}(1,K_1-j) \ = \ H_{i_2}(j+1,K_1) \ , \ j=1 \ \ldots \ J_2 \qquad (9\text{-}11a)$$

$$\underline{\text{TIR} \longrightarrow \text{OSIR}} : \ H_{i_3}^T(1,K_1-j) . \ A_3^j \ = \ H_{i_3}^T(j+1,K_1) \ , \ j=1 \ \ldots \ J_3 \qquad (9\text{-}11b)$$

where the MBH (Markov Block Hankel) matrix appears as:

$$H_a(b,c) \ \triangleq \ P \ . \begin{bmatrix} H(b) & H(b+1) & \ldots & H(c-a+1) \\ \ldots & \ldots & & \ldots \\ H(b+a-1) & H(b+a) & \ldots & H(c) \end{bmatrix} . \ Q^{-1} \qquad (9\text{-}12a)$$

$$J_2 \ \leqslant \ K_1 - i_2 \qquad (9\text{-}12b)$$

$$J_3 \ \leqslant \ K_1 \ \ - i_3 \qquad (9\text{-}12c)$$

$$i_2 \ \geqslant \ i_o \qquad (9\text{-}12d)$$

$$i_3 \ \geqslant \ i_c \qquad (9\text{-}12e)$$

and P and Q are any nonsingular matrices of appropriate dimensions.

The bounds (9-9c),(9-10c),(9-12b), and (9-12c) on J_2 resp., J_3 still assume that the matrix A_2 resp., A_3 are identical for all equations (9-9a) resp., (9-10a),(9-11a) and (9-11b).The linearization of the problem formulation however requires that for each power j, the latter sets of equations can be solved independently. Then the bounds on J_2 and J_3 become:

$$J_2 \leqslant K_1 - i_2 - i_c + 1 \qquad (9\text{-}13a)$$

$$J_3 \leqslant K_1 - i_o - i_3 + 1 \qquad (9\text{-}13b)$$

d. Minimal realizations.

The formulations in the preceding paragraphs still allow for a multitude of solutions. A significant degree of freedom left, is the dimension of the realization. There are practical, and more fundamental reasons [46] to require solutions of minimal dimension. In the TIR case, one simply requires a minimal dimension n for matrix A in eq.(9-1). In the CSIR and OSIR case however, the dimension N is a given parameter. A consistent requirement is then to find a matrix couple A_2^*, B_2^* of lowest A-matrix dimension n, that generates a sequence

$$X^*(k) = A_2^{*\,k-1} . B_2^* \qquad (9\text{-}14a)$$

that is equivalent, in dynamic behaviour, to the given sequence X(k). By this, we mean that there should exist a nonsingular similarity transformation between X*(k), imbedded in an N-dimensional ambient space, and X(k). Analogously, for the OSIR case, one requires a matrix couple A_3 , B_3 of lowest A-matrix dimension n, that generates a sequence

$$\Xi^*(k) = C_3^* \; A_3^{*\,k-1} \qquad (9\text{-}14b)$$

The general technique consists in projecting a nonminimal realization on its controllable subspace (CSIR) or along its unobservable subspace (OSIR) or both (TIR). The (unique) controllable subspace S_c is found as

$$S_c \overset{\text{CSIR}}{=\!=\!=} \text{Span}_{\text{col}} [X(1) \ \ldots \ X(K_2)] \qquad (9\text{-}15a)$$

$$\overset{\text{TIR(CSIR)}}{=\!=\!=} \text{Span}_{\text{col}} [H_{i_2}(1,K_1)] \qquad (9\text{-}15b)$$

For the quotient spaces along the unobservable subspaces, any (direct-sum) complement is valid. For numerical reasons the (unique) ORTHOGONAL complement S_o is the most appropriate:

$$S_o \overset{\text{OSIR}}{=\!=\!=} \text{Span}_{\text{row}} [\Xi(1) \ \ldots \ \Xi(K_3)]^T \qquad (9\text{-}15c)$$

$$\overset{\text{TIR(OSIR)}}{=\!=\!=} \text{Span}_{\text{row}} [H_{i_3}^T(1,K_1)] \qquad (9\text{-}15d)$$

As described in chapter 2, these spaces are readily available from an SVD of the involved matrices, and the associated projections can be performed in an easy and numerically stable way. Without loss of generality, we further only consider this orthogonal projection. [Remember that the matrices P and Q are still free to choose in eq.(9-12)]. Then with the SVD's:

$$\underline{\text{CSIR}} \quad : \quad [X(1) \ \ldots \ X(K_2)] = \tilde{U}_X \ \tilde{\Sigma}_X \ \tilde{V}_X \tag{9-16a}$$

$$\underline{\text{OSIR}} \quad : \quad [\Xi(1) \ \ldots \ \Xi(K_3)]^T = \tilde{U}_\Xi \ \tilde{\Sigma}_\Xi \ \tilde{V}_\Xi \tag{9-16b}$$

$$\underline{\text{TIR}} \quad : \quad H_a(1,K_1) = \tilde{U}_{H_a} \ \tilde{\Sigma}_{H_a} \ \tilde{V}_{H_a} \tag{9-16c}$$

the minimal realizations follow from the nonminimal ones as follows:

$$\underline{\text{CSIR}} \quad : \quad A_{02} = \tilde{U}_X^t \ A_2 \ \tilde{U}_X \tag{9-17a}$$

$$B_{02} = \tilde{U}_X^t \ B_2 \tag{9-17b}$$

$$\underline{\text{OSIR}} \quad : \quad A_{03} = \tilde{V}_\Xi \ A_3 \ \tilde{V}_\Xi^t \tag{9-18a}$$

$$C_{03} = C_3 \ \tilde{V}_\Xi^t \tag{9-18b}$$

$$\underline{\text{CSIR} \longrightarrow \text{TIR}} \quad : \quad A_{12} = \tilde{U}_{H_{i_2}}^t \ A_2 \ \tilde{U}_{H_{i_2}} \tag{9-19a}$$

$$B_{12} = \tilde{U}_{H_{i_2}}^t \ B_2 \tag{9-19b}$$

$$C_{12} = C_2 \ \tilde{U}_{H_{i_2}} \tag{9-19c}$$

$$\underline{\text{OSIR} \longrightarrow \text{TIR}} \quad : \quad A_{13} = \tilde{V}_{H_{i_3}^T} \ A_3 \ \tilde{V}_{H_{i_3}^T}^t \tag{9-20a}$$

$$B_{13} = \tilde{V}_{H_{i_3}^T} \ B_3 \tag{9-20b}$$

$$C_{13} = C_3 \ \tilde{V}_{H_{i_3}^T}^t \tag{9-20c}$$

Where the wiggle on a matrix denotes that only the submatrices associated to the principal left and/or right principal singular spaces of dimension n are used. Based on all the above, we can now state the final

e. Minimal Transfer Impulse Response Realization problem:

For a given sequence of K_1 Transfer Impulse Response matrices H(k), find a triple of matrices (A_1,B_1,C_1) satisfying the minimality conditions (9-16), (9-19) and (9-20), and in which the nonminimal triples (A_2,B_2,C_2) and (A_3,B_3,C_3) satisfy the general realization conditions (9-8)...(9-13).

f. Discussion of the final formulation.

The formulations (9-8)-(9-20) are generalizations of the classical Markov Block Hankel (MBH) formulations, with attractive properties. Indeed, stated as above, the TIR problem has an appealingly simple geometric interpretation: For instance, in the CSIR case the X(k) obtained DIRECTLY from the given H(k), form a (multiple) state impulse response for the couple $[A_2,B_2]$. Eq.(9-9a) and (9-10a) relate these samples separated over j time-steps through A_2^j . Notice that all earlier MBH-formulations only consider the case j=1. Eq.(9-9b) and (9-10b) relate the input matrix B_2 to the successive state impulse response samples via these A_2^j. Again in the literature one finds I.B = B = X(1) only. Now stated as such, impulse response realization is a mere fitting of matrices A_2,B_2,C_2 to these state impulse responses. Also, the reduction to a minimal SSR is the mere projection of the large trivial state space, associated to A_2, on the subspace S_c, in which the X(k) live. The OSIR case is completely analoguous.

Notice above all the formal simplicity of the results, and the transparency of their interpretation. Observe also, that in the final formulation, all unknown matrices can be obtained from explicite expressions, or as the solution of standard linear problems such as SVD, or the generalized solution of over- or underdetermined sets of linear equations. Notice that the eqs.(9-9) to (9-11) are redundant in the case of exact data, but they are put into a form, particularly suited for a reliable solution in the case of noisy data, where least squares techniques can be used to reduce the effect of random inaccuracies. Moreover, the choice of matrices P and Q in eqs.(9-5) and (9-6) is an additional degree of freedom, that offers promising numerical advantages such as optimal row and column scaling as a preprocessing step on the MBH matrix (9-12) before attacking its decomposition. All this will allow to design motivated solution strategies depending on the additional requirements on the realization, and to discern clearly the meaning, the advantages and the dangers of each of these strategies. Finally, it will be easy to relate most of the particular results of this general approach to earlier results in the literature.

B. TYPES OF SOLUTION STRATEGIES.

In this section we still assume exact data and exact arithmetic. The main purpose is to gain insight in the geometric significance of the algebraic manipulation of each of the discussed strategies, in order to orient the procedure to be followed for reliable realization of noisy data in finite precision arithmetic (the approximation problem). Notice the difference between a strategy and an algorithm. The STRATEGY decides on what is important in the realization (computation speed; storage efficiency; numerical reliability; input oriented, output oriented or balanced results; a complete but possibly unreliable model, a restricted but highly reliable submodel...). The ALGORITHM implements the solution algebra according to this strategy. In the case of exact data, all (correct) strategies will lead to a same result (up to a similarity transformation). But in the case of noisy data, results that follow from different strategies may be substantially different. Typically, the backward differences between results of different strategies will be of the order of the precision on the DATA, but these differences may be of conceptual significance. On the other hand (provided the algorithms are stable of course), the backward differences between results of different algorithms within one strategy will not differ more than by a modest multiple of the MACHINE PRECISION, and should NOT be of any conceptual importance. The significance and relevance of these general statements should emerge from the succeeding sections.

a. Fast (recursive and/or Gaussian) algorithms.

Fast algorithms, requiring $O(N^2)$ operations for the solution of equations (9-11a),(9-11b) with $J_2=1$ or $J_3=1$ exist in the literature [21,28,29,50]. They are recursive algorithms based on Gaussian elimination, but are numerically unreliable in the case of noisy data. De Jong describes a numerically stable recursive algorithm [21], requiring however $O(N^3)$ operations. Staar [103] described an even faster pick-out procedure, that has the additional nice property of using a lot of the available data DIRECTLY in the model, thus avoiding accumulation of the errors.

The main inconvenience of these algorithms however, is that they are all still based on Gaussian elimination, which is CONCEPTUALLY ill in the case of approximations based on noisy data. Apart from the well known argument in numerical analysis, that Gaussian elimination is not suitable for rank determination, the reader will easily understand the point from the following simple case study:

Suppose that the rank of the MBH matrix H is 1 (first-degree system). Then all columns in H are proportional to one and a same column. This column spans the actual sub State Space to which the ambient State Space has to be reduced, and the coefficient relating one column to the next one is the eigenvalue of the minimal system. Now

1. Recursive algorithms will use the precise number of H(k) AS THEORETICALLY REQUIRED BY THE DEGREE OF THE SYSTEM, i.e. they will compute the eigenvalue as the inverse ratio of the first two successive values of the impulse response and neglect all other available measurements.

This approach will obviously never lead to high quality results, and particularly not in the quite usual case where the first impulse response samples are small, and the inaccuracies have a fixed absolute level over the entire segment $1...K_1$.

2. Gaussian elimination with partial pivoting will use a pivot in the first column, and compute the eigenvalue as the inverse ratio between this pivot and the corresponding entry in the second (block-) column. Intrinsically, this is already better for impulse responses, that grow before they finally decay, because the largest among all entries in the first column can now be chosen as a pivot. For a situation of inaccuracies with a constant absolute level, the largest entries in the MBH matrix have the highest relative accuracy, and so has then their quotient (i.e. the resulting eigenvalue).

Gaussian elimination with complete pivoting, dicussed in [103], is even more reliable, and in fact obtains the best possible result for this type of algorithms. Essentially however, the eigenvalue is still computed as the ratio of TWO successive AMONG ALL available measurements of the impulse response. Also, the associated column is readily taken as the basis of the minimal state space. Clearly an overall (properly weighed) average of all the ratios over all pairs of successive impulse response samples is a much more reliable measure for the systems eigenvalue. And the direction, closest to all the involved columns is definitely a better basis for the minimal State Space.

From the above, one understands that Gaussian elimination is certainly not the most reliable tool for realization purposes, and that much better methods should exist. From chapter 2, the reader can have the impression that the above ad hoc (and quite limited!) approach is somehow what SVD performs in general. This impression will appear to be correct.

And so we will not further consider the above fast algorithms based on recursion or Gaussian elimination, but rather focus the attention on more reliable realization schemes, strongly suggested here to be based on SVD.

b. CSIR-type (input oriented) and OSIR-type (output oriented) strategies.

Choosing P=Q=I, $J_2=1$, $J_3=1$, the simplest solutions to eqs.(9-9)-(9-20) follow from the SVD's of the involved MBH matrices as follows:

CSIR: $$H_{i_2}(1,K_1) = \tilde{U}_2 \tilde{\Sigma}_2 \tilde{V}_2 \qquad (9\text{-}21)$$

$$A_{12} = \tilde{U}_2^t \; H_{i_2}(2,K_1) \; H_{i_2}^+(1,K_1-1) \; \tilde{U}_2 \qquad (9\text{-}22a)$$

$$B_{12} = \tilde{U}_2^t \; X(1) = \tilde{\Sigma}_2 \tilde{V}_2 \; [\, I_m \; 0 \,]^T \qquad (9\text{-}22b)$$

$$C_{12} = [\, I_1 \;\; 0 \,] \tilde{U}_2 \qquad (9\text{-}22c)$$

OSIR: $$H_{i_3}^{T}(1,K_1) = \tilde{U}_3 \tilde{\Sigma}_3 \tilde{V}_3 \qquad (9\text{-}23)$$

$$A_{13} = \tilde{V}_3 \; H_{i_3}^{T+}(1,K_1\text{-}1) \; H_{i_3}^{T}(2,K_1) \; \tilde{V}_3^{t} \qquad (9\text{-}24a)$$

$$B_{13} = \tilde{V}_3 \; [\, I_m \;\; 0 \,]^T \qquad (9\text{-}24b)$$

$$C_{13} = \Xi(1) \; \tilde{V}_3^{t} = [\, I_1 \; 0 \,] \; \tilde{U}_3 \; \tilde{\Sigma}_3 \qquad (9\text{-}24c)$$

Now, if the reduction is performed based on the smaller MBH matrices

$$H''_{i_2}(1,K_1\text{-}1) = \tilde{U}''_2 \; \tilde{\Sigma}''_2 \; \tilde{V}''_2 \qquad (9\text{-}25a)$$

$$H''^{T}_{i_3}(1,K_1\text{-}1) = \tilde{U}''_3 \; \tilde{\Sigma}''_3 \; \tilde{V}''_3 \qquad (9\text{-}25b)$$

(leading to only marginally less reliable results when K_1 is large enough) then eqs.(9-22a),(9-24a) become extremely simple:

$$A''_{12} = \tilde{U}''^{t}_2 \; H_{i_2}(2,K_1) \; \tilde{V}''^{t}_2 \; \tilde{\Sigma}''^{+}_2 \qquad (9\text{-}26a)$$

$$A''_{13} = \tilde{\Sigma}''^{+}_3 \; \tilde{U}''^{t}_3 \; H_{i_3}^{T}(2,K_1) \; \tilde{V}''^{t}_3 \qquad (9\text{-}26b)$$

c. Balanced strategies.

Comparing both above simple expressions (9-26) for $i_3 = K_1 - i_2$, then from

$$H''_{i_2}(1,K_1\text{-}1) = \tilde{U}''\tilde{\Sigma}''\tilde{V}'' = H''^{T}_{i_3}(1,K_1\text{-}1) \qquad (9\text{-}27)$$

it follows that the CSIR and OSIR approximations in (9-26) are EXACTLY similar, and that the similarity transformation is $\tilde{\Sigma}''$. This very important observation implies that there is no essential difference between the CSIR and OSIR APPROXIMATION, or more precisely: the I-O characteristics of both CSIR and OSIR STRATEGIES are EXACTLY equal. For symmetry purposes one can then derive a balanced realization

$$A'' = \tilde{\Sigma}''^{-1/2} \; \tilde{U}''^{t} \; H_{i_2}(2,K_1) \; \tilde{V}''^{t} \tilde{\Sigma}''^{-1/2} \qquad (9\text{-}28a)$$

$$B'' = \tilde{\Sigma}''^{1/2} \; \tilde{V}''^{t} \; [\, I_m \; 0 \,]^T \qquad (9\text{-}28b)$$

$$C'' = [\, I_1 \quad 0 \,]\, \tilde{U}''^{t}\, \tilde{\Sigma}''^{\,1/2} \qquad (9\text{-}28c)$$

that can be considered as a "geometric mean" of both approaches.

d. Some remarks.

The above simple types of algorithms correspond fairly well to the original idea of Ho and Kalman: the MBH matrix $H(1,K_1\text{-}1)$ is reduced by operators U and V to a new matrix, in which only the upper left (nxn) nonsingular matrix is nonzero. Then the minimal realization is obtained by applying the same operators to the MBH matrix shifted over one (block) column resp., one (block) row, and by restricting the result to the nonvanishing part ([26],[36], and eqs.(9-26)-(9-28)). But the expression

$$\tilde{U}_2^{t}\; H_{i_2}(2,K_1)\quad H_{i_2}^{+}(1,K_1\text{-}1)\quad \tilde{U}_2 \qquad (9\text{-}29)$$

in eqs (9-22)-(9-24) also show a very close similarity to a result of Silverman [27]: eq.(9-22a) is a generalization of his expression $T_1\, T^{-1}$, in which T is a nonsingular submatrix of $H(1,K_1\text{-}1)$, and T_1 is the submatrix of $H(2,K_1)$ with the same row and column indices as T. All this shows the very close relationship between the Ho and the Silverman algorithms.

The use of SVD for the numerical improvement of these types of algorithms was first proposed by Zeiger and Mc Ewen [30]. Moore [64] presents results for continuous time systems, using SVD and geometric arguments joining the geometric interpretation given above. Finally, eq.(9-28) is closely related to the idea of Kung [31] on balanced realizations.

e. Strategies to improve the reliability of B or C.

When solving eq.(9-11) for j=1 only, it will probably be felt that plant matrix A is obtained from a matching of a sufficient amount of data, whereas B and/or C in (9-9b),(9-10b) use much less data. Indeed, if one uses B=X(1) for instance, then the state impulse response after one time step corresponds EXACTLY to the GIVEN (possibly poisened) DATA, whereas the subsequent samples move more and more away from the given data, due to a propagation of errors in the successive powers of A. Therefore, a matching of B or C to more data, available in H, is advisable. The required powers of A can either be obtained directly from the data using eq.(9-11a) resp., (9-11b) with $j>1$, or can be calculated as powers of the matrix A obtained for j=1. The latter has the advantage that the obtained B matrix will compensate directly the accumulation of errors in the USED plant matrix A, hence is advisable when the purpose of the realization is a simulation of the system behaviour. The former has the advantage that it works on more available data, hence is expected to yield a B- or C-matrix closer to the correct B or C. Often however, the complete solution of (9-9)-(9-10) is far too expensive. Then a good compromise is to solve B or C from

$$[I \quad A_2^{k_1} \quad A_2^{k_2} \quad]^T B_2 = [\, X(1) \quad X(k_1+1) \quad X(k_2+1) \,]^T \qquad (9\text{-}30a)$$

where $X(k_1+1)$ and $X(k_2+1)$ are two significant state impulse response samples, well distributed over the time interval $0 < k_1 < k_2 \leq K_1$.

Analogously, for the OSIR case, one solves

$$C_3 \; [I \quad A_3^{k_1} \quad A_3^{k_2} \;] = [\, \Xi(1) \quad \Xi(k_1+1) \quad \Xi(k_2+1) \,] \qquad (9\text{-}30b)$$

f. Special strategies for slow systems.

When the considered system is slow, it might be numerically much more rewarding to match A^j (relating IR states, j time-steps separated), and then to extract the actual matrix A from A^j. For example:

$$A_{12}^j = \tilde{U}_2^t \; H_{i_2}(j+1, K_1) \; H_{i_2}^+(1, K_1-j) \; \tilde{U}_2 \qquad (9\text{-}31a)$$

$$A_{12} = (A_{12}^j)^{1/j} \qquad (9\text{-}31b)$$

Notice that A^j can be considered as the A matrix of a discrete time system sampled j times slowlier, and hence eq.(9-31) is valid only when no fast modes are present in the IR. The computation (9-31) is however more accurate than when obtained from a simple decimation of the input and output data, since ALL the data are used. Of course the computation of matrix A in eq.(9-31) is not easy [65], and selecting the correct j-th (real) root is not at all trivial. Hence this strategy is advisable only in extremely critical situations.

g. An exotic strategy.

As a conclusion of the above overview of possible strategies to attack the realization problem in a reliable way, we present a rather exotic scheme, that is valid for finite impulse response systems, but shows somewhat confusing properties when analysed in some more detail.

For a system with a Finite Impulse Response [i.e. $H(k)=0$, $k=K_1+1 \ldots$], compute the SVD:

$$H_{K_1}(1, 2.K_1-1) = \tilde{U} \; \tilde{\Sigma} \; \tilde{V} \qquad (9\text{-}32)$$

Then:

$$A_{22} = \tilde{U}^t . [\ 0_{1.K_1,\ 1} \quad I_{1.K_1,\ 1.(K_1-1)}\ ] . \tilde{U} \qquad (9\text{-}33a)$$

$$B_{22} = \tilde{U}^t\ [\ H(1) \ \ldots\ H(K_1)\ ]^T = \tilde{\Sigma}\ \tilde{V}\ I_{1.K_1,\ m} \qquad (9\text{-}33b)$$

$$C_{22} = I_{1,1.K_1} . \tilde{U} \qquad (9\text{-}33c)$$

The proof of this realization scheme is based on the reduction of an over-dimensioned trivial realization of autoregressive order K_1 [103] that matches exactly the given impulse response sequence, and in which the A matrix contains zero autoregressive coefficients.

This trivial realization is completely observable and its reduction to a minimal realization is a restriction to its controllable subspace. But in this reduction the zero AR part of A drops out in the computations. This procedure is exotic because the poles of the system are fixed at 0, and all other information on the system is contained in U and X(1), while the singular values in Σ are not necessarily used. Since no approximation is made, for exact H(k) the realization procedure yields an exact [A,B,C] model. But for sufficiently large K_1, the discrete time impulse response of any asymptotically stable linear system can come arbitrarily close to such a FIR impulse response. The conclusion is that the positions of the model poles are mainly determined in the reduction step of the algorithm. This realization scheme is closely related to the one presented by Kung [31] for infinite, stable sequences, and its involved paradox will again show up in chapter 10, in the note on parametrized identification.

C. THE APPROXIMATION PROBLEM.

a. Introductory treatment.

For the time being, assume P=Q=I in eqs. (9-5)-(9-12). As indicated in the previous section, the choice of the block dimensions of H is fairly free, provided the conditions (9-13) are met. This choice might be restricted by practical limitations (storage capacity, computation power...), but can otherwise be based on numerical arguments. For conceptual purposes however, it is interesting to consider the particular case of CSIR where the MBH matrix H has more rows than columns. The solution of eq.(9-11a) is then an underdetermined problem, and in the generic case

$$\text{span}_{\text{row}}\ [H_{i_2}(j+1,K_1)] \quad \subset \quad \text{span}_{\text{row}}[H_{i_2}(1,K_1-j)] \qquad (9\text{-}34)$$

so that an A_2^j can always be solved WITHOUT approximation. If the H(k) are noisy, the solution will not be correct, but it will match eq(9-11a), and hence the noisy sequence X(k) EXACTLY. Now taking B_2=X(1), A_2 from eq.(9-11a) with j=1, and C_2 as in (9-11c), the given sequence H(k) is matched EXACTLY by the matrix triple (A_2 B_2 C_2), which in this way forms an

EXACT, but still overdimensioned realization of the noisy sequence

$$H(k),\ k=1\ldots K_1 \qquad (9\text{-}35)$$

and the full approximation problem can be studied in the model reduction procedure (9-20). Notice that this notion of "exact" but overdimensioned realization is common in the transfer domain as well as in the state space domain, but our approach will allow for a precise and physically more meaningful interpretation of the succeeding reduction procedures. This will lead to a physically motivated degree determination criterion, based on the signal to noise ratio concept of chapter 2.

From the SVD of the state impulse response matrix H, one indeed obtains singular directions u_i and weighting coefficients σ_i, indicating for how much this noisy state impulse response seems to live in these directions.

From the noise level on the data it is now possible to compare the oriented energy, expected purely from noise and inaccuracies, with the actually measured energies. We will further denote the state impulse response as "signal", and the inaccuracies on this impulse response, whatever their origin, as "noise". From a required signal to noise ratio, it is then possible with algorithm (2-1), to extract a state impulse response of highest column-span dimension n, that is as close as possible to the given data, and has the highest MSNR of all possible approximations with this column-span dimension. The reduction of the state space dimension to n implies a model reduction to degree n, the rejection of N-n poor SNR-directions in the ambient state space, and hence the rejection of an energy

$$E_r = \sum_{i=n+1}^{N} \sigma_i^2 \qquad (9\text{-}36)$$

that can be considered to be associated to the relatively most heavily noise-poisened part of the signal, if not pure noise at all.

The MSNR invariance for linear transformations (Theorem 2-2) implies the invariance of this signal-to-noise ratio criterion for similarity transformations on the State Space Realization. This means that with this procedure, one extracts an n-dimensional approximation of the state impulse response, so that

1. An MSNR is obtained, higher than the required signal to noise ratio.
2. No extraction of dimension higher than n can meet this requirement.
3. No higher MSNR is possible in any State Space basis of dimension n.

The last property leaves the choice of the matrices P and Q totally free for row and column scaling operations motivated by numerical arguments, but the author has never explicitely exploited this degree of freedom.

Notice that the above reduction to a State Space of a lower dimension is only a first step to the full realization. To this dimensionally reduced state impulse response, the three operators A,B,and C, (the dimensions of which are now well determined...) can now be matched in a LLS sense.

Two remarks are in order here:

1. Although both steps in the algorithm are "optimal" in LLS sense, the successive application of these steps could not be proved to be globally optimal. The author believes that, if not strictly globally optimal, this strategy is (very) close to the globally optimal solution because the approximations in both steps are "orthogonal" in the sense that the corrections applied to the state impulse response in both steps are performed within orthogonal subspaces. As said, this is only a feeling, no proof is available, and neither is any other globally optimal method known to the author, in order to make simulative comparison possible. This question, thus, remains open.

2. Although the successive steps in the algorithm are treated cronologically for conceptual purposes, their implementation can be highly integrated, so that the (storage and time expensive) intermediate results are not to be explicitely computed in the actual implementation.

b. Approximate realization with guaranteed signal to noise ratio.

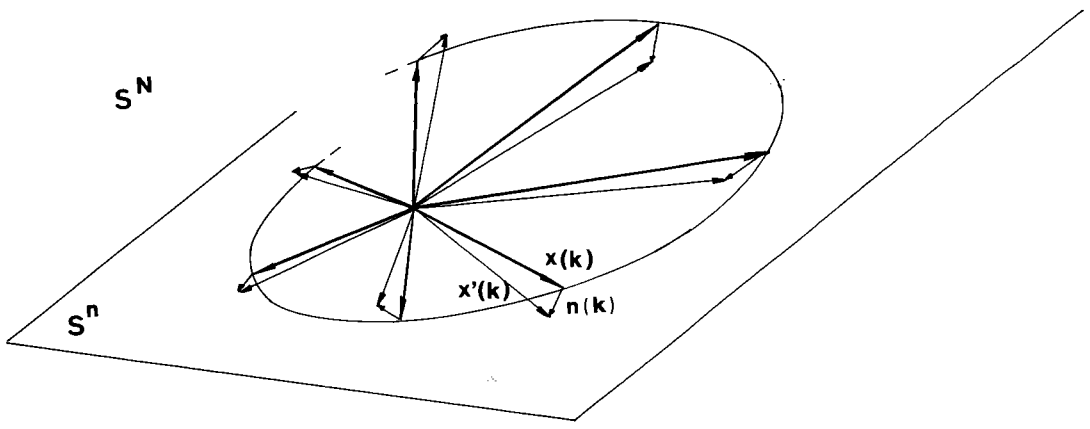


Fig.9-1: Illustration of the approximate realization problem.

The procedure is briefly described as follows (Fig.9-1): from the algebraic treatment, it follows that the sequence of exact (multivariable) transfer impulse response samples

$$H'(k) \ , \ k=1...K_1 \qquad (9\text{-}37)$$

can be associated with a sequence of state impulse response samples X(k) or $\Xi(k)$ in some basis , which we propose to call the "natural input resp., output basis" for the ambient state space of the system. These vector sequences are the columns resp., rows of the Markov Block Hankel (MBH) expansion of this sequence.

This (exact) state impulse response lives in an n-dimensional subspace S^n of its ambient space (notice that fig.9-1 assumes m=1; for m>1, each time sample is composed of m vectors). With an SVD of the MBH matrix, this internal state impulse response of the system can be decomposed in n orthonormal signals, the sum of energies of which is equal to the total energy in the state impulse response. ZERO ORIENTED ENERGY is measured in the directions orthogonal to S^n. The restriction of the state space to this subspace naturally leads to a minimal realization for the exact sequence. The influence of additive noise n(k) is due to the inaccuracy of the available H'(k) and causes X(k) to leave the minimal subspace S^n. So generically, the noisy X(k) live in S^N. In the directions where the state impulse response makes large excursions, the noise cannot affect the measurements significantly. In the directions, where the exact impulse response makes no excursions at all, the motion of the noisy H(k) is due totally to the inaccuracies, and should certainly be dropped. When the oriented energy of the exact State Impulse Response is relatively small, then the noise will significantly poisen the behaviour in this direction, and the problem of dropping or taking into account such a mode (which essentially determines the degree) becomes a decision rather than a question with a computable solution. An important observation is that there might be modes in the exact sequence that have no significant energy compared to the noise level, and that NO TRANSFORMATION OF THE BASIS NOR ANY LINEAR MEASUREMENT METRIC CAN CHANGE THIS FACT (Invariance of the MSNR). In that case it would not be wise to require a realization of the same degree as the exact realization, even if for physical reasons it would be possible to independently determine a theoretical degree, because a realization of such a degree would essentially model a noise sequence in a deterministic way.

The proposed method now consists in approximating the given data X(k) by an approximate sequence X"(k) of maximal MSNR, but living in a subspace of dimension n. 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 on the entries of H(k), and a required S/N ratio R_o.

Then from chapter 2 it follows that if $K = m.K_2 > l.i_2$, and

$$\sigma_1^2 \geq \sigma_2^2 \geq \ldots \sigma_n^2 \geq K \sigma_v^2 R_o > \sigma_{n+1}^2 \geq \ldots \geq \sigma_N^2 \qquad (9\text{-}38)$$

then S^n has an expected MSNR $R \geq R_o$, where

$$R = \frac{\sigma_n^2}{K \cdot \sigma_v^2} \qquad (9\text{-}39)$$

Hence it is obvious that the system degree should be n.

Once this choice is made, a realization of degree n follows by an optimal matching of the operators A,B, and C to the n-dimensionally reduced data X", in such a way that the matching error of the n-dimensionally reduced

state impulse response vectors X"(k) to X"(k+j) is minimized:

$$\sum_{k=1}^{K_2-j} \left\| A_{12}^{j}.X"(k) - X"(k+j)\right\|_F^2 = \min_{A^j} \sum_{k=1}^{K_2-j} \left\| A^{j}.X"(k) - X"(k+j)\right\|_F^2 \qquad (9\text{-}40)$$

Stating it loosely, eq.(9-40) expresses that A_{12}^{j} is chosen so that the vector relation X"(k), X"(k+j) is matched optimally in a linear least squares sence, i.e. the vector field associated to A_{12}^{j} is matched optimally to the available vector couples.

Notice that if arbitrary data are fed to the algorithm, it might very well be (and in fact it will generically be the case) that the n-dimensionally restricted data do not lead to a matching within the assumed noise level. This is because an arbitrary set of vectors, living in an n-dimensional space, is not necessarily close to the state impulse response of any n-dimensional linear system. In this case the set of data are flagrantly conflicting and simply can not correspond to the impulse response of an n-th degree linear system. (One should not expect to be able to model any arbitrary behaviour with a linear finite dimensional model!). But in the remainder we will assume that the data form indeed the impulse response of a linear system within a given level of inaccuracies, in which case the above MSNR can be guaranteed.

Simplified procedure.

If K_1 is large enough, the dimension reduction on X(k) can be performed with the U matrix of a decomposition (9-26) of a smaller MBH matrix

$$H_{i_2}(1,K_1-j) = \tilde{U}"_H \, \tilde{\Sigma}"_H \, \tilde{V}"_H \qquad (9\text{-}41)$$

with this decomposition, the dimension reduction and the approximation can be performed with one and the same SVD on one and the same matrix.

REMARKS

1. The simplified procedure for j=1 and a fixed n is equivalent to the original Zeiger and Mc-Ewen [30] approach.

2. For each j, the solution of eq.(9-23a) yields useful information on A. It is an unsolved problem how to combine the information from higher powers of A in an optimal way, to obtain the most reliable global estimation for the matrices A,B,C.

3.The important advantage of our approach is the motivated choice for the degree of the realization.

4.The algorithm models stable as well as unstable impulse responses. Contrary to some other algorithms, there is no explicit nor inherent requirement that the realized SSR of a stable system be stable. However, except

for extreme examples (a pure oscillator or a pure integrator for instance), no problems have been encountered. In fact, if the given data are by accident closer to an unstable impulse response than to a stable one, one can hardly blame the algorithm for finding this unstable system representation rather than a more distant stable model.

5. As said earlier, the solution is found in two conceptual steps, each of which is LLS optimal: the optimal reduction of the state impulse response to an n-dimensional subspace; and the optimal matching of operators A,B,C to this dimension-reduced vector sequence. It is not known whether or not the resulting solution is also globally optimal.

D. FEASIBILITY ANALYSIS.

From the noise criterion (9-38) it is easily understood that the complete extractability of a system model from noisy data is critically dependent on the value of the smallest significant singular value associated to the impulse response of the undisturbed system. Therefore we analysed the behaviour of this singular value for a test set of representative impulse response sequences:

The unit energy impulse response.

Any sampled impulse response of a strictly proper, time invariant finite dimensional linear system is a linear combination of sequences

$$h(\eta;\alpha,\psi_s;A,\psi_o) \stackrel{\Delta}{=} A\, k^{\eta-1}\, \alpha^k \cos(\psi_s k + \psi_o) \qquad (9\text{-}42)$$

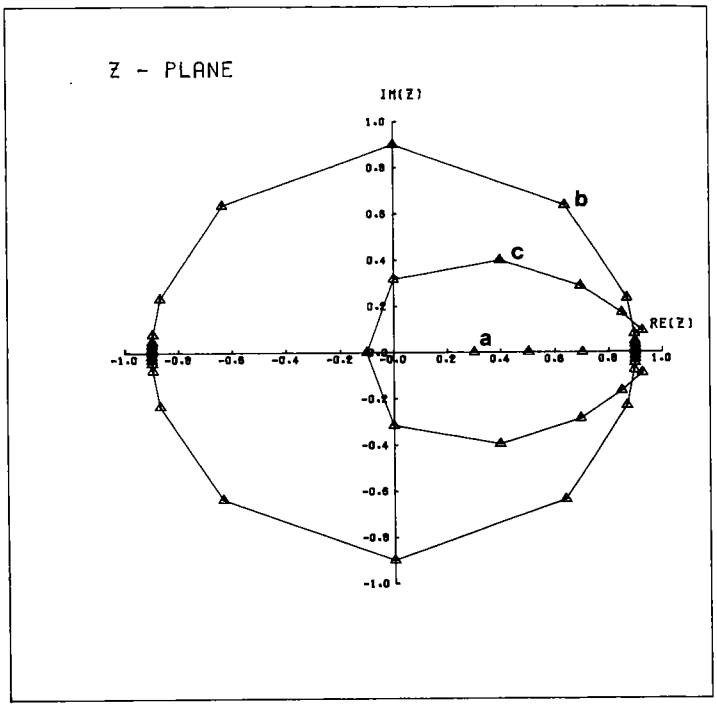
For these sequences the scale factor A is chosen so that the total energy in the sequence is unit. For such sequences of equal energy, it is meaningful to compare the SV's. This was done for a set of test functions, the poles of which are represented in Fig.(9-2a). Fig.(9-2b) illustrates the wide variety in the analysed set with a representative selection of associated IR's in the time domain.

From extensive and systematic simulations on this set of test functions, we came to the following conclusions.

a. Monopole monovariable multiplicity- and poleposition effects [101, 107].

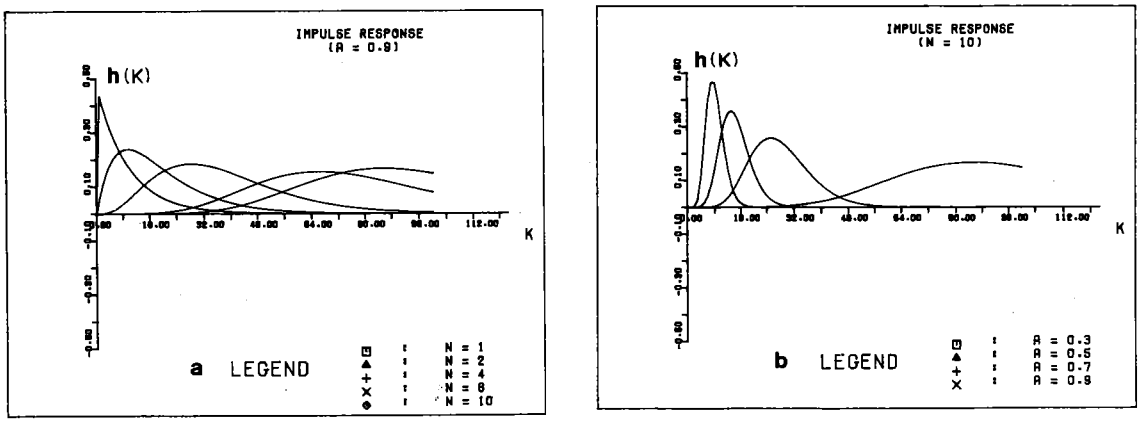
1. For a fixed number of available samples, the smallest significant SV of a unit energy mode, associated to one multiple pole, is a function of the damping α of that mode, and a function of the multiplicity η of the pole.

One clearly sees in Fig.(9-3), that for small values of α, as well as for values of α tending to unity, the realizability of high-multiplicity modes decreases significantly. The reason for the former is that when α tends to zero, the signal dies out too quickly for a decent realization, whereas for the latter, the response is so slow, that the used finite number of samples is insufficient to model the mode decently.



a.Real poles (no oscillation, varying damping)
b.Conjugate complex pole pairs of constant damping (varying oscillation).
c.Pole pairs associated to one signal, sampled with increasing frequency.

Fig.9-2a: An overview of the pole positions, analysed in the feasibility analysis.



a.One real pole at z=0.9 , for multiplicities N=1 up to N=10
b.One real pole of multiplicity N=10, for a damping a=0.3 to a=0.9

Fig.9-2b: Some mono-pole impulse responses. ($N \triangleq \eta$; $A \triangleq \alpha$)

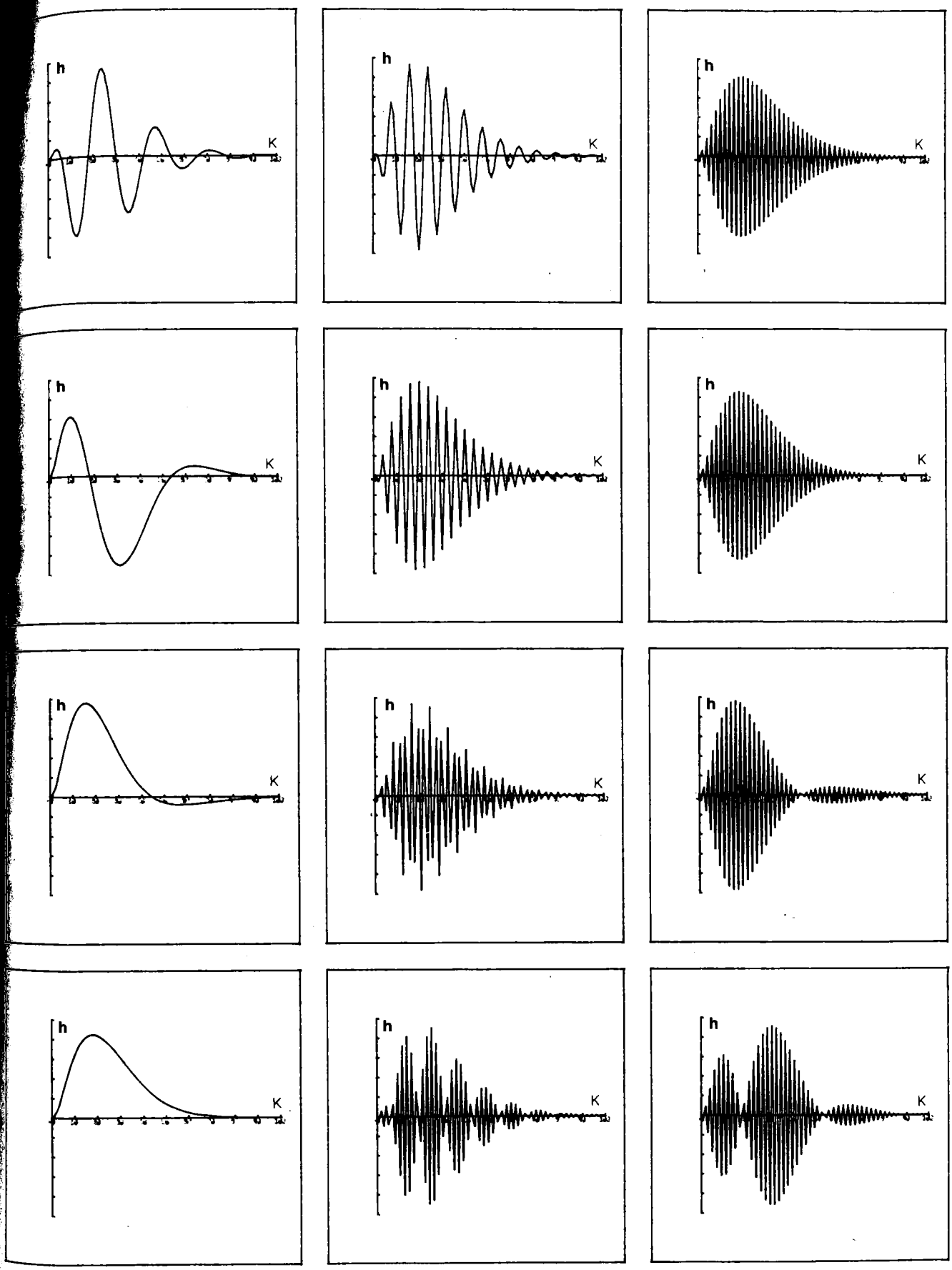


Fig.9-2b (cont): Some more examples of analysed impulse responses.
Fixed degree and damping, varying sample angle ψ.

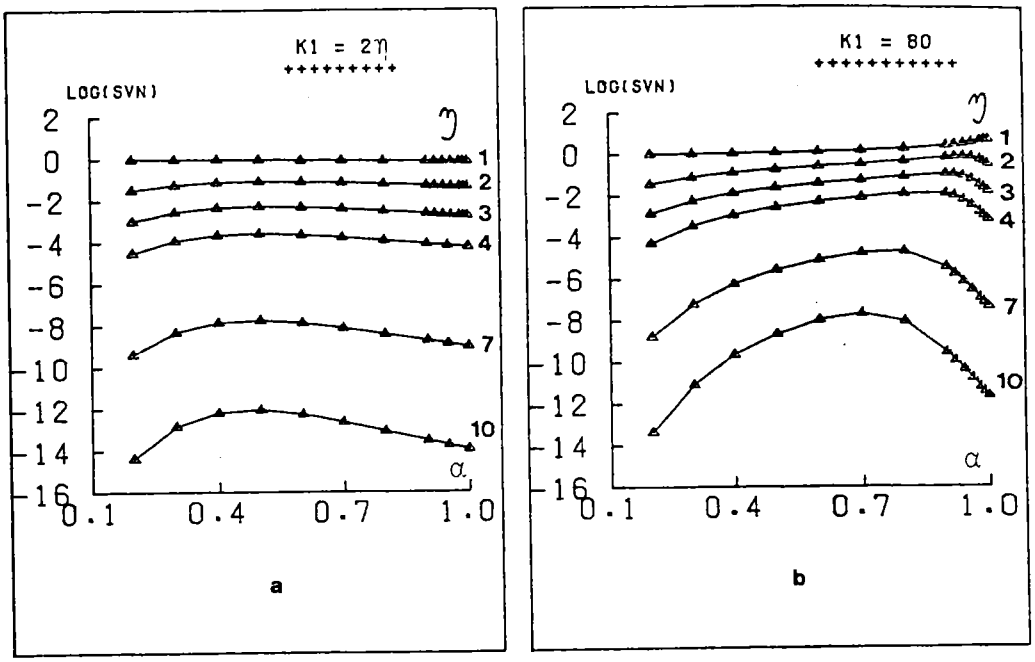
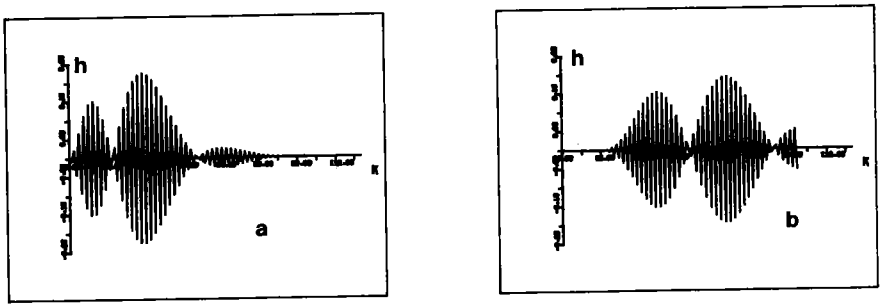


Fig.9-3: The smallest significant SV of a monovariable monopole unit energy mode as a function of α and η, when using:
(a) $K_1 = 2\eta$ samples in the realization.
(b) $K_1 = 80$ samples in the realization.

Increasing the number of samples in the realization will improve the realizability as long as there is sufficient energy in the additional samples taken into consideration. As an example, Fig.(9-4) shows pathological impulse responses, heavily degraded not only by the multiplicity effect, but also by a proximity effect. These examples make it intuitively clear that a higher number of samples will make the realization far more reliable. This is quantitatively confirmed by Fig.(9-5), relating the evolution of the singular spectrum [for the example of Fig.(9-4a)] as a function of the number of used samples, and the associated accuracy (in two-norm for the impulse response) obtained in double precision (14 digit) arithmetic.



a: h(3;0.9,175°;A,0)

b: h(7;0.9,175°;A,0)

Fig.9-4: two pathological impulse responses, heavily degenerated by pole multiplicity and proximity effects.

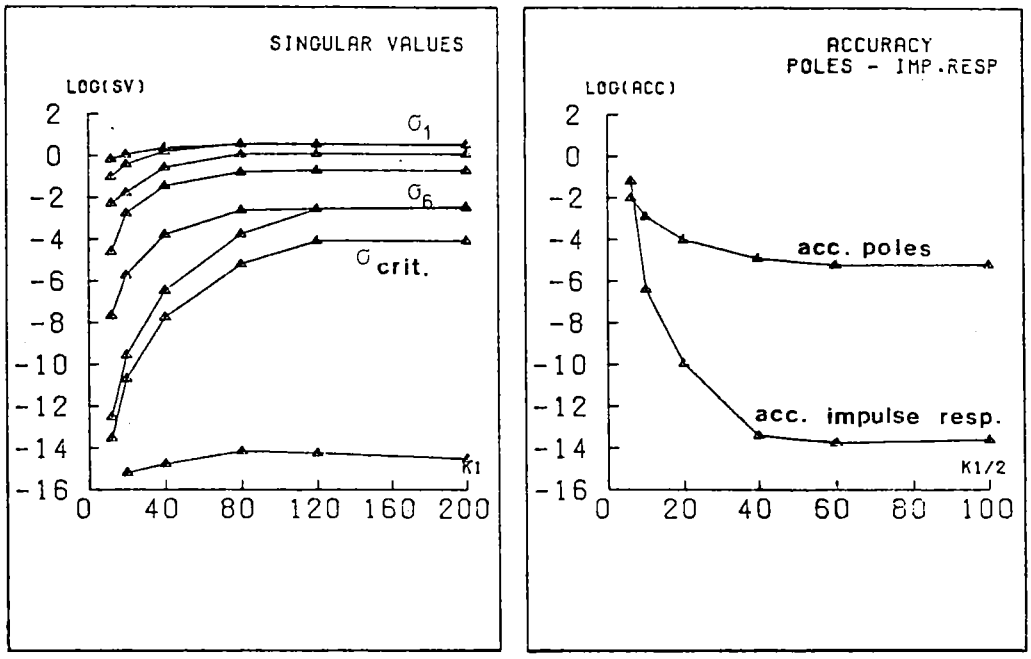


Fig.9-5 : singular spectra for increasing numbers of samples, used in the (double precision) realization of example 9-4a.

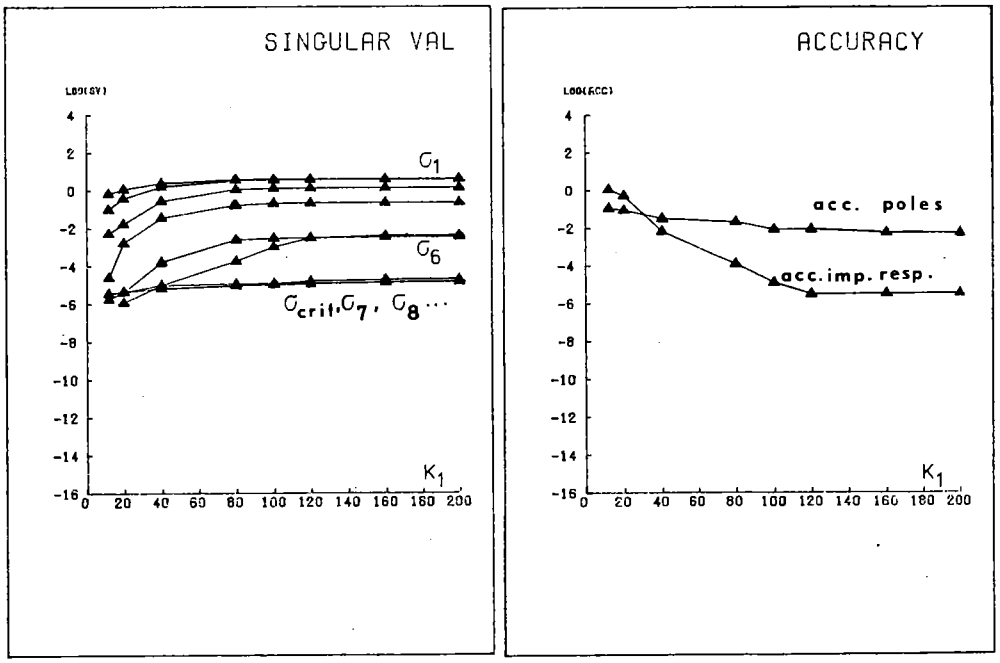


Fig.9-6: same singular spectra as in Fig.(9-5), but now for a realization under white noise disturbance with a SNR of 0.000001. The smallest SV "emerges" from the noise from 40 used samples on.

The improvement obtained by increasing the number of samples from 12 (the theoretical minimum) to 200 is about ten orders of magnitude. One clearly sees the singular value σ_6 grow from 2.95 10^{-13} to 3.22 10^{-3}, and the accuracy of the impulse response and poles of the realized SSR improve accordingly. Fig.(9-4) is also a good example to understand why recursive or G.E. based algorithms will never be able to guarantee a reliable solution (recall the arguments of 9.B.a).

In the case of noisy data the smallest singular value is observed to emerge in a similar manner above the noise level from a certain number of used impulse response parameters on Fig.(9-6):

2. For a fixed number of samples, the singular spectrum of a unit energy mode associated to one pole (of possibly higher multiplicity) is independent of ψ_s

This statement was verified by comparing the singular spectra associated to configurations of poles that are related by a rotation around z=0. The singular spectra of such configurations are identical, provided the energy of each of the truncated signals is a constant. Small variations will occur when varying ψ_o, but the variation is not dramatic.

b. Multipole monovariable proximity effects.

Systems that have different poles in the unit circle of the z-plane, can be analysed as superpositions of one-pole systems, as long as the poles are not too close to each other.

When two poles of multiplicity η_1 and η_2 are near each other, then this $\eta_1+\eta_2$ dimensional system behaviour is close to the behaviour of a system of which the degree can vary from $\text{Max}(\eta_1,\eta_2)$ to $\eta_1+\eta_2$, depending on the respective ψ_o values. Fig. (9-7) shows how a mode may disappear out of a (very slowly) time varying system, when the poles of two modes with identical ψ_o come closer until they finally coincide.

c. Multivariable systems.

1.The input and output dimensions can make the MBH matrix considerably larger for a same number of samples. For a large class of systems however (with 1...5 inputs and/or 1...5 outputs, and a system degree up to 10), multivariable techniques can improve the controller design significantly, and still require no excessive computing power with our algorithm ASVD.

2. Proximity of system poles cause the same (nefast) effects as in the monovariable case only if the these poles occur in interacting signals. Poles associated to orthogonal SIR's do not show proximity effects.

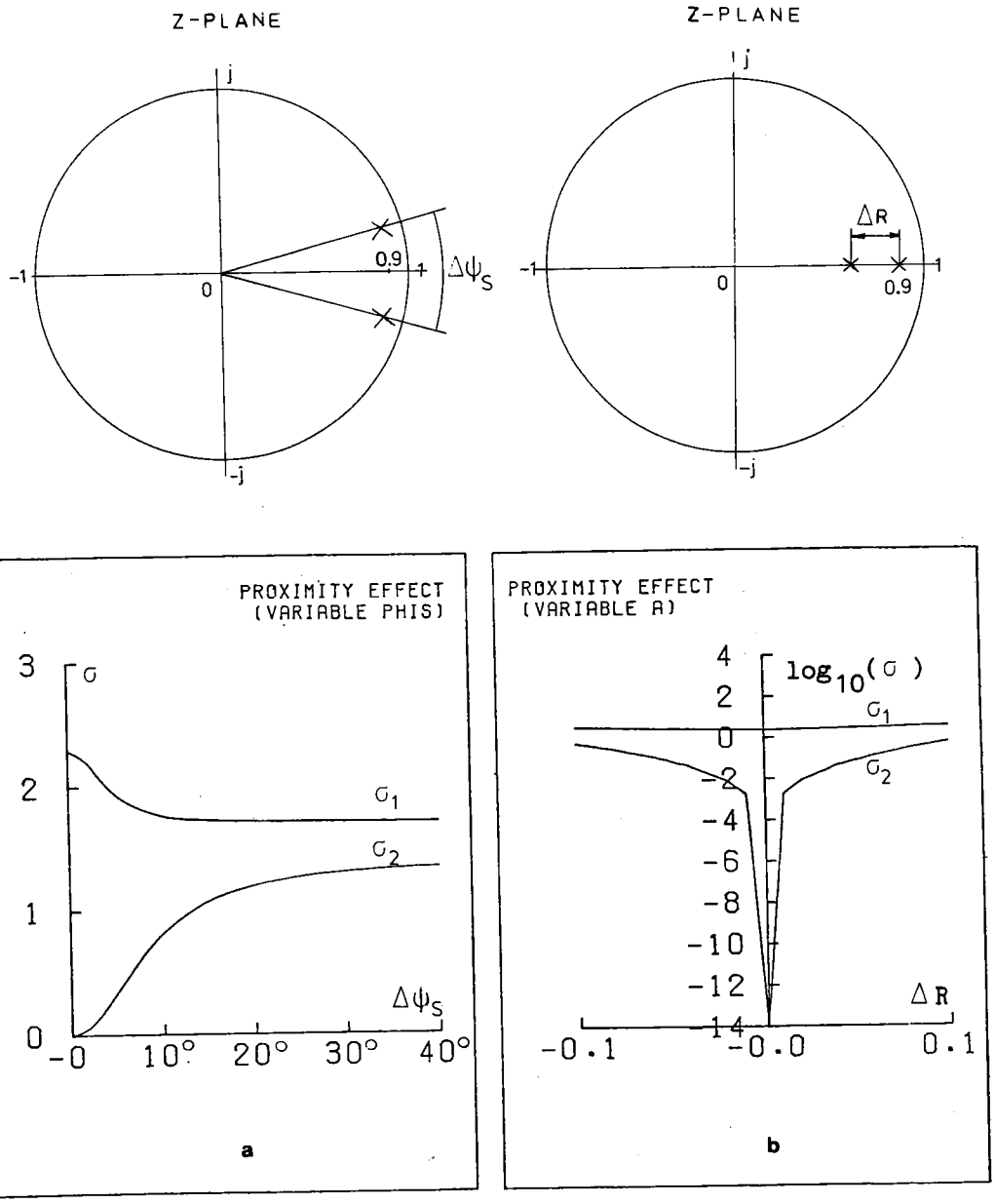
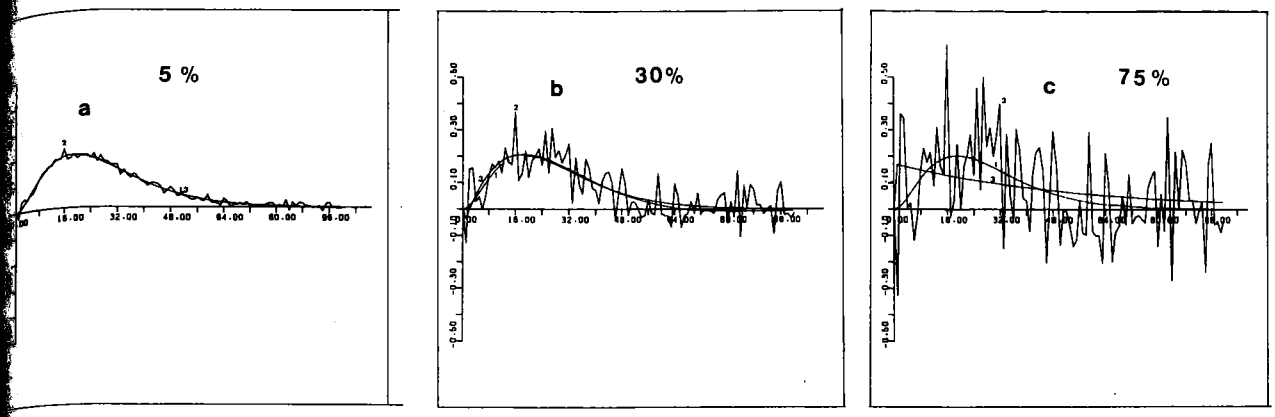


Fig.9-7: Proximity effects for poles coming closer
(a) along a circle of constant radius.
(b) along a radius of constant ψ_s

E. EXAMPLES.

The implementation of the balanced realization (9-28) together with the degree criterion (9-38) leads to a quite reliable algorithm for most applications. It has the additional advantage that it can be totally inplemented using the ASVD algorithm described in chapter 5. Due to the (Block) Hankel structure of the MBH matrix, the resulting algorithm AREAL can be implemented in a storage-, and computation time efficient manner, the real advantage of which fully appears when it is imbedded in an adaptive identification scheme, as described in chapter 10. The now following (early) results and examples are restricted to one-shot realization. They were obtained with the classical SVD routine from the NAG subroutine library. The main purpose is to illustrate the power of the algorithm, and to point at some typical effects that are frequently observed.

a. Example (9-1): Fig.(9-8)



1: Clean signal.
2: Noisy signal, fed to the algorithm.
3: Recovered signal, obtained after realization.

Fig.9-8: Exact, available and realized IR for a monovariable system.

The impulse response of a monovariable monopole system of degree 3, with a triple real pole at z=0.9 was realized from simulated measurements, poisened with white noise levels of 5, 30 and 75 percent. A noise level of λ percent denotes that the standard deviation of the disturbing (white) noise sequence is λ % of the maximum value of the exact impulse response. For increasing noise levels, the algorithm extracts reduced models of decreasing degree. 100 samples were involved in the realization, and an MSNR of 2 was required. Clearly:

- Up to a noise level of 5%, the degree as well as the model are recovered satisfactorily.

- For a noise level of 30%, one mode has to be dropped, and a system model of (reduced) degree 2 is extracted. The impulse response of the realized model however still closely reconstructs the original (undisturbed) IR.

- For 70% of noise finally, the algorithm can extract no better than a first degree system. Notice however that the recovered mode has a pole, that is still meaningfully related to the original system.

An important observation is, that when the realization procedure was forced to use system degrees higher than those suggested by the algorithm, oscillating, and sometimes unstable modes were observed in the resulting models.

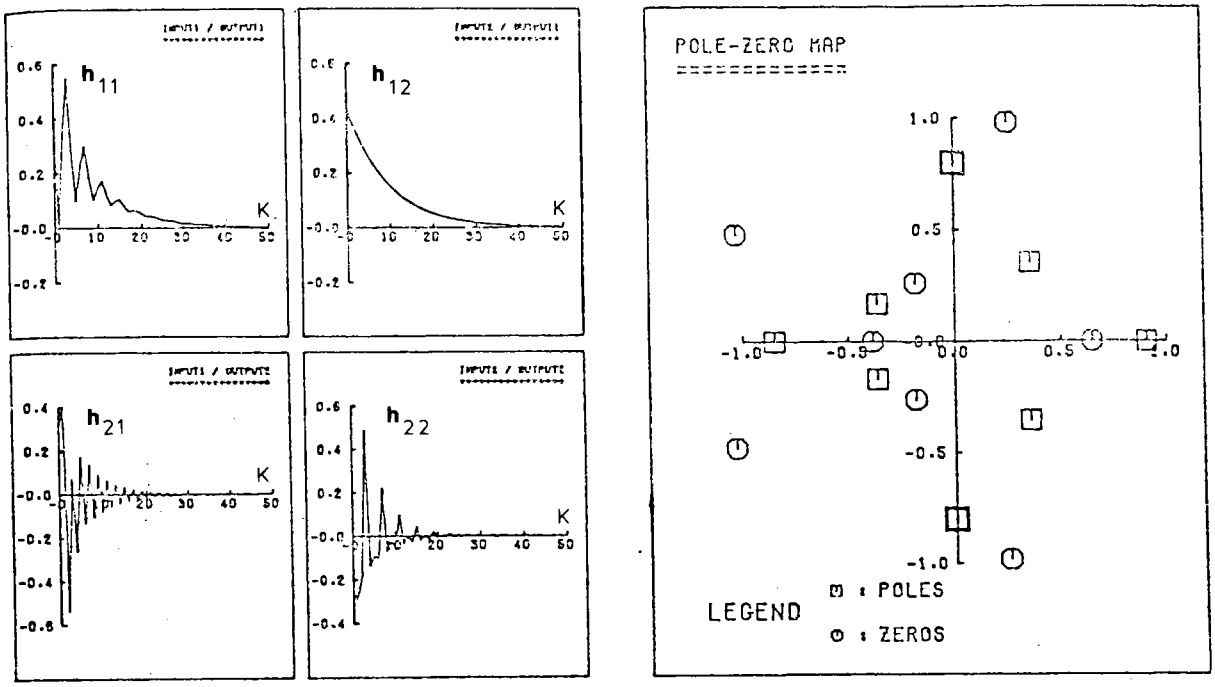


Fig.9-9: IR and pole zero configuration for a (2x2) multivariable system.

Noise level (%)	ACCURACY (%) (50 MP)				
	[1,1]	[1,2]	[2,1]	[2,2]	
0	$1.1\ 10^{-12}$	$1.3\ 10^{-12}$	10^{-12}	$1.3\ 10^{-12}$	a
0.05	0.02	0.03	0.008	0.02	
0.5	0.22	0.3	0.08	0.18	
5	2.0	3.3	0.8	1.3	
20	9.3	15	6.8	9	

Pole Nr.	No Noise	0.05 %	0.5 %	5 %	20 %	
1	0.9∢0	0.899996	0.899957	0.89979	0.89737	
5	0.85∢180	0.85013	0.85098	0.85045	0.8683	
2	0.5∢45	0.5003∢44.96	0.49∢46.6	0.58∢45.4	———	
2	"	"	"	"	———	
4	0.4∢155	0.39∢150	———	———	———	
4	"	"	———	———	———	
3	0.8∢90	0.79998∢90	0.8004 ∢90.03	0.8059 ∢89.83	0.768 ∢85.8	b
3	"	"	"	"	"	
3	0.8∢90	0.79999∢90	0.7989 ∢90.06	0.7972 ∢91.04	0.833 ∢94.5	
3	"	"	"	"	"	

TABLE 9-1: Accuracy of (a) the realized impulse response
(b) the associated pole-configuration
for increasing noise levels on the given data.

ample (9-2): Fig.(9-9)

ltivariable result is shown in Fig. (9-9) and Table (9-1). 50 samples used in the realization, and increasing noise levels were imposed. ce that for increasing noise levels, those poles of the realized model become inaccurate first, also disappear first. These are poles asso- ted to weaker modes in the impulse response: they are most sensitive to inaccuracies, and from a certain level of noise become so unreliable, it is best to discard them. As one can appreciate, this is elegantly automatically achieved by our degree criterion.

Example (9-3): Fig.(9-10)

close the section on examples with a 4-input, 3-output system of degree known as the example of ROVEDA [66]. The original (exact) system has a gular spectrum for the MBH matrix, over a range of 7 orders of magni- e. The original data were spoiled with 5% white noise, and recovered h an MSNR of 3. Notice that the low-energy entries of the impulse res- se are seriously affected, whereas the high energy entries are quite ely recovered (notice the different scales for different entries!). resulting system has a reduced degree 7, and the poles of the recove- system are close to 7 poles of the exact system. 5 modes are unrecove- ble from the noisy data.

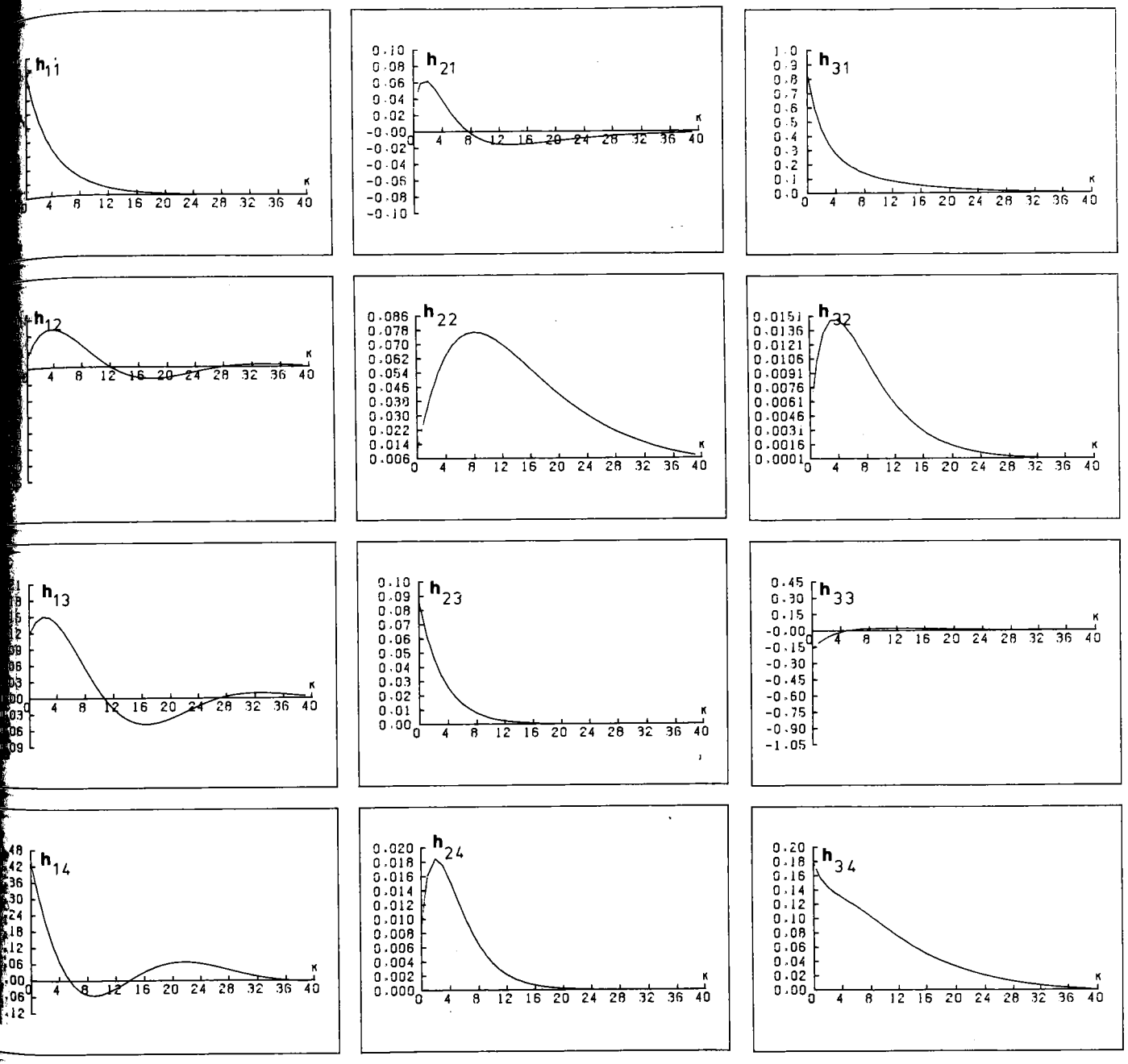


Fig. 9-10a: ROVEDA'S example. Clean impulse response of a 4-input 3-output system. The system degree is 12.

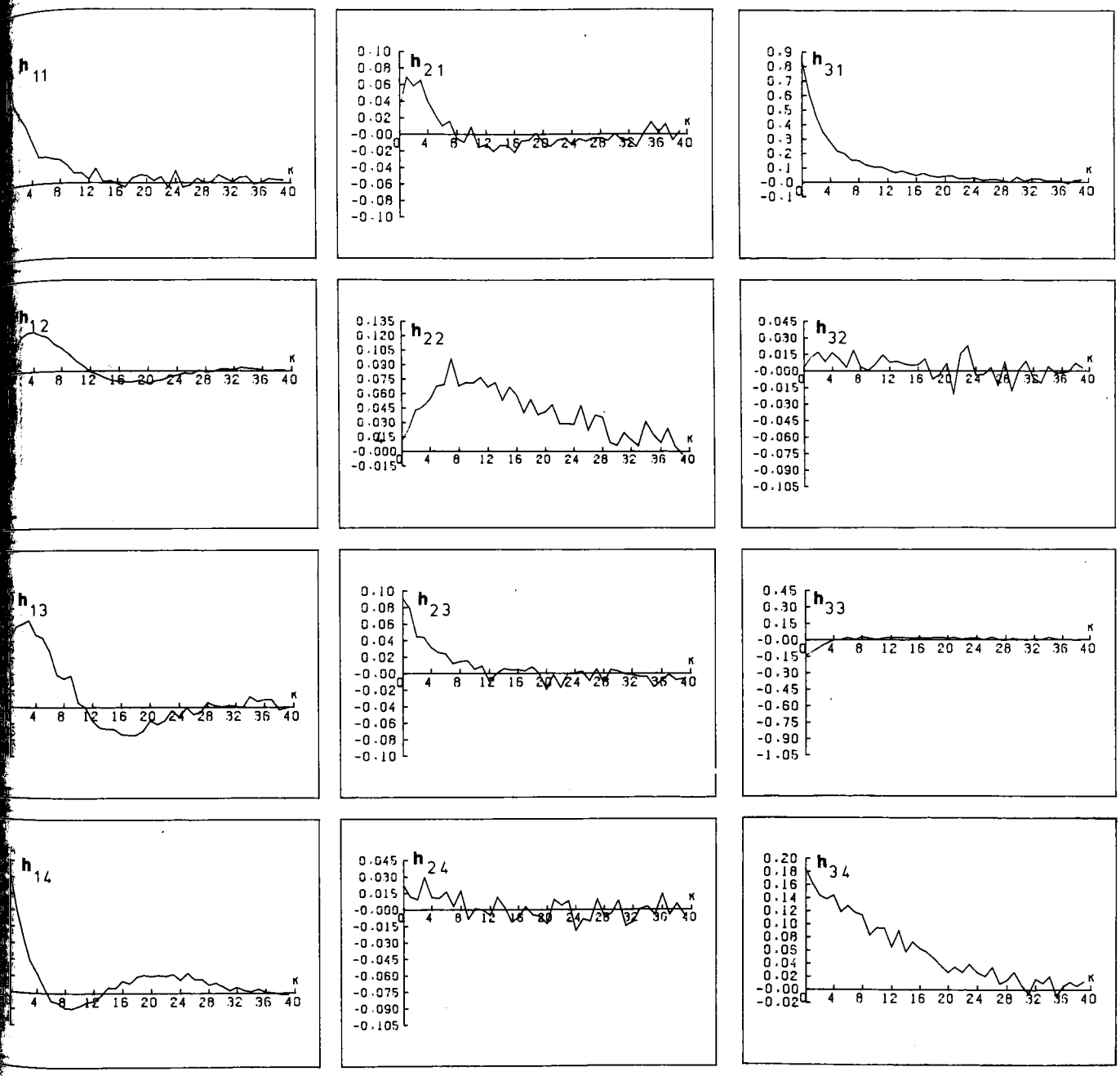


Fig. 9-10b : ROVEDA'S example. The same impulse response as in Fig.9-10a, but poisened with 5% noise.

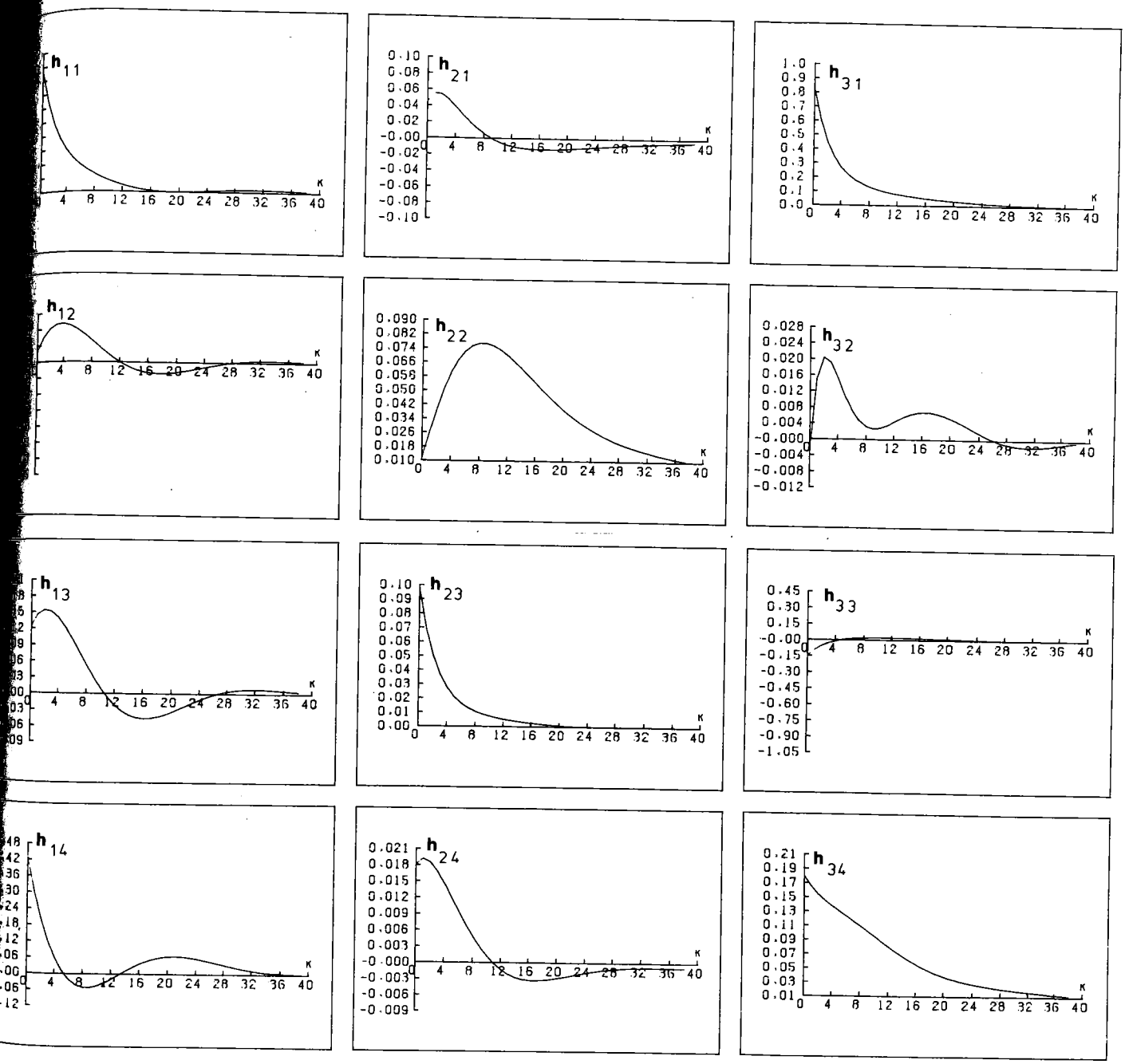


Fig.9-10c: ROVEDA'S example: impulse response, associated to the realized State Space Representation. The (automatically) reduced model degree is 7.

Chapter 10

ON ADAPTIVE ON-LINE IDENTIFICATION OF A LINEAR STATE SPACE REPRESENTATION FROM MEASURED INPUT OUTPUT DATA

The purpose of this last chapter is mainly to put the ideas developped in the previous pages into a kind of unifying perspective, by presenting the identification procedure that has served as a guide and test-object all through my journey in numerical puzzle land, and on which the practical value of most encountered concepts and algorithms have been tested.

The reader should therefore not expect a detailed and formally rigorous development of one brandnew and fully analysed algorithm. Rather, he will find a report on the main results and on the experience gained from a host of apparently disparate simulations on reliability of adaptive State Space Representation identification for multivariable and possibly time varying linear systems.

An essential step in nearly any technical as well as nontechnical problem analysis is the extraction of a mathematical model from a set of available measurements. The type of model and the objectives and methods used may however considerably vary from one discipline to another. In the domain of process control, the extraction of a formal description of the internal and/or external dynamical behaviour of the plant under study is usually the crucial step in the design of a regulating unit. Unfortunately, and in spite of exhaustive availability of literature on the subject, this activity is quite often still essentially artisanal. Reliable procedures for model extraction are rare and usually dedicated, hence restricted to a very particular type of application.

Even within the class of linear systems the problem is quite tough when multivariable (i.e. multi-input, multi-output), possibly time varying systems of totally unknown internal structure have to be modelled based on disturbed (noisy) external measurements. When, additionally, this modelling has to be performed in real time and under (regulated) operation, the task is known to be extremely difficult.

And yet, recent developments in the microprocessor domain have emphasized the need and feasibility (from an implementational point of view) for such types of algorithms in the very near future. In an automatic digital adaptive controller for instance, a reliable on-line modelling algorithm will be a vital module. To be of any practical significance for such a device however, the algorithm should be totally autonomous and highly reliable, which means that it should be able to extract the desired system model directly, i.e. without any external (human) interference from the measured input-output data, and in such a way that it accounts only for system effects, that are sufficiently distinguishable from the noise. Moreover, it is in general not acceptable to have special requirements imposed on the applied input signal, as this signal is usually dictated by other requirements (e.g. control requirements in the case of on-line identification of an industrial process) or simply impossible to activate or disturb (e.g.

the identification of an observed natural phenomenon). Finally, if the model is to be used for advanced control or any another purpose with a comparable level of sophistication, it is highly desirable to come up with a model in State Space formalism, because this approach allows for the most elegant and sophisticated further analysis and design of derived (multivariable) observer and controller devices.

We conclude that it is challenging to design algorithms that can autonomously and reliably model multivariable, possibly slowly time varying systems by a linear(ized) State Space Representation.

As mentioned before we do not claim, and it is not the ambition of this concluding chapter, to present such an algorithm in fully worked out and thoroughly optimized form. The coordination of all practical aspects of such an implementation are far beyond the scope of this mainly conceptual thesis. But for many of the above mentioned aspects, the SVD concept has proved to be of great significance and the associated insight has been the main guide to quite succesful implementations of some crucial steps that are essential to the kind of universal linear identification device we would like ever to see realized.

The chapter is organized as follows:

To give a flavour of the complexity of the problem environment, we will first spend a section on the general scheme of an adaptive controller for time varying multivariable systems, and briefly discuss the type of peripheral problems that have to be solved before succesful operation can be reasonably expected.

Then we will focus the attention more on some details of the identification, and discuss some of the fundamental issues associated with the problem statement itself and the proposed solution scheme.

Finally we will show some direct applications of the concepts and algorithms discussed in this thesis, in an overview of results obtained from a set of significant simulations, which should convince the interested reader of the attractivity and the intrinsic power of the approach.

A. ON A GENERAL-PURPOSE, MULTIVARIABLE ADAPTIVE CONTROL SCHEME.

a. A general configuration.

Fig. 10-1 shows the general configuration of a system regulated by a multivariable multipurpose controller.

Such a controller measures the input-output behaviour (u^*,y^*) of the plant as (u,y). From those measurements, a control signal c is derived, which is sent to the actuator. A good controller should be able to exploit available general plant information α, and should be able to adapt the control action according to general plant behaviour requirements β. Very often the requirement is mainly that an external reference signal r be followed as closely as possible by the output. Usually, the weakest point in such a configuration is the (non-)availability of a reliable system model. The available a priori information is often incomplete, qualitative, and/or difficult to translate into formal and quantised model parameters. There-

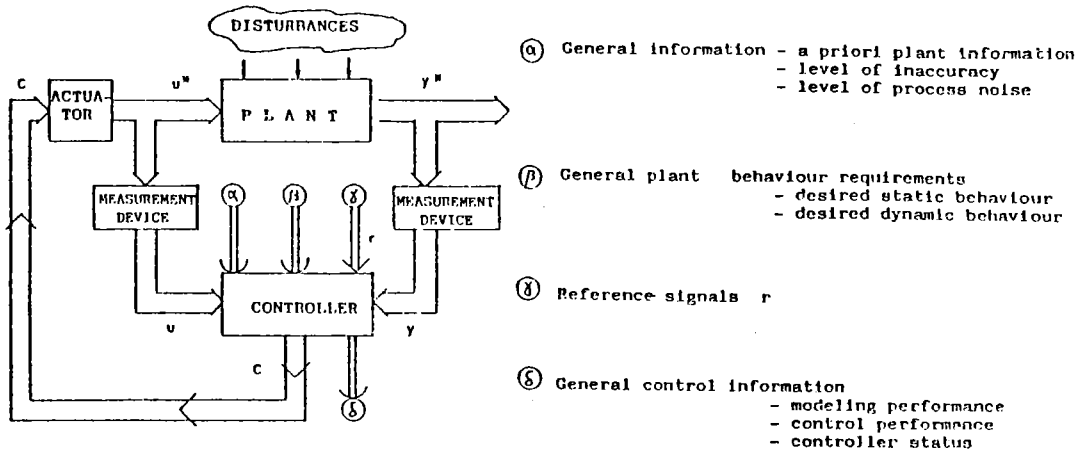


Fig.10-1: Typical configuration of a general purpose control scheme.

fore, in many cases, a controller that does not need such external information (except for some very fundamental properties such as causality, finite dimensionality, linearity,...) is a welcome device. Such a device will, explicitely or implicitely, execute three tasks (Fig. 10-2):

a. identification of a plant model S
b. estimation of the plant state x
c. computation of a control signal c based on S,x,u,y,r,α and β.

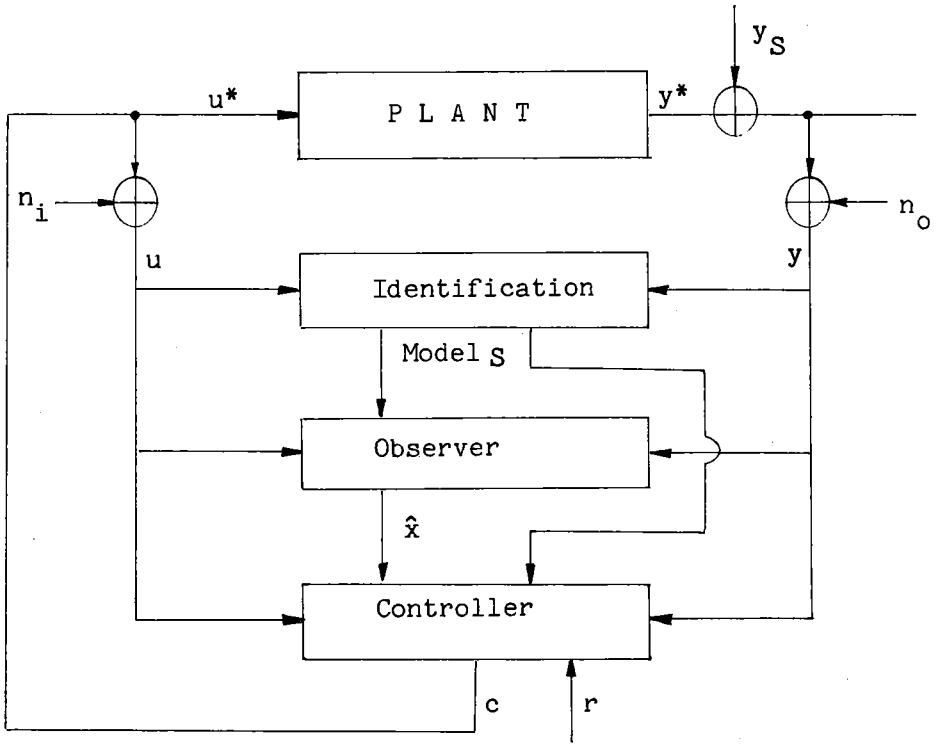


Fig.10-2: An explicit scheme of the three main tasks in a general purpose automatic controller: identification, state estimation and control.

In the ideal case, the major (and nontrivial!) contribution of the process engineer will be to connect the input, output, and control devices at control-significant points of the real system; to give a safe but realistic estimation of the levels of measurement inaccuracies and process noise; and to define an appropriate and motivated quantised description of what is understood by "plant behaviour of sufficient quality". This information may be fixed once and for all in the design stage, or may depend on the particular operational situation, which can then regularly be computed, or even transmitted from a central supervisory computer to a relatively independent subcontroller. Problems at this hierarchical level are fail-safe and efficient information transmission, elegant data centralization and displaying, and flexible overall organization.

b. An identification module.

In the above general context, Fig. 10-3 gives an outline for a general identification scheme for systems with significant deterministic as well as stochastic (u,y) plant behaviour. A similar scheme can be used (if necessary) for the (u,c) actuator behaviour.

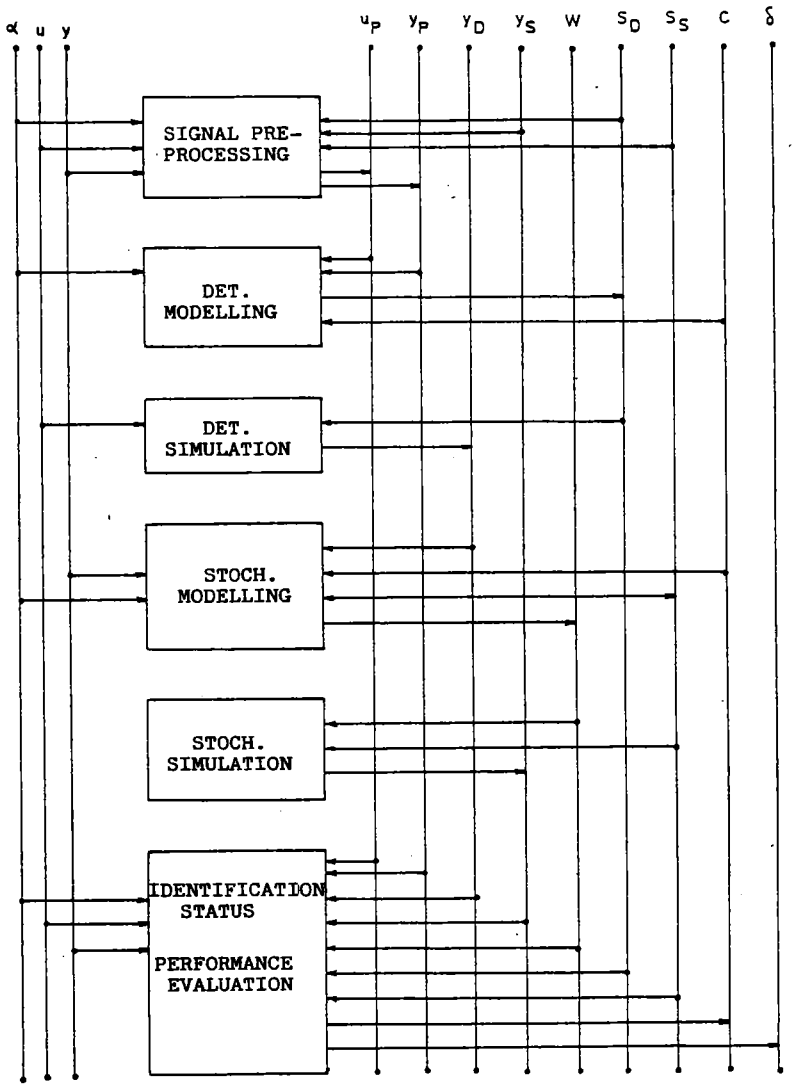
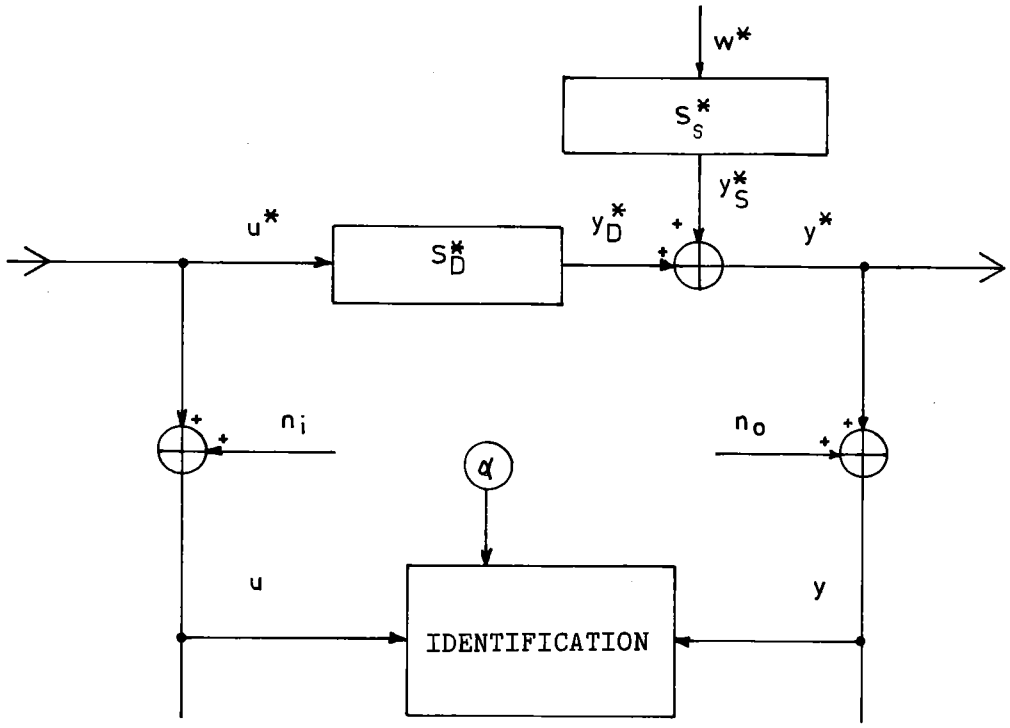


Fig.10-3: An identification scheme for a general purpose adaptive controller.

Notice that the deterministic modelling is performed on preprocessed signals. This preprocessing may include (frequency-)band filtering, based on the general information α; extraction of the stochastically modelled part of the behaviour; and extraction of the "tail behaviour" of the deterministic model in use (see later).

For later (more formal) purposes, Fig. 10-4 gives an overview of the most important definitions and notations used in the identification context.



u^*, y^*	real system input and output.
S_D^* , y_D^*	real system deterministic model and output.
S_S^* , w^* , y_S^*	real system stochastic model, generating white noise and (coloured) output.
n_i , n_o	input- and output measurement errors.
u , y , α	input, output and general information, available for the identification.

Fig 10-4: Definition and notation of the parameters and signals involved in the identification

The remainder of this chapter is restricted to the modelling of the deterministic part in such a dynamic (u,y) relationship. Fig.10-5 shows our two-step approach AIDENT with tail correction feedback, valid for causal, finite-dimensional discrete-time stable-system modelling [104-105].

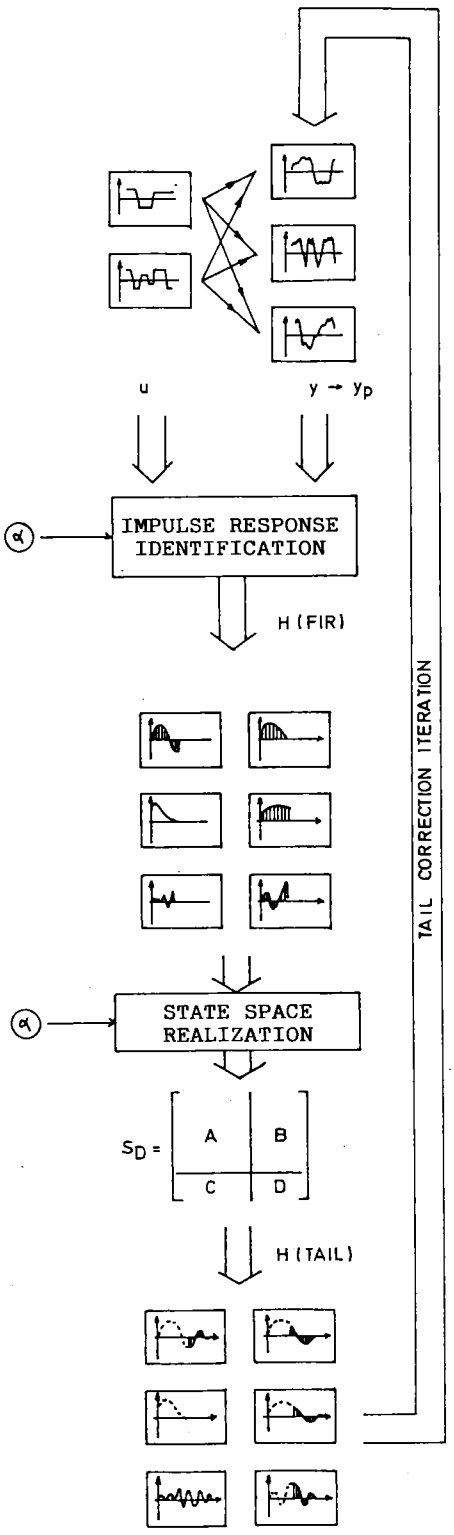


Fig.10-5: Scheme of AIDENT.

The algorithm's structure is roughly as follows: In a first step, the systems impulse response is obtained from the (preprocessed) input-output data by a Generalized Deconvolution (GD) procedure. In the second step, a State Space Representation (SSR) is extracted from the obtained impulse response via Realization. Now in step one, only a finite number of impulse response samples can be evaluated, which introduces a so called truncation error, that may be quite considerable when the systems impulse response is only slowly decaying. But the SSR model obtained in step two, represents a complete impulse response (over an infinite interval). Separating the effect due to the "tail" of this impulse response from the short term (original finite interval) effect, one can hope to compensate the available output data for the long term ("tail-") effects of the real system's impulse response. When the approximate SSR describes the long term behaviour better than a zero sequence does, the compensation of the output data, based on the approximate model, will lead to an artificial output sequence that is closer to the finite impulse response behaviour of the real system than the original data. Hence a renewed GD and Realization operation on the new pair of sequences should lead to a new SSR, that is closer to the real system model than the first approximation. When this is the case, further iteration in the same manner should eventually compensate the given output data completely for the long term (tail) behaviour. Hence in the GD there should be no more truncation error effect, and the only source of error should be the measurement inaccuracies and process noise.

The advantages of the two step approach are important. First of all, the procedure allows multivariable modelling with no difficulty. Moreover the first step is linear in the parameters, which allows easy (adaptive) updating, correction and iteration by simple superposition. Finally, no degree nor structure decision has to be taken until step two. And for this realization step the reliable degree and structure decision rules of chapter 9 can be used, based on the impulse response itself,

the noise level on the data, and a required minimal signal to noise ratio for the model. In this way, all requirements of the previous section are met. A drawback at first sight is certainly the required computing power. The development of the algorithms ASOLVE and ASVD in chapter 5 and 6 have been primarily inspired by the need to counter this objection. With the storage and computation-time savings obtained in these algorithms, quite complex systems can be identified with nowadays microcomputer units. Besides the general organizational problem, an essential issue in the context of a FULLY automated implementation on such a device is the global reliability, and of course the global convergence of the entire algorithm, as a function of a number of parameters that are still free to choose, such as the number of impulse response samples used in the deconvolution, the number of iterations per real time step, etc. A complete answer is not available. Some obtained partial results however, are quite encouraging. We will therefore, after this short and rough introduction, describe in some more detail the concrete implementation of AIDENT, and discuss some of the obtained results. Before we do so however, we think it is the time for a short interruption:

A short note on directly parametrized identification.

The reader may not really appreciate the above cited advantages, find the proposed scheme rather elementary in concept, and computationally involved compared to the so much more elegant directly parametrized identification. This objection is very common, and therefore let us, among the arguments cited above, analyse in some more detail the aspect of energy based structure determination versus direct parametrization in a context of fully automated identification. Suppose a system of unknown degree n is to be identified, and a transfer function of degree N is proposed, in the hope that coefficients of degree N>n vanish if necessary. Suppose the algorithm would come up with the following result:

$$H(z) = \frac{H(0).z^{N} + H(1).z^{N-1} + \ldots + H(N)}{z^{N} + 0.z^{N-1} + \ldots + 0} \qquad (10\text{-}1)$$

If the impulse response is quickly vanishing, this transfer function (10-1) can very tightly match the input-output data when the H(k) are the systems impulse response samples. Indeed, in this case (10-1) represents the Finite Impulse Response, Moving Average approximation of the system in its first N+1 samples, and the H(N+k), k>0 are negligeable. In fact, by choosing N sufficiently high, it is always possible to match an H(z) of the form (10-1) arbitrarily close to the available data of any stable linear system behaviour. The point to see now is, that whatever the real pole configuration of the system is, the pole configuration of the model is fixed: an N-tuple pole at z=0. Hence no pole information at all is present in the denominator of the identified model, and there is no need to explain what could be the consequence of using the denominator of this model for feedback control purposes. Now one could conclude that such a result could indicate that the considered system is critical for identification, or that its pole information would be nearly impossible to extract from the given data. This is not so: the pole information IS present in the identified model... in its

numerator coefficients! Indeed, these coefficients samples of the impulse response and can be perfectly realized into a State Space Representation whenever $N > i_c + i_o + 1$. Notice that the tight matching of the input-output data fallaciously seems to confirm the originally proposed degree N, as the only non-vanishing coefficient in the result is the coefficient of highest degree, even for system models of very low degree. Notice also, that the correct model could as well have been identified, in which case the entire pole information would have been present in the denominator, and no pole information at all would have been present in the numerator coefficients. A conclusion of this short analysis is that a priori choice of structure may lead to undetectable, hence unrecoverable loss of control on the identification reliability of essential system properties. The forward sensitivity and possible total loss of pole information in the presented example are essentially due to the mere fact of choosing a (wrong) model structure. This phenomenon is to be feared in each algorithm based on formal algebraic representations and/or manipulations, and hence is an inherent danger for any automatically, hence formally parametrized identification. And this is essentially why the author has a natural tendency to promote algorithms, in which each step has its direct (pseudo-) physical interpretation. Similar arguments can be set up in favour of algorithms that are multivariable in nature, rather than forced (and formal) extensions of essentially monovariable procedures; and in favour of algorithms that include the numerical aspects as an essential part of the problem formulation, because this is the only way to keep a permanent and guiding numerical control on the developped solution algebra; and finally in favour of geometrically inspired algorithms, simply because one can intuitively SEE the solution long before one is able to derive it algebraically, which allows a much tighter conceptual control on the algorithm's development than with any purely formal method.

The author believes that the proposed identification scheme nowhere essentially seems to obstruct these aspirations, which should become apparent from the subsequent pages on the results obtained so far.

B. A SOME MORE DETAILED DISCUSSION OF SSR IDENTIFICATION.

a. On Generalized Deconvolution.

It is well known that the "pure" (i.e undisturbed) output of a linear system can be obtained as a convolution of the "pure" input signal with the systems impulse response. With the notation of Fig 10-4, and the usual assumptions of (temporary) time invariance, causality and linearity, such a "pure" m-input, l-output multivariable system behaviour, started up at t=0 with zero initial state, is represented in matrix form as

$$\begin{bmatrix} y(T) & \dots & y(0) \end{bmatrix} = \begin{bmatrix} H_D(0) & \dots & H_D(T) \end{bmatrix} \cdot \begin{bmatrix} u(T) & \dots & \dots & u(1) & u(0) \\ u(T-1) & \dots & \dots & u(0) & 0 \\ \dots & \dots & & 0 & 0 \\ \dots & u(0) & & 0 & 0 \\ u(0) & 0 & \dots & 0 & 0 \end{bmatrix} \quad (10\text{-}2)$$

Notice the (often not fully realized) fact that Eq.10-2 is a collection of l sets of T+1 linear equations, each in m.(T+1) unknowns. This implies that as soon as the number of inputs is larger than 1, these sets of equations

AS SUCH are underdetermined, hence HAVE NO UNIQUE SOLUTION, even for very high T.

When the system has a finite state dimension N, then it also has a finite transfer degree n<N, and a finite Autoregressive Order r<n [46,47, also 36 or 103]. If and only if an upperbound is known for one of these structure parameters, it is possible to determine uniquely a Transfer and/or a Minimal State Space representation from the available measurements, provided T is sufficiently high. The reason for this required bound on the system complexity is, that a finite system structure guarantees redundancy in the impulse response samples (markov parameters in general) that can be described in an autoregressive, explicit formalism. A fixed, finite number of parameters can then be guaranteed to describe completely the exact system behaviour, and when T is sufficiently high, the sets of equations become well-, and finally overdetermined, thus guaranteeing a unique and exact solution in the case of exact data. Let us remind the disussion in the previous section on direct parametrization, in order not to use this property in a formally tempting, but numerically dangerous immediate parametrization procedure. Rather consider this convolution description first in its generalized form, that includes all the disturbances, represented in Fig.(10-4). The fact that the underdetermination is no fundamental obstacle suffices to continue the algebraic development, but will be used only in a later stage of the derivation:

$$
\begin{aligned}
&\left[\, y(T) \;\ldots\; y(T-M) \;\ldots\; y(0) \,\right] = \\
&\quad \left[\, n_o(T) \;\ldots\; n_o(T-M) \;\ldots\; n_o(0) \,\right] \\
&\quad + \left[\, y^*_S(T) \;\ldots\; y^*_S(T-M) \;\ldots\; y^*_S(0) \,\right] \\
&\quad + \left[\, H_D(0) \;\ldots\; H_D(N) \,\right] \cdot \begin{bmatrix} u(T) & \ldots & u(T-M) & \ldots & u(N) & \ldots & u(1) & u(0) \\ u(T-1) & \ldots & u(T-M-1) & \ldots & u(N-1) & \ldots & u(0) & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & & \ldots & \ldots \\ u(T-N) & \ldots & u(T-M-N) & \ldots & u(0) & \ldots & 0 & 0 \end{bmatrix} \\
&\quad + \left[\, H_D(N+1) \;..\; H_D(T) \,\right] \cdot \begin{bmatrix} u(T-N-1) & \ldots & u(1) & u(0) & 0 & \ldots & 0 \\ u(T-N-2) & \ldots & u(0) & 0 & 0 & \ldots & 0 \\ \ldots & & 0 & 0 & 0 & \ldots & 0 \\ \ldots & & \ldots & \ldots & 0 & \ldots & 0 \\ u(0) & \ldots & 0 & 0 & 0 & \ldots & 0 \end{bmatrix} \\
&\quad - \left[\, H_D(0) \;\ldots\; H_D(T) \,\right] \cdot \begin{bmatrix} n_i(T) & \ldots & n_i(0) \\ \ldots & & \ldots \\ n_i(0) & \ldots & 0 \end{bmatrix}
\end{aligned}
\qquad (10\text{-}3a)
$$

or in obvious shorthand matrix notation:

$$
Y(T,0) = N_o(T,0) + Y^*_S(T,0) + \mathcal{H}_D(0,N).U(T,T-N,0)
$$

$$
+ \mathcal{H}_D(N+1,T).[U(T-N-1,0,0)\;\;0] - \mathcal{H}_D(0,T).N_i(T,0,0) \qquad (10\text{-}3b)
$$

In the above expressions, the measurement errors and whatever remains of the stochastic disturbances are taken to be totally unpredictable (the predictable part is assumed to be compensated already...). These errors have to be accepted as pure inaccuracies on both the matrix and the Right Hand Side of the equation, the only information being perhaps their level. The (1xm) matrix sequence

$$H_D(k) \quad , \quad k=0 \ldots N \ldots T \ldots \infty$$

is the deterministic impulse response of the system, (formally) devided in a finite segment of N+1 samples, and a "tail".

Clearly, eq.(10-3) can only be solved in practice if it is reduced to an approximation of finite extent. Such a restriction only makes sense if the essential information of the solution can be contained in a finite number of samples. From the physical context, it is clear that this is very often a reasonable assumption, that however restricts the applicability of this (finite) deconvolution to the class of STABLE, FINITE DIMENSIONAL linear systems. We will therefore assume that the restriction of the impulse response to N+1 samples in a first instance does not destroy essentially the solution of this equation.

Notice that 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.

For all these reasons we call the operation that extracts a finite section of the impulse response from eq.(10-3) a Generalized Deconvolution.

If all the above assumptions are valid, then the solution of

$$\begin{bmatrix} y(T) & \ldots & y(T-M) \end{bmatrix} = \begin{bmatrix} H_D(0) & \ldots & H_D(N) \end{bmatrix} \cdot \begin{bmatrix} u(T) & \ldots & u(T-M) \\ u(T-1) & \ldots & u(T-M-1) \\ \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots \\ u(T-N) & \ldots & u(T-M-N) \end{bmatrix} \qquad (10\text{-}4)$$

is a valid first approximation of the solution of (10-3). In this way, the GD problem is essentially reduced to the problem of solving a finite set of linear equations (10-4), in which the right hand side is inaccurate due to the neglected output measurement inaccuracy n_o, the process noise y_S^* (or what remains of it after stochastic filtering), and (up to this instant) the tail-truncation; and in which the matrix is inaccurate due to the input measurement inaccuracies n_i. Now except for the tail-truncation error (which will be iterated out: see later), the sources of inaccuracies are unpredictable which suggests a linear least squares solution strategy for this overdetermined set of equations: a typical problem to be solved

with an SVD based algorithm. As the levels of the different sources of inaccuracies are assumed to be known, a condition analysis additionally allows to estimate quite accurately the expected level of inaccuracy on the result, using the sensitivity expressions (3-15) to (3-20), depending on the type of inaccuracies expected (fixed offset, white disturbance...). This is a straightforward application of SVD and conditioning properties.

But the condition of the input matrix can also be interpreted in a quite interesting manner in connection with the concept of "persistent excitation". This is explained in the next section.

b. On input condition and persistent excitation.

In most textbooks and papers on parameter estimation, one mentions the necessity for the input signal to be "persistently exciting". Though claimed to be essential for a succesful identification, this property is usually described quite vaguely as "the property to sufficiently excite all the modes of the system". With respect to persistent excitation, a dirac impulse function and white noise are usually taken as examples of perfect signals and a pure sine wave as an example of very poor signal. Now in the frequency domain both the dirac impulse and white noise happen to have a flat energy spectrum, whereas a sine wave has a very restricted (dirac-) energy distribution. This is then explained in terms of "containing enough frequencies to excite all possible modes of the system", and the "flatness" of the input signal's energy distribution in the frequency domain is suggested to be the criterion for this persistent excitation. Apart from the rather qualitative nature of this approach, the vague motivaton for the proposed criterion does not really satisfy the author, who doubts that the frequency domain is the right setting for the correct formulation of this concept.

An alternative approach is the following:

The Hankel expansion matrix U(T,T-N,T-M) that contains the M+N+1 input samples u(T-M-N) ... u(T) is the matrix of the set of linear equations, from which the impulse response of the system can be solved. This impulse response could be any (block) vector, and hence the matrix is the object to be examined for the reliablility of the solution. As shown in chapter 2 the condition number K(A) of this matrix is the simplest measure for the reliability of the solution that can be obtained from the (Generalized) Deconvolution, i.e. for its sensitivity to inaccuracies on the input measurements, errors on the output measurements, and process disturbances at the output. Therefore, we propose to use the condition number of the input matrix U(T,T-N,T-M) as a measure for persistent excitation.

This measure has a well defined value at each instant, it has a precise numerical and physical significance, and it can be evaluated in a (formally) simple manner. Of course, the exact value of the persistent excitation will depend on N and M, and have a real significance only for the recovery of the inpulse response based on a GD. The advantage gained from this apparent lack of generality is, that the thus defined condition on the input signal precisely describes the situation of identifiability for the identification algorithm that is really in use, and that the accuracy of the impulse response is a physically and numerically significant measure for the level within which the systems dynamical structure can be meaningfully recovered, with the AVAILABLE data.

This approach of more numerically based measurement of persistent excitation has been verified on its usefulness in a host of simulations, an example of which will be given in the section on examples. Its simplicity and self evidence make it attractive, but also invite the creative mind to investigate for some more refined and maybe even more useful definitions. One could for instance think of the following:

The conditional condition number $K_y(A)$ as defined in eq.(2-45) gives a more realistic measurement for the condition of the set of equations, when the main source of errors is in the output signal. But this condition number takes into account the output and is therefore, strictly speaking, more than an input condition (rather an input-output condition). The entire singular structure will of course give even more detailed information. For the purpose of quantifying persistent excitation however, our experience is that the condition number K(A) seems quite sufficient for the majority of practical situations.

Connection with the previous definition.

For a unit pulse function, the input condition number is unit; for white noise, the expectance of the condition number is also unit; and for a pure sine wave the condition number is infinite whenever m.(N+1) is larger than two. All this corresponds nicely with the examples cited above. But from the new definition, it is now possible to quantise situations in between these extreme cases, and to attribute a physically and numerically meaningful significance to this concept: the persistent excitation in a signal is defined as the inverse of the maximal increase in relative inaccuracy on the impulse response that can occur when solving the impulse response from the input output behaviour of the linear system to which this input is applied. A practical test has shown that a white noise input typically shows condition numbers varying from 1 to 5, when measured in matrices of typical dimensions (30x50 up to 150x200). It also shows that constant offsets cause quite large condition numbers (with one very large SV in the [1 1 ... 1] direction. And it finally reveals how dangerous some well established test signals for parameter estimation (such as the step function) really are (TABLE 10-1):

Some important remarks.

- Notice that the condition number concept is in fact not so restricted in significance as its definition based on GD would suggest. Indeed, if for instance the input-output condition is 100, and the available data are accurate within 1%, then this considered input-output behaviour can be described by two impulse responses that are 100% apart, so that based on these data, it simply makes no sence to make an estimation -with whatever method!- of the system model. One can therefore conclude that, provided M and N are sufficiently large, the GD based input condition is a quite general measure for persistent excitation.

- Notice also that in an on-line controller with a permanent identification task, it may be advisable either to keep track of the condition number (a VERY time consuming but safe solution), or to favour algorithms that do not further alter the result when the input is not sufficiently

well conditioned. This property can be obtained with ASOLVE if no acceleration rules are used, and if columns of too small a norm are discarded in the computations.

And finally, keeping track of the condition number allows to decide wether an update of the model makes sense or wether it should be rejected as due to error propagation only. It also can be used to detect periods of low identifiability, which may be dangerous periods for the process, if the system is likely to vary significantly during the considered time lapse. If such a situation would occur, keeping track of the involved condition number would allow to trigger an alarm signal or to activate the input up to the minimal identifiability level. In the section on examples, we will illustrate the importance of condition tracking with an example.

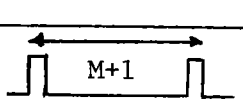
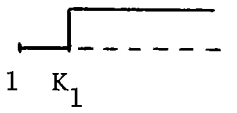
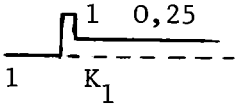
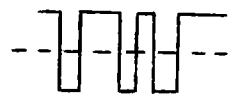
$\{u_k\}$	N	M	K_1	$K(U_{T,T-N,T-H})$
	X	X	X	1
	10 10 40 40	10 100 40 400	10 100 40 400	14.5 60.8 52.4 243
	10 10 40 40	10 100 40 400	10 100 40 400	2.76 9.0 8.15 35.14
	10 10 40 40	10 100 40 400	X X X X	2.2 1.4 1.7 1.6

TABLE 10-1: Some typical test signals and their input condition.

c. The Realization step.

Depending on the computational power that can be spent on the realization step, one can choose an appropriate algorithm among the solutions offered in chapter 9.

For our needs, we have extensively used realization 9-28:

$$H_{i_2}(1,K_1-1) \cong \tilde{U}'' \tilde{\Sigma}'' \tilde{V}'' \cong H^T_{i_3}(1,K_1-1) \qquad (10\text{-}5)$$

$$A = \tilde{\Sigma}''^{-1/2}\, \tilde{U}''^{t}\, H_{i_2}(2,K_1)\, \tilde{V}''^{t} \tilde{\Sigma}''^{-1/2} \qquad (10\text{-}6a)$$

$$B = \tilde{\Sigma}''^{1/2}\, \tilde{V}''^{t}\, [\, I_m \; 0\,]^T \qquad (10\text{-}6b)$$

$$C = [\, I_1 \; 0\,]\, \tilde{U}''^{t} \tilde{\Sigma}''^{1/2} \qquad (10\text{-}6c)$$

and the approximation sign in the SVD (10-5) means that the SSR is restricted to the n-dimensional subspace, that guarantees the required signal to noise ratio R_o, i.e.: if σ_v is the noise level on the entries of the H(k), and

$$\sigma_1^2 \geq \ldots \quad \sigma_n^2 \geq K\, R_o\, \sigma_v^2 > \sigma_{n+1}^2 \geq \ldots \geq \sigma_N^2 \qquad (10\text{-}7)$$

then the singular triplets associated to $\sigma_1 \ldots \sigma_n$ should be involved in the realization only. (K is the maximal dimension of the MBH matrix).

The above scheme has the advantage that it can be implemented directly, using the ASVD algorithm of chapter 5. In this manner, the storage efficiency and the computational involvement make the implementation on a microcomputer feasible.

d. The Tail Correction.

In the preceding sections on generalized deconvolution and realization, the long-term (k N) effects of the impulse response were assumed "not to disturb the identification essentially". Under this assumption, the realization of N+1 samples of the impulse response yields a State Space Model, that adequately represents an impulse response of infinite length. Using this model, it is possible to adjust the given output for the share, attributed to the "tail"

$$H(k) \quad , \quad k > N \qquad (10\text{-}8)$$

of the impulse response. The adjustment is a simple substraction of the

output of a (simulated) "tail-system" from the available output data. The impulse response of this "tail system" is zero in its first N+1 samples, and equal to the impulse response of the approximate system model in the remaining samples. The SSR of the tail system is easily derived from the original model, as follows:

$$\xi(k+1) = A \quad \xi(k) + A^{k_1} B\, u(k-N) \qquad (10\text{-}9a)$$

$$\eta(k) = C\, A^{k_2}\, \xi(k) \qquad (10\text{-}9b)$$

where $N = k_1 + k_2$, with k_1 and k_2 are integers of nearly equal value (simply equal when N is even) for balancing reasons.

Indeed, this State Space Representation has an impulse response

$$H(k) = 0 \quad , \quad k=0 \ldots N \qquad (10\text{-}10a)$$

$$H(k) = C\, A^{k-1} B \quad , \quad k=N+1 \ldots \qquad (10\text{-}10b)$$

that is zero in its first N+1 samples (the identified segment obtained from the GD), and equal to the impulse response of the realized system in all the remaining samples. Therefore the adjusted output

$$y^{**}(k) = y(k) - \eta(k) \qquad (10\text{-}11)$$

is the output of a system that has a finite impulse response, equal to the original impulse response in its first N+1 samples. We propose to call this adjustment a "tail-correction" adjustment.

The assumption now is that the approximation for the SSR is so close to the correct SSR, that the resulting tail-correction adjusts the output in an improving fashion, i.e. in such a way that the next GD and realization leads to an even more accurate approximation of the impulse response and SSR. If this is the case, successive iterations with GD, Realization and tail-correction should finally lead to an equation in which the truncation error is not present any more. This behaviour has indeed been observed, and has allowed to extract SSR's of systems, that contain so slowly varying components that a regular GD and realization would have required unrealistically high storage and computation time.

e. On State Vector Transmission.

Whether a system is really time varying, or whether its model varies due to convergence towards a more correct result, one will always have to transmit a state vector from one SSR to the next. In our identification procedure, this problem becomes of a quite fundamental nature.

The point is that from an input-output point of view, an SSR is unique only up to a similarity transformation. Whenever such an SSR has to be reco-

vered from input-output measurements, successive estimations may very well be equivalent (or nearly equivalent), but described in a completely different basis. There is in fact no essential relation between two such successive bases, and so when the estimation (or simulation) of an associated state vector is required, this state vector has to be "transplanted" not only from the previous model into the next one, but also from the basis associated to the previous model, into the (possibly totally different) basis, associated to the next model. The only way to break this deadlock, is to impose additional requirements on the SSR, so that the transgression from one model, TOGETHER with its associated state vector, is meaningfully determined. One directly sees the danger of canonical forms, that are defined in a purely formal way (e.g. companion canonical forms), and also understands the importance of a kind of "natural basis", that guarantees identical or at least close bases for realizations of closely related impulse responses. With the proposed algorithm, such a natural basis is provided. Indeed, the columns of matrix H in the realization (10-6) form the principal left singular basis of the natural state space, associated to the impulse response of the system by a (similarity invariant!) Hankel expansion . From any x vector in the associated SSR it is easy to reconstruct the representation in the ambient State Space:

$$\tilde{x}_k(k+1) = A_k \cdot \tilde{x}_k(k) \qquad (10\text{-}12a)$$

$$x(k+1) = \tilde{U}_k \cdot \tilde{x}_k(k+1) \qquad (10\text{-}12b)$$

The transplantation into the new basis is then performed by projection into the new State Space, followed by a decomposition in the new basis.

$$\tilde{x}_{k+1}(k+1) = \tilde{U}^t_{k+1} \cdot x(k+1) \qquad (10\text{-}13)$$

Notice that the above procedure implies that when the impulse response changes slowly, the variation per new estimation will be small, and the projection from one subspace into the next one will not affect the state vector itself very much (insensitivity of the principal singular subspace). But the representation within this subspace may alter completely from one SSR to the next (forward sensitivity of the orientation of two singular vectors of close singular values).

This again illustrates that the physically important aspect in the SVD is the global orientation of the principal subspaces; the internal orientation of the basis within this subspace is a formal problem, that has no physical implication. A very severe change in this orientation will affect the outlook of the result (sudden quite large variations in the formal components of the state vector), but will in no way affect the results of the identification nor estimation process. Notice however, that a sudden change of basis may seriously affect the computations of the observer and controller units, if this effect is not explicitely accounted for. This aspect is not analysed in this thesis, but one should appreciate that in general, a situation with identical singular values is rare, and that analoguously to the GD, the iterative nature of the total algorithm has a smoothing effect whenever this situation would occur.

Notice finally that tail correction requires a state vector for the computation of the tail adjustment, and hence that state vector transmission is unavoidable whenever tail correction is used, even when no observer nor controller are used (e.g. for pure modelling purposes).

C. IMPLEMENTATION AND EXAMPLES.

a. Implementation.

It is not necessary to recall in detail the conclusion of chapter 4, that advises a Global Linear Least Squares approximation as the most reliable solution of an overdetermined set of equations such as eq.(10-4) for the GD. If this approach is chosen, it will usually be advisable to downscale the RHS (which can be compensated by an equal upscaling of the solution) in order to scale the columns of the matrix up to an equivalent (absolute) inaccuracy level as the RHS. Indeed, the level of the input measurement inaccuracies will usually be small compared to the output measurement inaccuracies, combined with the process noise and (temporary) truncation error. If GLLS is chosen, no fast, storage-efficient, recursive algorithm is available yet. Hence this approach is still unaccesible for microcomputers. On the other hand, since the SVD has to be performed entirely, the input condition is readily available at each time instant, and one can easily detect moments of dangerous input condition. So if reliability is the major concern, there is no alternative to GLLS, and this step requires the high storage capacity for the full Hankel expanded matrix U(T,T-N,T-M), and the computation power for its full SVD.

If classical LLS can be assumed to be sufficiently reliable, the algorithm ASOLVE of chapter 6 allows for a considerable storage reduction: as the elements of U are used, never altered, it is sufficient to store the involved input vector samples instead of their Hankel expansion: M+N+1 vectors of m entries rather than (M+1).(N+1) vectors of m entries. Moreover, as the solution is not likely to vary much per iteration, the computational involvement can be restricted to a quite small amount: depending on the computing time available, one might perform simply one column update (which then becomes very similar to classical adaptive LS estimation...); a complete set of iterations over all columns; or even a few complete all-column iterations per time sample. This method is usually sufficiently reliable, but has the inconvenience that the input condition cannot be tracked permanently.

It is useful to point out here that the classical "cumulative correlation method", as still described in [102],[104] is completely overruled by the availability of ASOLVE: First, the occurrence of the U^t.U-type operation squares the singular values of the involved matrix, and hence the condition of the problem statement. Second, the U^t.U matrix has no more than symmetric structure, so that no storage reduction comparable to the one possible with U is possible, even for very large M. In fact ASOLVE implicitely chooses a windowed approach, but it is so storage efficient that the window can be taken quite long. Moreover the updating aspect of ASOLVE in a certain sense performs the same task of keeping very old information: indeed, it requires some iterations on well oriented columns in the input matrix (hence a well conditioned input!) before an earlier estimation of H can be iterated into a completely new one. One could say that ASOLVE is

the key to microcomputer GD, and ASVD the key to microcomputer realization for large system models: if N is chosen large (which is possible thanks to ASOLVE and ASVD), then a quite large class of linear systems is identifiable from the generalized deconvolution and realization. The need for sufficiently high N is required both by degree arguments ($N \geq i_c + i_o$) and by the requirement to identify reliably a broad range of damping coefficients and eigenfrequencies (see chapter 9).

Within one iteration, one usually observes two phenomena: first the measurement errors are smoothed due to the overdetermination of the set of equations involved in the GD, and then they are filtered after the (reduced degree) realization step.

The tail correction iteration is usually clearly seen to improve the identification accuracy down to the level of input and/or output accuracy, possibly limited (under start-up conditions) to the level of dynamic behaviour due to the unknown initial state vector at t=0.

Finally, it should be kept in mind that no algebraic analysis is available on the circumstances under which convergence can be guaranteed. The only evidence on the practical applicability of the global identification algorithm is a set of tests and simulations that were performed, the results of which are presented now.

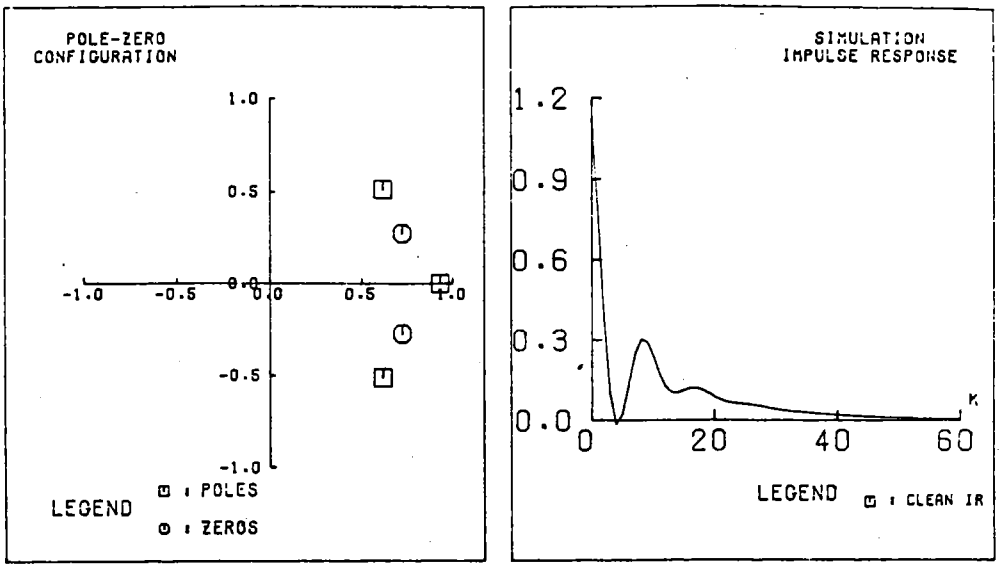
b. Examples.

The idea of a global SSR identification based on realization arose after the succesful implementation of the realization algorithms. Early examples, investigations and results, obtained in the laboratory were described in [102, 106, 107]. Some experiments are about to conclude, others are still in process [68, 69]. The following gives a status of the results obtained up to now, referring to work in process if necessary:

1. An introductory example [105, 106]:

The input-output behaviour of a time invariant monovariable system of degree 3, with the pole-zero configuration and impulse response represented in Fig.(10-6a), is measured as indicated in Fig. (10-6b). Its output is known to be disturbed by process noise with a level of 5% of the output power level; the measurement precision at in- and output is 1%. An additional constant offset of 2%., and drift of 2% per 1000 samples completes the poisening of the data. Fig 10-7 then compares the accuracy of the obtained impulse response (in 2 norm) for a small-windowed GD, the method of cumulative correlation and the GD with tail correction. No comment is necessary to appreciate the net improvement obtained with the tail correction adjustement. (N=20 was chosen).

Fig.10-7 clearly illustrates how useful it is to track the condition number. For each of the 3 methods, there is a clear relationship between the condition number of the involved input matrix, and the accuracy of the obtained impulse response.



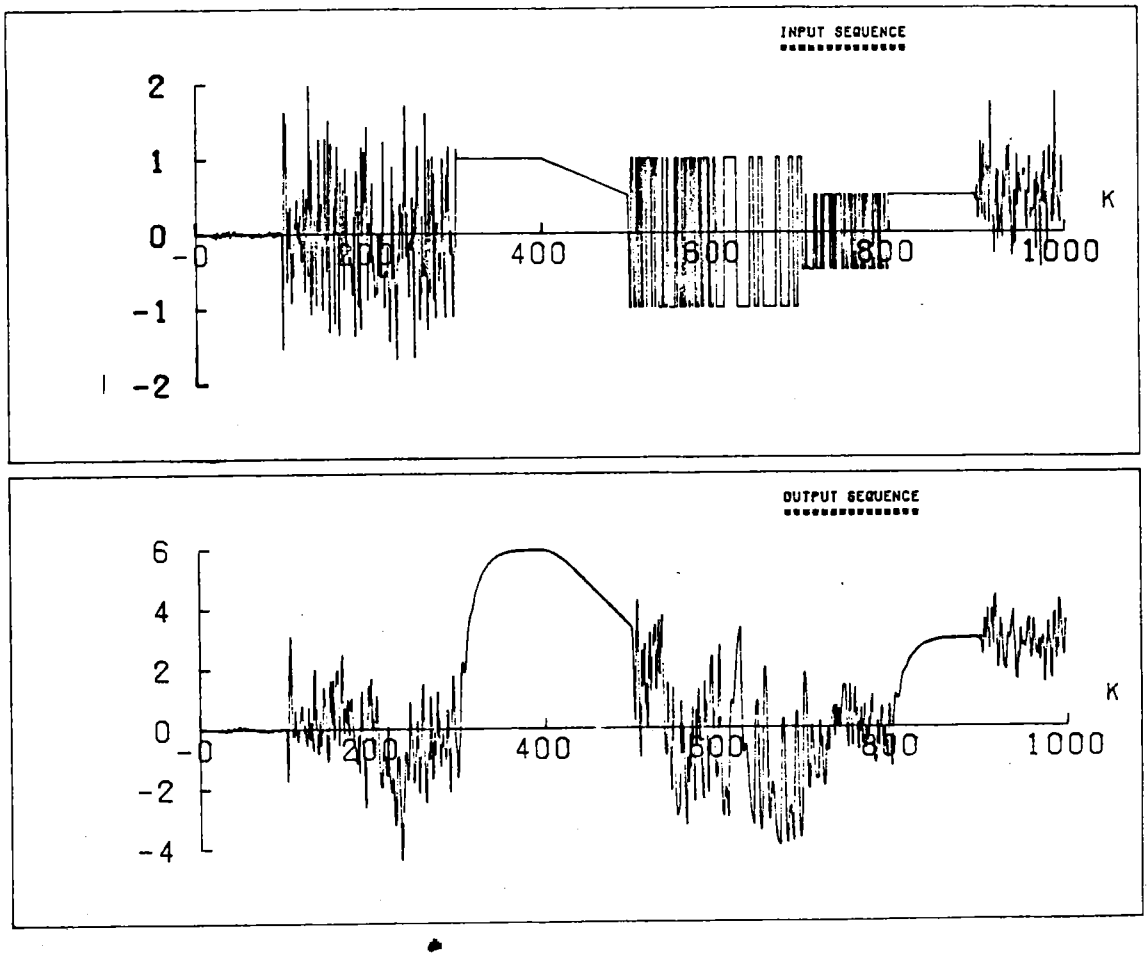


Fig.10-6a: Pole-zero plot, impulse response, correct (but unavailable) input-output behaviour for a monovariable system of degree 3.

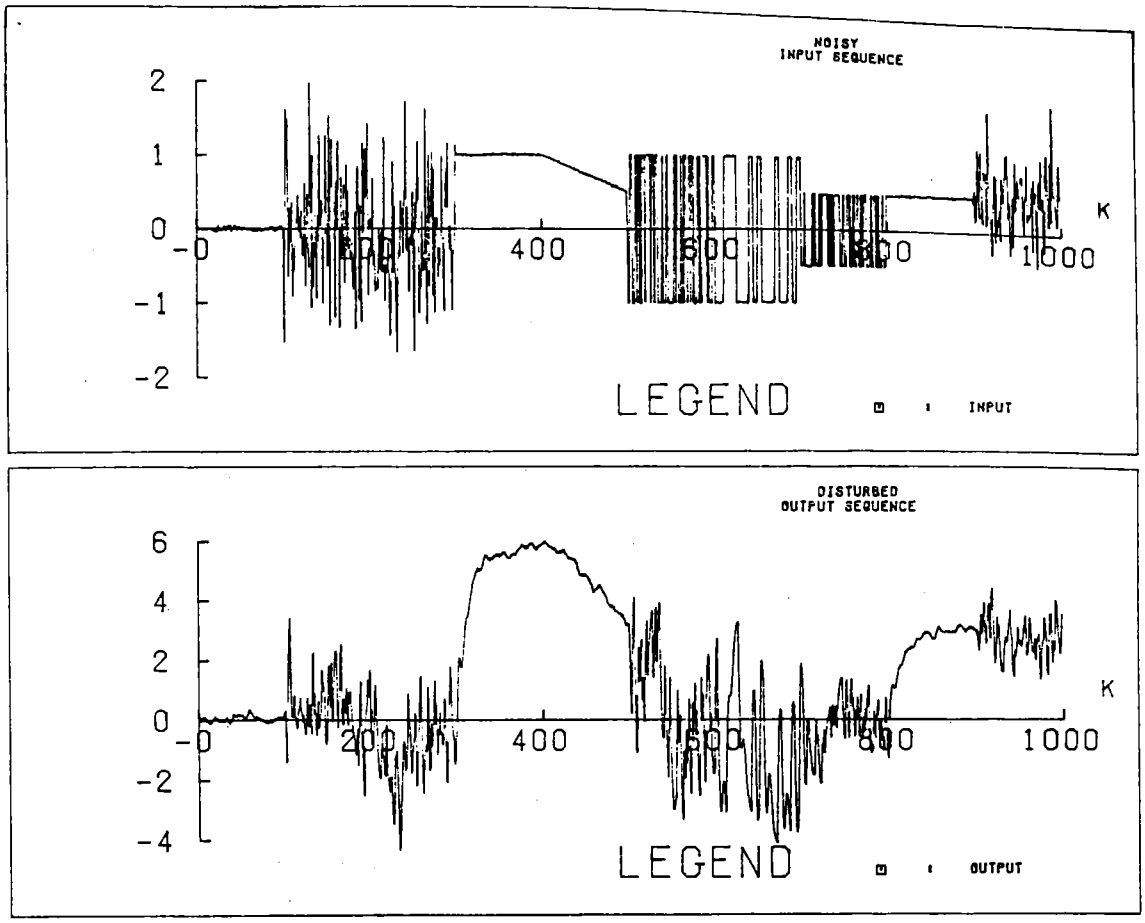


Fig.10-6b: Measured I-O behaviour.

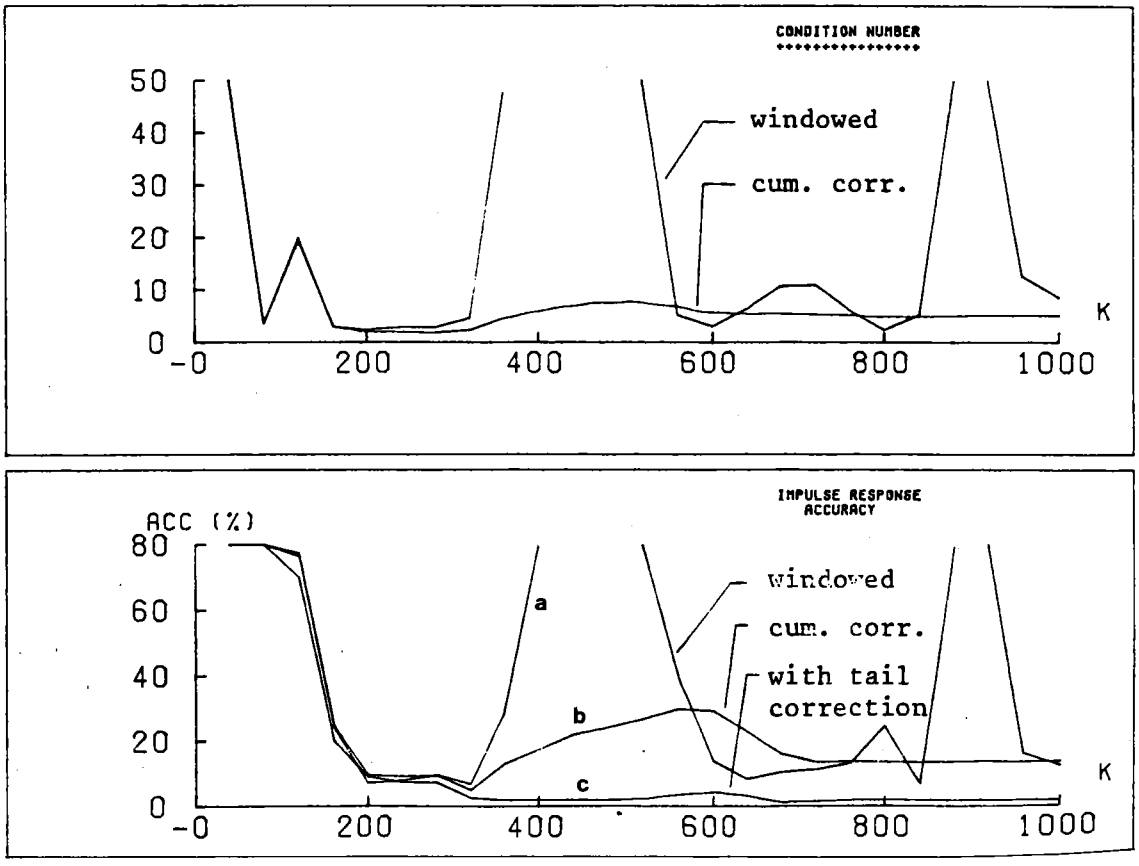


Fig.10-7: The noisy (available) input-output measurements, the computed input condition, and the obtained accuracy on the identified model, for three different variants of AIDENT
(a) and (b) without tail correction; (c) with tail correction.

2. A rough analysis on tail correction.

A simple exercise can put the problem in its correct perspective. If the slowest dominant mode in an impulse response dies out with a damping ratio a, then this mode will have decreased over a relative factor F

$$F = a^N \qquad (10\text{-}14)$$

after N time steps. If one had to eliminate the truncation error by using enough samples in the GD, even for mild requirements on F, the finite section of the impulse response in the GD would need to be quite long, whenever the damping ratio is close to unity (Fig.10-8).

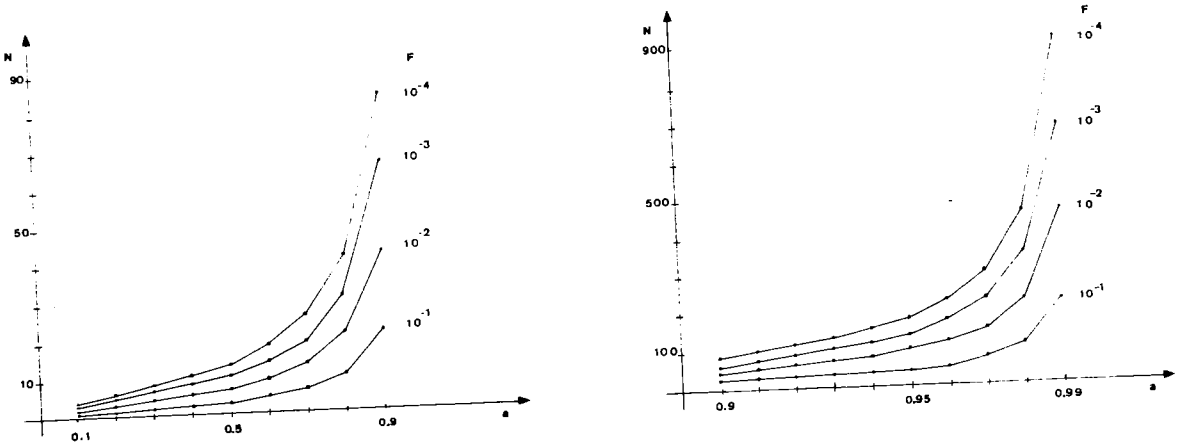


Fig.10-8: number N of samples in the GD, required to keep the truncation disturbance below a level F.

With F=0.001 and a=0.95 for instance, N=135 samples would be necessary in the GD step. Whenever one has to cope with systems of reasonable stiffness (with slow modes mixed with fast modes in the impulse response), the Nyquist rate will impose a minimal sample frequency, and the slow modes will automatically be associated to poles close to the unit circle. With tail correction, the dimensions are no longer so strictly dictated by the slowness of the impulse response, rather by more fundamental restrictions such as the upperbound on the system degree, the numerical properties of the realization (chapter 9), and the convergence behaviour of the tail correction. To these aspects, we focus the attention in the next section.

3. Convergence of the tail correction [69].

Fig 10-9 was obtained recently in a masters thesis [69] and shows the convergence behaviour of the algorithm AIDENT, that is typical provided a sufficient number of samples is used in the GD.

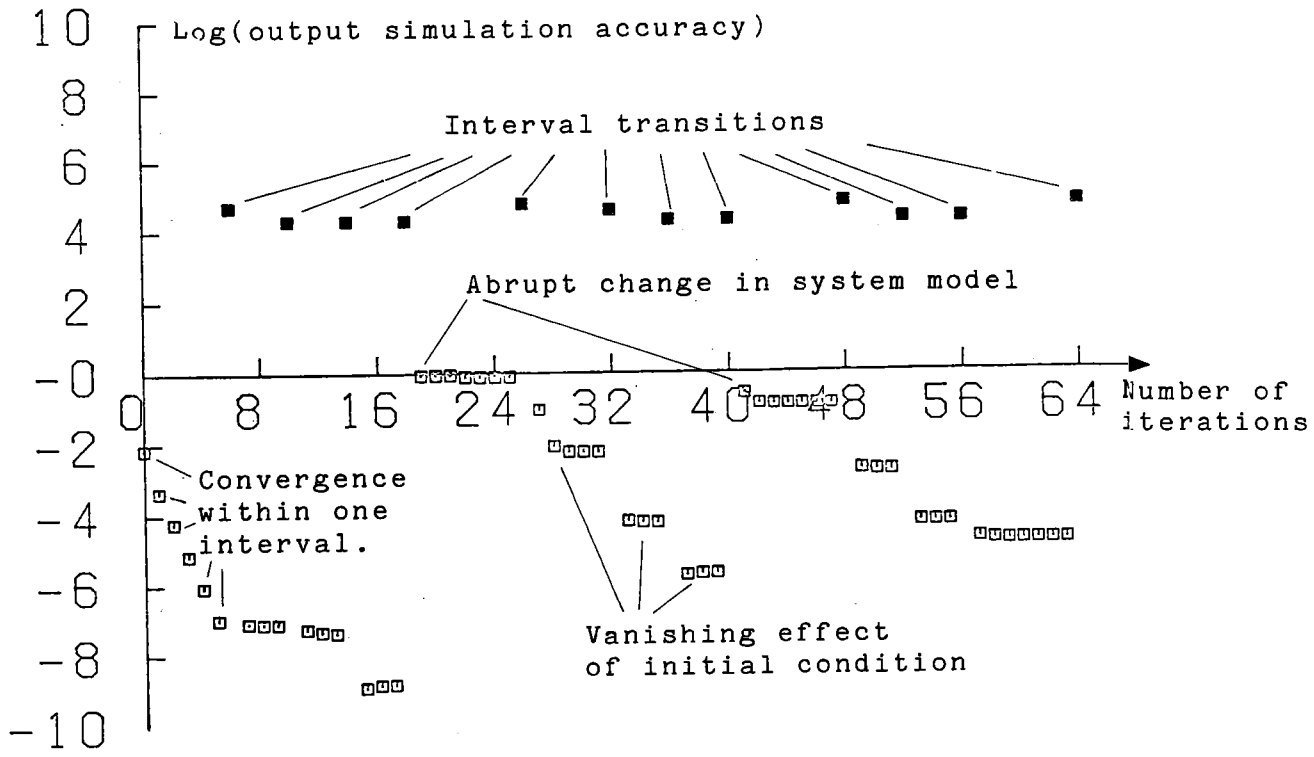


Fig.10-9: typical convergence behaviour of a tail correction algorithm.

One clearly distinguishes two convergence patterns: the first convergence is within one time interval, and shows improvement in the accuracy of the result, due to tail correction adjustment. This convergence saturates at the level of the data accuracy (which means that the truncation effect is iterated out), or at the level of the error due to the unknown initial condition of the system (as in the shown example). This error level is seen to die out gradually as time increases.

The above example is a simulation with an algorithm that still uses cumulative correlation, and that treats the data in time intervals of M samples. There is not yet an integrated algorithm that combines ASOLVE, ASVD, and the tail correction procedure.

From a series of simulations, it would appear that N=40 is a sufficiently long finite interval, to guarantee a global convergence for modes up to a=0.96 [69]. Together with the requirement for 60...80 samples in the impulse response, in order to guarantee a reliable realization over a quite broad range of systems (chapter 9), it seems that N=60...80 and hence M=90.m ... 120.m is a quite safe choice that allows a reliable identification over the class of nearly all linear systems of practical

significance. With ASOLVE and ASVD, the computations are implementable on todays standard microcomputer equipment, the only limitation being the speed.

Recent experience seems to confirm these claims, but a more rigorous analysis on the correctness of the above statements is certainly necessary.

4.Time varying systems [114].

In general, for time invariant systems, one complete iteration per time step has appeared to converge sufficiently fast to the system impulse response, when the input signal is close to white noise. For safety one could provide for 2 or 3 iterations for the case that the input signal would be somewhat worse in condition, but more is not advisable as the nice smoothing property would then disappear. For time variant systems, the behaviour is still satisfactory, but the full analysis of the algorithm's behaviour is not yet available. Instead, consider the following examples:

a. Abrupt transitions:

Some examples of systems with abrupt transitions from one model to another are shown in Figs.(10-10)-(10-12). Such transitions could for instance be the consequence of an abrupt change in working conditions of a nonlinear system. The evolution of the GD, obtained with ASOLVE from the associated input-output data is shown in the suggestive Figs.(10-10)-(10-12) showing the evolution of the identified IR along a third "real time" axis. One iteration over all columns of U was used.

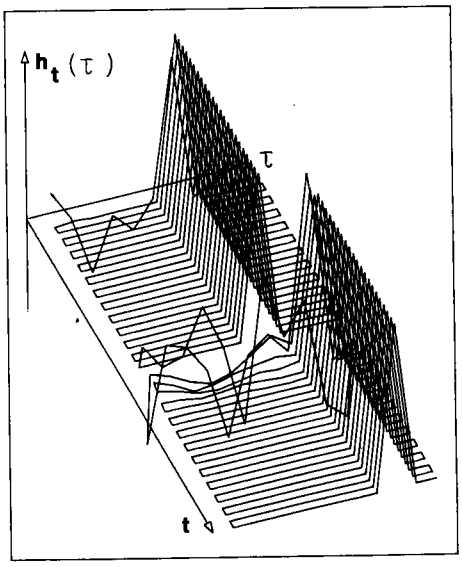


Fig. 10-10: A pure-delay system, with abrupt change in the length of the delay.
Fast convergence of the algorithm.

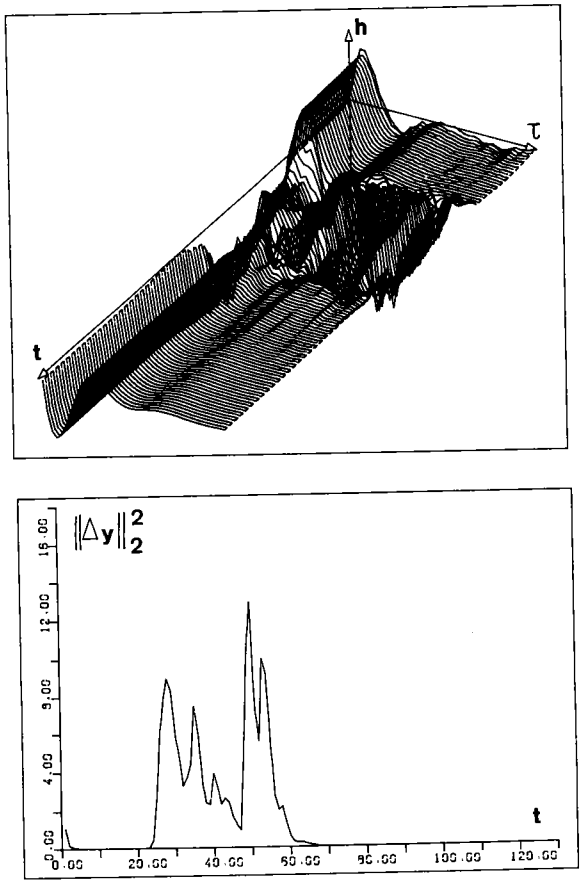


Fig.10-11: Identification of an abrupt change in system model: inversion of impulse response.

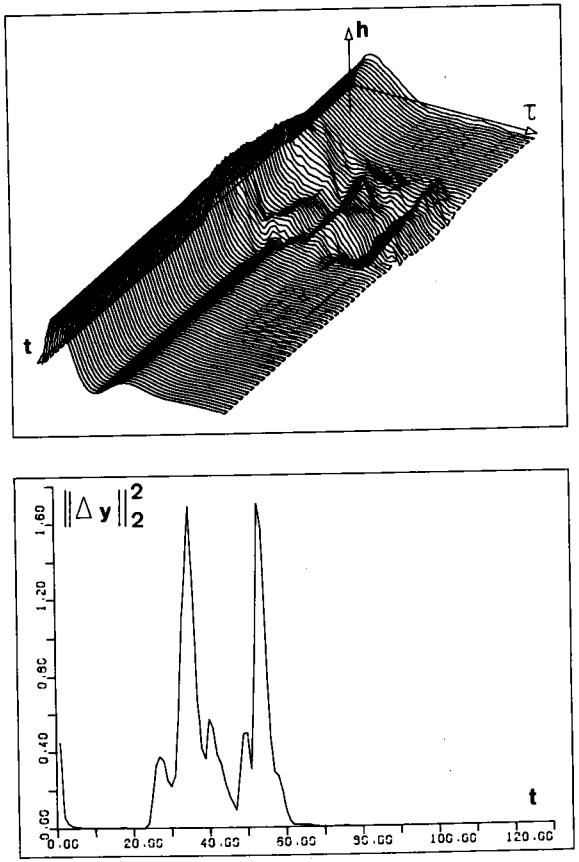


Fig.10-12a: Identification of an abrupt change in system model: doubled eigenfrequency.

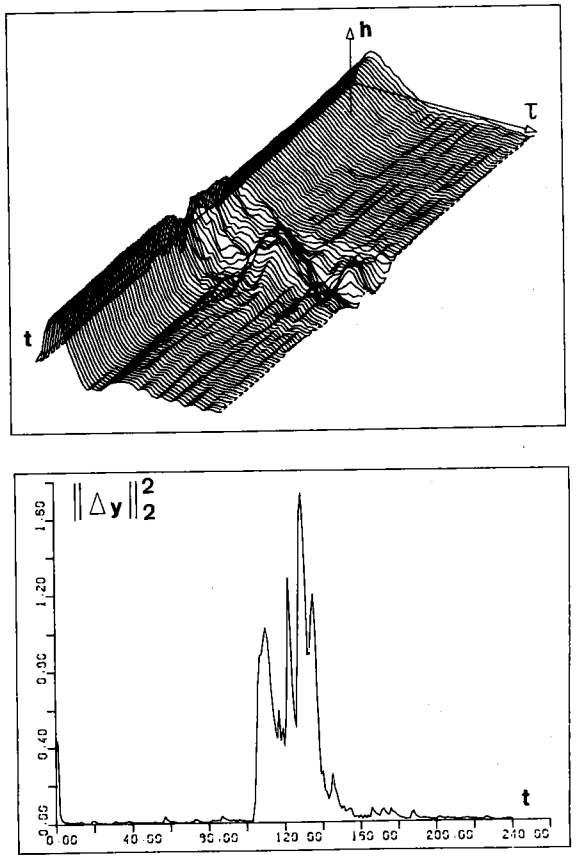


Fig.10-12b: Identification of an abrupt change in system model:same problem as Fig.(10-12a), but with 5% noise level.

Notice that the algorithm needs a certain period of transition. This is inevitable, and should even be considered as a positive property if one realizes that the algorithm does not use any a-priori structure nor degree information on the system. If the structure and degree are left free, the transient behaviour around the abrupt transition could be matched on a variety of system models, and it is to be considered a good property, that the algorithm does not stick a priori to one of these models. The correct, and the recovered impulse response for exact data and data blurred with 1% and 5% noise on both input and output are shown. These examples and the shown results, obtained using the above described methods, were recently worked out by Claeys and De Bock in their Master's Thesis.

Notice that the error on the RHS of the GD equation, plotted along with the result, appears to be a good detector for such critical instances.

b. Smoothly varying systems:

Very often the system characteristics of a process will change slowly and smoothly (e.g. due to slow drift in working condition). Such a situation is simulated in Fig. (10-13), and in the next multivariable example.

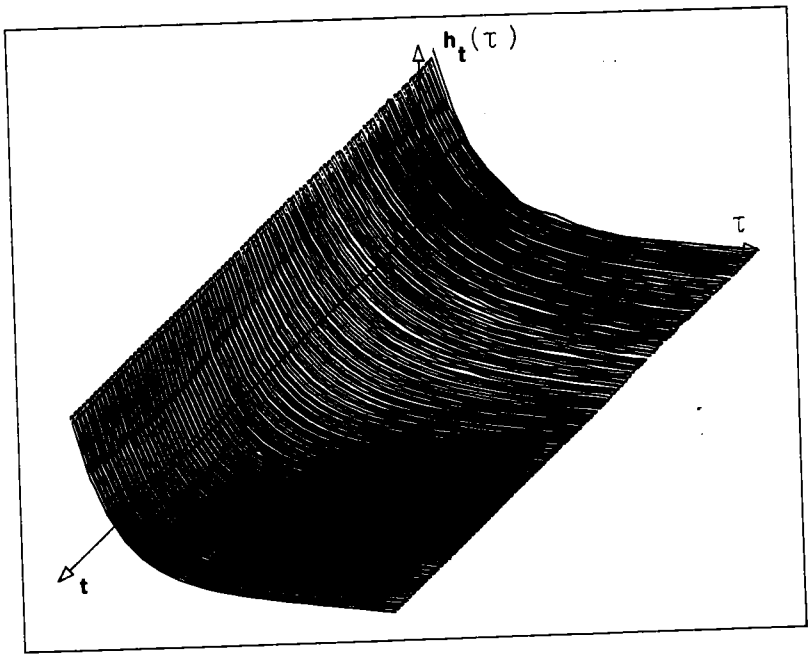


Fig.10-13: Identified impulse response of a monovariable system with one real pole, that smoothly varies from 0.9 to 0.7.

c. A multivariable example:

The input output behaviour of a 2-input 2-output system was simulated, and poisened with 5% noise. The recovered impulse response from ASOLVE is shown in Fig. (10-14). The equivalent of AIDENT, that combines the new ASOLVE and ASVD routines with the Tail-correction algorithm into one multivariable identification procedure is not yet available. Hence the impulse responses were obtained from a generalized deconvolution without realization, and therefore still show the typical noisy behaviour, that is filtered out after the realization.

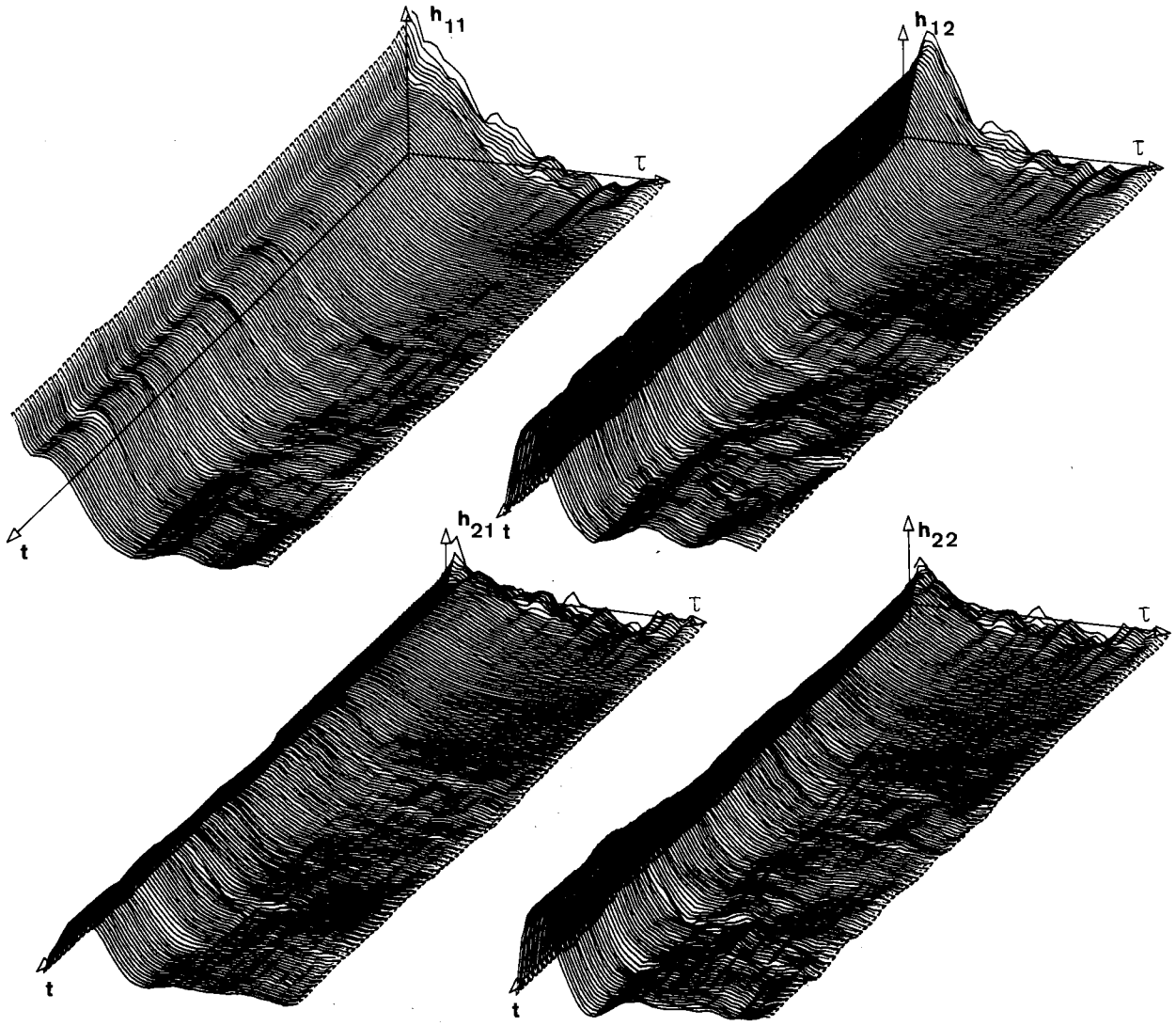


Fig.10-14: Identified impulse response (without realization) of a (2x2) multivariable system of third degree, the poles of which smoothly vary in time.

5.Conclusion: further investigation is necessary.

It will be clear from the above discussion, that the analysis of the algorithms, the concepts of which were introduced and discussed in this chapter, is far from being fully achieved. Further work has to be done on the precise convergence conditions of the global algorithm, with respect to the class of systems recoverable (especially in the case of time variant systems), the levels of allowable noise, and the optimal strategy of iteration sequences. Also, the implementation aspects for application on microprocessor level is an aspect that has not been analysed in full detail. But some conclusions are already unquestionable: this State Space oriented adaptive and fully multivariable algorithm is reliable and allows in a unique, automated way to determine the structure and degree, based on the given measurements only. The author is convinced of its nice future.

CONCLUSION

In a conclusion there is always a temptation to exhibit some feelings of selfcongratulation. Instead, I would like to express my sincere gratitude to the readers that reached the end of this manuscript without skipping the chapters 1 to 10. I hope they enjoyed the reading and I would feel very proud, might they ever apply a concept or idea introduced in this text.

The main objective of this document was to share with the reader some essential ideas, concepts and conclusions that prove to be important in the analysis and implementation of linear modelling procedures. The basic guideline throughout the thesis is a conceptual understanding of the geometric significance of the Singular Value Decomposition (SVD)

After a short historical note we have briefly reviewed the classical SVD properties. In a conceptual analysis, the fundamental SVD-related geometric properties of a matrix were analysed: the classical interpretation of the SVD of a matrix seen as a linear operator, and a new interpretation of the SVD of a matrix seen as a sequence of column or row vectors. For the latter, the concepts of ORIENTED ENERGY and MINIMAL SIGNAL-TO-SIGNAL RATIO for vector sequences observed from a subspace were introduced, and their properties established. These geometric interpretations of SVD have been the further guideline throughout the thesis.

Chapter 3 reviews the sensitivity properties of SVD and of the solution of a set of linear equations. The essential difference between the CONDITION of a problem, and the NUMERICAL STABILITY of the algorithms available to solve it, is discussed in the light of our geometric interpretation. A nice pictural view at the classical 2-norm condition number of a MATRIX, and at the 2-norm condition number of a SET OF EQUATIONS readily followed from this analysis. In chapter 4 a new Linear Least Squares approximation technique for the solution of overdetermined sets of linear equations is discussed in the same line of thought: the key idea is that for sets of equations where both the matrix and the Right Hand Side are inaccurate, a global minimization of the reconstruction error on the matrix TOGETHER with the RHS is more meaningful than a minimization of the reconstruction error on the RHS only. Evidence of this statement was given in a simple example, and the computation of Global Linear Least Squares solutions is discussed.

Two new algorithms that showed up during the study of fast and storage efficient reliable identification methods are then described. ASVD performs a Singular Value Decomposition, and ASOLVE computes the LS solution of a set of linear equations. Both algorithms are directly related to the geometric SVD interpretation, and show the nice property not to alter the involved matrices: instead, the entries of the matrix are used to iterate continuously towards the solution, and the algorithms become extremely advantagous when large sequences of problems with slowly varying, highly structured matrices have to be solved.

The next four chapters finally discuss results obtained in the domain of linear system theory.

Chapter 7 discusses the numerical implications of the choice of a reference node in the nodal analysis of large circuits, and the main conclusion is that ALGEBRAIC EQUIVALENCE between two circuit descriptions certainly DOES NOT IMPLY NUMERICAL EQUIVALENCE: one description may be ill conditioned whereas the other may be well conditioned. The type of description that will lead to the most reliable analysis appears to depend on the circuit itself but tight bounds on the possible variation in condition from one description to another are presented. The final question on the determination of the type of description to be used for optimal reliability, from a preliminary analysis of the circuit, remains unanswered.

An application in the domain of analytical mechanics is found in chapter 8, where the main philosophical conclusion is that reliable algebra should be developped under guidance of numerical arguments, from its earliest statements on.

Chapter 9 treats the reliable realization of a State Space Representation from a given discrete-time impulse response sequence, and chapter 10 finally considers the problem of the complete identification of a State Space Representation, directly from the measurement of its (multivariable) input-output behaviour. The presented algorithm AIDENT is believed to be the first on-line model-tracking procedure for multivariable State Space Realizations that is implementable on a standard microcomputer even for model degrees up to 15.

The general conclusion is that the analysis of the numerical behaviour of linear system modelling procedures is still in a quite early stage of its development. Quite fundamental and surprising properties still regularly appear, and it is certainly worthwile to further investigate this fascinating aspect of modelling theory.

Leuven, june 29, 1982.

REFERENCES.

[1] C.C. Mac Duffee, The theory of matrices. Berlin: Springer, 1933.

[2] L. Autonne, Bull. Soc. Math. France, Vol. 30, pp. 121-133, 1902.

[3] C. Eckart and G. Young, "A principal axis transformation for non-Hermitean matrices," Bull. Amer. Math. Soc., Vol 45, pp. 118-121, 1939.

[4] G.H. Golub and W. Kahan, "Calculating the Singular Values and Pseudo inverse of a Matrix," SIAM J. Numer. Anal., 2, No. 3, pp. 205-224, 1965.

[5] G.H. Golub and C. Reinsch, "Singular Value Decomposition and Least Squares Solutions," Numer. Math., Vol. 14, pp. 403-420, 1970.

[6] G.W. Stewart, "Error and perturbation bounds for subspaces associated with certain eigenvalue problems," SIAM Review, Vol. 15, No.4 pp. 727-764, 1973.

[7] S.L. Campbell and C.D. Meyer, "Generalized Inverses of Linear Transformations," London: Pitman 1979.

[8] G.E. Forsythe and C.B. Moler, "Computer solution of Linear algebraic systems", Englewood Cliffs, N.J., Prentice Hall 1967.

[9] B. Garbow et al., "Matrix eigensystem routines - EISPACK guide extension," (Lecture Notes in Computing Sciences, Vol. 51) New York: Springer 1977.

[10] J. Dongarra et al., "Linpack User's Guide", Philadelphia: SIAM 1979.

[11] V.C. Klema and A.J. Laub, "The Singular Value Decomposition: Its computation and Some Applications," IEEE Trans. Aut. Contr., Vol AC-25, No. 2, pp. 164-176, 1980.

[12] Special Issue on Multivariable Control Systems, IEEE Trans. Aut. Contr., Vol AC-26, No. 1, 1981.

[13] P. Van Dooren, "Numerical Linear Algebra: an increasing interest in Linear System Theory," Proc. ECCTD, The Hague, pp 243-251, August 1981.

[14] C.L. Lawson and R.J. Hanson, "Solving Least Squares Problems," Prentice Hall Series in Automatic Computation, Englewood Cliffs 1974.

[15] T. Kato, "Perturbation theory for Linear Operators," Berlin: Springer 1980.

[16] Bjorck and G.H. Golub, "Numerical methods for computing angles between linear subspaces," Math. Comput., Vol. 27, pp. 579-594, 1973.

[17] G.E. Forsythe M.A. Malcolm and C.B. Moler, "Computer Methods for Mathematical Computations", Englewood Cliffs, N.J., Prentice Hall 1977.

[18] G.W. Stewart, "On the perturbations of pseudo inverses, projections and linear least squares problems," SIAM Review, Vol. 19, No.4 pp. 634-662, 1977.

[19] J.H.Wilkinson "The algebraic Eigenvalue problem," London, Oxford Univ. Press, 1965.

[20] W.M. Wonham, "Linear Multivariable Control: A geometric approach," New York: Springer 1974.

[21] L.S. De Jong "Numerical aspects of Realization algorithms in Linear Systems theory.," Doct. dissertation, T.H. Eindhoven, 1975.

[22] L.S. De Jong "Numerical aspects of the recursive realization algorithm," SIAM Journal on Control and Optimization, Vol. 16, pp. 646-659, 1978.

[23] L.A. Zadeh and C.A. Desoer, "Linear System Theory," New York: Mc Graw Hill, 1963.

[24] C.A. Desoer, "Notes for a second course on Linear Systems," New York: Van Nostrand Reinhold, 1970.

[25] T. Kailath, "Linear System theory," Englewood Cliffs, N.J., Prentice Hall 1980.

[26] B.L. Ho and R.E. Kalman, "Effective Construction of Linear State Variable models from Input/Output functions," Proc. 3-rd Allerton Conf. Circuits and Systems Theory, pp. 449-459, 1965.

[27] L.M. Silverman, "Realization of Linear Dynamical Systems," IEEE Trans. Aut. Contr., Vol AC-16, No. 6, 1971.

[28] J.L. Massey, "Shift register synthesis and BCH-decoding," IEEE Trans. Information Theory, Vol IT-15, pp. 122-127, 1969.

[29] J. Rissanen, "Realization of matrix sequences," IBM Techn. Report, 1972.

[30] H.P. Zeiger and A.J. Mc Ewen, "Approximate Linear Realizations of given dimension via Ho's algorithm," IEEE Trans. Aut. Contr., Vol AC-19, pp. 153, 1974.

[31] S. Kung, "A new identification and model reduction algorithm via Singular value decomposition," Proc 12-th Asilomar Conf. Circ. Syst., comp., Nov. 1978.

[32] J. Wilkinson, C. Reinsch, "Handbook for Automatic Computation II: Linear Algebra," New York: Springer, 1971.

[33] K. Astrom and P. Eyckhoff, "System Identification: A survey," Automatica, Vol. 7, pp. 123-162, 1971.

[34] T. Kailath, "A view of of three decades of Linear Filtering Theory," IEEE Trans. Inform. Theory, Vol IT-20, pp. 146-181, 1974.

[35] C. Eckart and G. Young, "The Approximation of One Matrix by Another of Lower Rank," Psychometrika, Vol. 1, 211-218, 1936.

[36] C.T. Chen, "Introduction to Linear System Theory," H.R.W. Series in Electrical Engineering, 1971.

[37] B.L. Ho and R.E. Kalman, "Effective construction of Linear State Variable Models from Input/Output Functions," Regelungstechnik, Vol. 14 heft12, pp. 545-548, 1966.

[38] P. Eyckhoff, "System Identification," Wiley, 1974.

[39] H. Kwakernaak and R. Sivan, "Linear Optimal Control Systems," New york: Wiley Interscience, 1972.

[40] E. Oran Brigham ,"The Fast Fourier Transform," Englewood Cliffs: Prentice Hall, 1974.

[41] J.E. Ackermann, "Die minimale Ein-Ausgangs-Beschreibung von mehrgrossen Systemen und irhe Bestimmung aus Ein-Ausgangs-Messungen," Regelungstechnik Vol. 19, pp. 203-206, 1971.

[42] D.R. Audley, "A method of constructing minimal approximate realizations of linear input/output behaviour," Automatica Vol. 19, pp. 409-415, 1977.

[43] E.G. Gilbert, "Controllability and observability in multivariable control systems," SIAM Journal on control, Vol. 1, pp. 128-151, 1963.

[44] C.J. Gueguen and E. Toumire, "Comments on irreducible Jordan form realization of a rational matrix," IEEE Trans. Aut. Contr., Vol. AC-15, 1970.

[45] R. Guidorzi, "Canonical structures in the identification of multivariable systems," Automatica, Vol. 11, pp. 361-374, 1975.

[46] R.E. Kalman, "Mathematical description of linear dynamical systems," SIAM Journal on Control, Vol. 1, pp. 152-174, 1963.

[47] R.E. Kalman, "Irreducible realizations and the degree of a rational matrix," SIAM Journal on Control, Vol. 1, pp. 520-544, 1963.

[48] Y.L. Kuo, "On the irreducible Jordan form realization and the degree of a rational matrix," IEEE Trans. Circ. Theory, Vol. CT-17, pp. 322-332, 1970.

[49] S.P. Panda and C.T. Chen, "Irreducible Jordan form realization of a rational matrix," IEEE Trans. Aut. Contr., Vol AC-14, No. 1, pp. 66-69, 1969.

[50] J. Rissanen, "Recursive identification of linear systems," SIAM Journal on Control, Vol. 9, pp. 420-430, 1971.

[51] H.H. Rosenbrock, "State space and Multivariable Theory," London: Nelson 1970.

[52] W.A. Wolovich, "Linear Multivariable Systems," Berlin: Springer 1974.

[53] D.C. Youla and P. Tissy, "N-port synthesis via reactance extraction - part I," IEEE Int. Conv. Rec. Vol. 4 (7), pp. 183-205, 1966.

[54] C.A. Desoer and M. Vidyasagar, "Feedback Systems:Input-Output properties," New York, Academic Press, 1975.

[55] L.C.Suen and R. Liu, "The determination of the rank of a large noisy matrix," Proc. Workshop on Large Scale Networks and Systems, IEEE Symposium on Circuits and Systems, Houston (Texas), April 1980.

[56] J.H. Wilkinson, "Elementary proof of the Wielandt-Hoffman theorem and its generalization," Stanford University Report No CS-150, Stanford (California), 1970.

[57] R.J. Hoffman and H.W. Wielandt, "The variation of the spectrum of a normal matrix," Duke Math. Journal, Vol. 20, pp. 37-39, 1953.

[58] C. Davis and W. Kahan, "The rotation of eigenvectors by a perturbation III," SIAM J. Num. Analysis, Vol. 7, pp. 1-46, 1970.

[59] P. Wynn, "On a device for computing the em(Sn) Transformation", MTAC, Vol.10, pp.91-96, 1956.

[60] G.A. Campbell, "Direct Capacity Measurement," Bell System Technical Journal, Vol.1 Nr.1 pp. 18-38, 1922.

[61] L.O. Chua and P.M. Lin, "Computer aided analysis of electronic Circuits," Prentice Hall, 1975.

[62] E. Cohen, "Performance Limits of a dedicated CAD system," Proc IEEE Symp. on Circuits and Sytems, pp. 708-717, April 1980.

[63] W.F. Hughes and E.W. Gaylord, "Basic equations of engineering science," Shaum publishing company, Ney York, 1964.

[64] B.C. Moore, "Principal Component Analysis in Linear Systems:Controllability, Observability, and Model Reduction," IEEE Trans. Aut. Control, Vol AC-26 pp.17-32, Feb. 1981.

[65] F.R. Gantmacher, "The theory of Matrices", Chelsea, New York 1959.

[66] C. Roveda, R. Schmidt, "Upperbound on the Dimension of Minimal Realizations of Linear Time-Invariant Dynamical Systems," IEEE Trans. Aut. Control, Vol AC-6 pp.639-644, Dec. 1970.

[67] D.G. Luenberger, " Introduction to Dynamic Systems," John Wiley and Sons, New York 1979.

[68] H. de Bock and F. Claeys, "ASOLVE: een efficient identificatie algoritme voor systemen met meerdere ingangen en meerdere uitgangen," Master's thesis 1982 (in Dutch).

[69] L. De Vos and F. Dezeure, "Een multivariabel adaptief toestandsmodel-identificatie algoritme met staartcorrectie-procedure," Master's thesis 1982 (in Dutch).

[70] J.R. Bunch and C.P. Nielsen, "Updating the Singular Value Decomposition," Numerische Math. Vol 31 (2) pp.111-130, 1978.

[100] J. Staar, "On Direct Realizations of Companion Canonical State Space Representations with M.B.H. procedures," Internal report ESAT laboratory, June 1980.

[101] J. Staar, J. Vandewalle and M. Wemans, "A Comparison of multivariable MBH algorithms in the presence of multiple poles, and noise disturbing the Markov sequence," Proc. 4-th Int. Conf. on analysis and optimization of systems, (Lecture notes in Control and Information Sciences), pp. 141-160, Springer Verlag, Versailles, December 1980.

[102] J. Staar, J. Vandewalle, and M. Wemans, "Realization of measured multivariable input-output sequences with prescribed uncertainty," Proc. 1st IASTED Int. Symp. Modelling, Identification and Control, pp. 62-66, Davos, Feb. 1981.

[103] J. Staar, "A study of MBH realization algorithms," AUTOMATICA, Vol. 17, No 3, pp. 523-534, May 1981.

[104] J. Staar, J. Vandewalle, and M. Wemans, "Reliable modeling algorithms for adaptive controllers," Proc. 13th CIRP Int. seminar on manufacturing systems, (Faculty press Ljubljana, Yugoslavia) pp. 119-134, Leuven, May 1981.

[105] M. Wemans, J. Staar, and J. Vandewalle, "ADAPT : an interactive program for the analysis and design of adaptive controllers," Proc. 13th CIRP Int. seminar on manufacturing systems, (Faculty press Ljubljana, Yugoslavia) pp. 135-150 Leuven, May 1981.

[106] J. Staar, J. Vandewalle and M. Wemans, "A general class of numerically reliable algorithms for the realization of truncated impulse responses," Proc. IFAC Wold congress VIII, Vol. 1, pp. 7-13, Kyoto (Japan) Aug. 1981.

[107] J. Staar, J. Vandewalle and M. Wemans, "Realization of truncated impulse response sequences with prescribed uncertainty," Proc. IFAC Wold congress VIII, Vol. 1, pp. 13-19, Kyoto (Japan) Aug. 1981.

[108] J. Staar and J. Vandewalle, "On the feasibility of model identification," Invited paper, Proc. ECCTD pp. 832-837, The Hague, Aug. 1981.

109] J. Staar and J. Vandewalle, "Automatische modellering van multivariabele lineaire systemen: theorie en voorbeelden," Proc. Studiedag Vlaamse Systeemgroep (in Dutch), Antwerpen, April 1981.

[110] J. Staar and J. Vandewalle, "On the numerical implication of the choice of a reference node in the nodal analysis of large circuits," Int. Jrnl. Circuit Theory and Applications, Vol.9, pp.488-492, 1981.

[111] J. Staar and J. Vandewalle, "On the relationship between Singular Value Decomposition, and the Principal Axes and Moments of Inertia of a rigid body," to be submitted.

[112] J. Staar and J. Vandewalle, "Singular Value Decomposition : A reliable tool in the algorithmic analysis of linear systems," Journal A, Vol.23 No.2, pp.69-74, april 1982.

[113] J. Vandewalle and J. Staar, "Modelling of linear systems:critical examples, problems, and numerically reliable approaches," Proc. ISCAS, pp. 915-918, Rome May 1982.

[114] J. Vandewalle and J. Staar, "On the recursive Identification of multivariable State Space Models," Proc. Symposium on applications of multivariable System Theory, Plymouth, October 1982.

[115] J. Staar, "Some considerations about 'The determination of the rank of a large matrix'," Internal ESAT Report, June 1980.

APPENDIX.

This quick-reference brings together some standard notations, conventions and definitions that are used throughout the text. More specific terminology is defined at its first appearance in the thesis.

List of notations and conventions.

$A_{M,N}$: Matrix A, with M rows and N columns. The subscripts M and N denote the actual dimensions of the considered matrix [so $A^t_{N,M}$ denotes $(A_{M,N})^t$]. A_{mn} denotes a submatrix of A, with entries of m among the M rows and n among the N columns of A. An entry on row i, column j in A is denoted as a(i,j). If matrix is a block-matrix i.e. if it is partitioned into submatrices, then $A_{m,n}(i,j)$ is the mxn submatrix of A, at the block position i,j . If no confusion is possible, the comma or even the dimension-indices are omitted.

$A^t_{M,N}$: The transpose of matrix A, i.e:

$$A^t_{M,N} = B_{M,N} \quad \text{iff} \quad \forall i,j : a(i,j) = b(j,i)$$

$A^T_{M,N}$: Block transpose of matrix A [with blocks $A_{m,n}(i,j)$] :

$$A^T = B \quad \text{iff} \quad \forall i,j : A_{m,n}(i,j) = B_{m,n}(j,i)$$

$I_{m,n}$ Pseudo-unit matrix: upper left mxn submatrix of a sufficiently large unit matrix.

S^i : Linear (sub)space of dimension i.

S^m_U , S^n_V : Linear (sub)spaces of dimension m resp., n , in the domain resp., co-domain of a considered matrix.

$A = U \Sigma V$: Singular Value Decomposition of a matrix A.

σ_p : The p-th singular value of a considered matrix, i.e. (p-th diagonal element in Σ).

u_m, v_n : The m-th left resp., n-th right singular vector in an SVD, i.e. the m-th column in U, resp. the n-th row in V.

S^m_U , S^n_V : m-th left, resp., n-th right principal singular subpaces of a considered matrix:

$$S_U^m = \text{Span} (u_1 , \ldots , u_m)$$

$$S_V^n = \text{Span} (v_1 , \ldots , v_n)$$

$S = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$: System matrix for a linear, discrete time State Space Representation (SSR), denoting the following dynamical relations between the (discrete time) input vector function u(k), and output vector function y(k):

$$x(k+1) = A.x(k) + B.u(k)$$

$$y(k) = C.x(k) + D.u(k)$$

Nomenclature.

Matrix A is orthonormal	:	$A^t A = A A^t = I$
Matrix A is a Hankel matrix	:	$a(i,j) = a(i-1,j+1)$
Matrix A is a Toepliz matrix	:	$a(i,j) = a(i+1,j+1)$