MASTER

The estimation of minimal polynomial coefficients and a start sequence of Markov parameters

van der Weijden, J.L.J.M.

Award date:
1984

4855

DEPARTMENT OF ELECTRICAL ENGINEERING
EINDHOVEN UNIVERSITY OF TECHNOLOGY
Group Measurement and Control

THE ESTIMATION OF MINIMAL POLYNOMIAL COEFFICIENTS AND A START SEQUENCE OF MARKOV PARAMETERS.

by J.L.J.M. van der Weijden

This report is submitted in fulfillment of the requirements for the degree of electrical engineer (M.Sc.) at the Eindhoven University of Technology. The work was carried out from October 1983 until August 1984 as an assignment by:

Prof. dr. P. Eykhoff

under supervision of:

ir. A.C.P.M. Backx (Philips)
dr. ir. A.A.H. Damen
dr. ir. A.K. Hajdasinski
ir. P.M.J. Van den Hof

SUMMARY

The model of a Multi-Input/Multi-Output (MIMO) system can be based on the multivariable impulse responses: the Markov parameters.
The realizability criterion can be used to describe the linear dependences of the Markov parameters on a number of preceding Markov parameters. This expression is called the MINIMAL POLYNOMIAL relation.
In this report two methods are derived for the estimation of the coefficients of this linear relation (the so called minimal polynomial coefficients) and the start sequence of Markov parameters. Both methods are based on the Least Squares principle.
The first method tries to find the minimum of a quadratic loss function (being the squared distance of the estimated Markov parameters to the available noise corrupted set of parameters)by means of an iteration process.
The second algorithm uses a hill climbing procedure to find the optimum of the error function (in this case the squares of the differences between the estimated output signals and the measured signals).
Both procedures are compared with each other by performing many tests on simulated data. The comparison of the described methods with an explicit iterative algorithm that estimates truncated impulse responses is also presented.

1 INTRODUCTION.

During the last twenty years, there has been a progressive increase of the interest in the control of complicated systems. Especially optimal (in some way) control systems have become more and more important, not only in technical systems but also for example in economical and biomedical applications.

Most of those systems have in common to be ´multivariable´ and ´dynamical´. In this case, ´multivariable´ means that the systems have several inputs and several outputs. Further the inputs may influence more than one output at one time (see fig. 1). Systems with these properties are called MIMO systems (Multi Input, Multi Output).

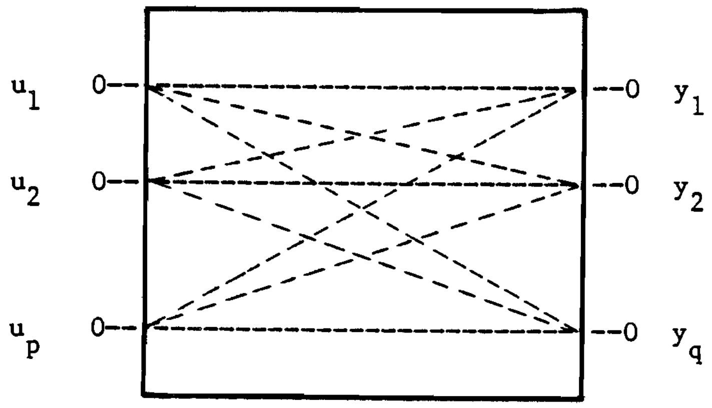


fig. 1: The black box representation of a MIMO system.

Starting point in the design of a control system usually is some model that describes the behaviour of the system under consideration.

The problem of finding some mathematical description of the system is called SYSTEM IDENTIFICATION.

Information that can be used to solve this problem is a priori knowledge of the system and measurements of the input and output signals.

In practice, we always deal with measurements, corrupted with some kind of noise. So, based on these observations, we will try to estimate a model for the system.

When applying estimated models in control techniques, we will usually consider a system in a ´black box´ approach; this means that we are not interested in the internal physical structure but only in the

2 THE GERTH APPROACH WITH SUCCESSIVE SUBSTITUTION.

In this report we will only discuss linear, time invariant, time discrete stable systems. The Markov parameters of such a system represent the sampled impulse response of the system. Because of the uniqueness of the impulse response of a system, Markov parameters give a unique representation of the system. And thus this way of modelling is very attractive for estimation purposes. A great disadvantage of the description in Markov parameters is that the number of the parameters needed in general, is infinite. But for a finite dimensional system in general, we can find some scalars r and a_i, (i=1,2.. r), in a such way that the following recurrent relation holds [7 : theorem 1]:

$$M(r+j) = \sum_{i=1}^{r} a_i \; M(r-i+j) \qquad j \geq 1 \tag{2.1}$$

This implies that the impulse response of a finite dimensional system can be described completely by $\{a_i, M(i)\}_{i=1,2..r}$. We can find a value of r, in such a way that the first r Markov parameters are independent and that all other parameters can be computed from of these start parameters and the coefficients of the recurrent relation. In this way a finite number of parameters can give a description of the infinite impulse response (r is less then or equal to the minimal dimension n of the system, see appendix B).
The minimal value of r, such that relation 2.1 is valid, is called the degree of the minimal polynomial. In case we deal with noise corrupted Markov parameters, the system is not of a finite dimension and thus we cannot find scalars r and a_i of a recurrent relation. We will try to construct a finite dimensional realization from of a finite number of noise corrupted Markov parameters, M(i). We assume to know the degree of the minimal polynomial beforehand. So we compute an approximated sequence of Markov parameters which satisfies equation 2.1 and which is a best fit , in a certain way, on the given sequence. This fit we are looking for, implies that we minimize a loss function S, which is defined in eq. 2.2:

$$S = \sum_{i=1}^{k} \text{trace} \left[\{\hat{M}(i)-\tilde{M}(i)\}^T \cdot \{\hat{M}(i)-\tilde{M}(i)\} \right] \qquad 2.2$$

'k' is the number of given Markov parameters.
Because $\hat{M}(i)$ is not linear in the $\hat{a}_i$ parameters, the loss function will not be a simple quadratic function. Hence, it will, in general, not be possible to find the minimum of S in a closed form. So we will need some iterative (hill-climbing) procedure to determine the minimum of S by varying $\{\hat{M}(i),\hat{a}_i\}_{i=1..r}$. Using the GERTH method [4] an estimation $\{\hat{a}_i, \hat{M}(i)\}_{i=1..r}$, will be found in two steps. First we will minimizing S with respect to $\hat{a}_i$, and during the second part we will find the optimal set of start parameters $\hat{M}(i)$ by the computed $\hat{a}_i$ and the given $\tilde{M}(j)$ (for i=1,2.. r and j=1,2.. k).

2.1 Estimation of the coefficients of the minimal polynomial.

The recursive properties of the Markov parameters (eq. 2.1) can be formulated in a matrix equation. As we deal with noise corrupted Markov parameters, we would like to find minimal polynomial coefficients such that equation 2.3 is true:

$$\begin{bmatrix} \tilde{M}(1) & \tilde{M}(2) & \cdots & \tilde{M}(r) \\ \vdots & & & \vdots \\ \tilde{M}(k-r) & & \cdots & \tilde{M}(k-1) \end{bmatrix} \cdot \begin{bmatrix} a_r I_p \\ \vdots \\ a_1 I_p \end{bmatrix} = \begin{bmatrix} \tilde{M}(r+1) \\ \vdots \\ \tilde{M}(k) \end{bmatrix} \qquad 2.3$$

I_p is a pxp identity matrix (p is the number of inputs of the system under consideration).

If we want to use standard routines of the NAG-library for the computation of the least squares 'solution', eq. 2.3 is not suited for the estimation of the coefficients a_i. These routines all expect, at least if we want to find one value for $\hat{a}_i$, i=1:r, equations of the form like eq. 2.4:

$$H \underline{a} = \underline{v} \qquad 2.4$$

So we introduce a (pxqx(k-r),r) matrix H and a (pxqx(k-r) vector $\underline{v}$ (q is the number of outputs of the system). The matrix H and the vector $\underline{v}$ are composed from the given noise corrupted Markov parameters. This composition can be done in several ways; the one we are going to apply, is not the composition suggested by W. GERTH [4 : page 48], but one that is better suited for our representation of the noise corrupted parameters. In fact by changing the rows, the GERTH composition can be found from the form suggested by eq. 2.5.

$$H_{rk} = \begin{bmatrix} \tilde{M}_{1,1}(1) & \cdots\cdots & \tilde{M}_{1,1}(r) \\ \tilde{M}_{1,2}(1) & \cdots\cdots & \tilde{M}_{1,2}(r) \\ \cdot & & \cdot \\ \tilde{M}_{p,q}(1) & & \tilde{M}_{p,q}(r) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \tilde{M}_{1,1}(k-r) & & \tilde{M}_{1,1}(k-1) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \tilde{M}_{p,q}(k-r) & \cdots\cdots & \tilde{M}_{p,q}(k-1) \end{bmatrix} \qquad \underline{v}_{rk} = \begin{bmatrix} \tilde{M}_{1,1}(r+1) \\ \tilde{M}_{1,2}(r+1) \\ \cdot \\ \tilde{M}_{p,q}(r+1) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \tilde{M}_{1,1}(k) \\ \cdot \\ \cdot \\ \tilde{M}_{p,q}(k) \end{bmatrix} \qquad 2.5$$

$$\underline{a}^T = [\, a_r, a_{r-1}, \ldots\ldots, a_1 \,]$$

$\tilde{M}_{1,k}(i)$ is the (1,k) element of the i^{th} noise corrupted Markov parameter.

The estimation of the minimal polynomial coefficients uses a weighted least squares method considering the loss function in eq. 2.6.

$$J = (H\,\underline{a} - \underline{v}\,)^T\, B\, (H\,\underline{a} - \underline{v}\,) \qquad 2.6$$

with $B^{-1} = \text{cov}\,(\underline{v})$

By minimizing J with respect to $\underline{a}$, we will find the estimation of $\underline{a}$ in eq. 2.7 :

$$\underline{\hat{a}} = (H^T\, B\, H)^{-1}\, H^T\, B\, \underline{v}^T \qquad 2.7$$

If we suppose to deal with white noise corrupted Markov parameters, this implies that $\text{cov}(\underline{v})=I$, we will find a simplified expression for the optimal values $\underline{\hat{a}}$ in eq. 2.8:

$$\underline{\hat{a}} = (H^T\, H)^{-1}\, H^T\, \underline{v}^T \qquad 2.8$$

Instead of solving equation 2.8, we have used a standard routine for finding the least squares solution of eq. 2.4. This routine, F04AMF of the NAG-library [9], reduces matrix H to upper right triangular form, R, by applying Householder transformations, Q, with pivoting so that Q H=R. The right hand side vector V is transformed into vector C by applying the same transformation matrix Q, Q V=C, and an approximate solution X is found by back subsitution in the equation R X=C. The residual vector P=V-H X is computed and a correction D to X is found by solving the linear least squares problem H D=P, i.e. R D = QP. X is replaced by (X+D) and the correction process is repeated until full machine accuracy is obtained. Additional precision accumulation of inner products is used throughout the calculation.

2.2 The estimation of the start sequence $\{\hat{M}(i)\}_{i=1,2..r}$.

The estimation of the start sequence $\{\hat{M}(i)\}_{i=1,2..r}$, will have a great deal in common with the estimation of the minimal polynomial coefficients. We will minimize the loss function (eq. 2.9):

$$S' = \sum_{i=1}^{k} \text{trace}\,\{\hat{M}(i)-\tilde{M}(i)\}^T \cdot \{\hat{M}(i)-\tilde{M}(i)\} \qquad 2.9$$

$\hat{M}(i)_{i=1,2..k}$ is the reconstructed sequence of Markov parameters, based on the already estimated minimal polynomial coeffients and the requested start sequence of Markov parameters.

To simplify the procedure, we define a partial loss function according to eq. 2.10, where we assume to deal with white noise corrupted Markov parameters.

$$S'_{jh} = \sum_{i=1}^{k} (\hat{M}_{j,h}(i)-\tilde{M}_{j,h}(i))^2 \qquad 2.10$$

If we have minimized all partial loss functions, we also have minimized the loss function of eq. 2.9, because all partial systems are mutual independent. Like eq. 2.4 we define eq. 2.11:

$$G\,\underline{m}_{jh} = \underline{n}_{jh} \qquad 2.11$$

$\underline{m}_{jh}$ is a vector with length r and consisting of the (j,h) elements of the requested start series of Markov parameters.

$\underline{n}_{jh}$ is a vector with length k and consisting of the (j,h) elements of the given white noise corrupted Markov parameters.

G is a (k,r) matrix, consisting of polynomials of a_i (i=1,2.. r):

$$G^T = [\ I_{r,r} \mid R\ E_r \mid R^2\ E_r \mid \ldots\ldots \mid R^{k-r}\ E_r\ ] \tag{2.12}$$

with: $$R = \left[\begin{array}{c|c} 0 & a_r \\ \ldots & . \\ I_{r-1,r-1} & . \\ & a_1 \end{array}\right] \quad \text{and} \quad E_r = \begin{bmatrix} 0 \\ . \\ 0 \\ 1 \end{bmatrix}$$

We mention here, that the expression for matrix G in eq.2.12 is not used during the composition of the matrix in the GERTH algorithm (in [12 : subroutine SOL] the implemented composition of matrix G is shown). It is easy to see that the calculation of the required powers of the matrix R, will cause large numerical errors (because R^T is a matrix in a companion form). Like equation 2.6, the expression for the partial loss function S'_{jh} can be formulated as eq. 2.13:

$$J'_{jh} = (G\ \underline{m}_{jh} - \underline{n}_{jh}) \cdot (G\ \underline{m}_{jh} - \underline{n}_{jh}) \tag{2.13}$$

By minimizing this loss function with respect to $\underline{m}_{jh}$, we will find

the estimation $\underline{\hat{m}}_{jh}$ in eq. 2.14:

$$\underline{\hat{m}}_{jh} = (\ G^T\ G)^{-1}\ G^T\ \underline{n}^T_{jh} \tag{2.14}$$

We notice that solving this least squares problem by computing eq. 2.14, seems to be more efficient than using a standard routine. This because the matrix $(G^T\ G)^{-1}\ G^T$ can be used during the estimation of all pxq partial systems.
But the matrix G can be ill conditioned and so the computation of eq. 2.14 can lead to large numerical errors. So instead of solving eq. 2.14, we have used a standard routine for finding the least squares estimation of the set equations (eq 2.11). This is the same procedure as used for the estimation of the coefficients of the minimal polynomial.

2.3 The iteration process.

The original GERTH algorithm executes both estimation procedures for $\hat{a}_i$ and $\hat{M}(i)$ only once. If the loss function would be quadratic in both $\hat{a}_i$ and $\hat{M}(i)$, applying the algorithm one time would lead to the optimum (i.e. the least squares "solution" of $\hat{a}_i$ and $\hat{M}(i)$). As the loss function is not quadratic in $\hat{a}_i$, we are not able to find the minimum by applying the orginal GERTH algorithm, and using an iteration process can bring a solution. The design of a suited iteration scheme is the next subject in this report.

A possible iteration scheme, according to the successive substitution method, is to compose the matrix H_{rk} of eq. 2.5 from a sequence of Markov parameters, $M^j(i)$(M(i) eq. 2), consisting of start sequence of $\hat{M}(i)$ and the expanded sequence of Markov parameters, by using the recurrent relation 2.1 and the estimated coefficients of the minimal polynomial $\hat{a}_i^j$.

In the used notations, the number of j´s in the superscripts "j", "jj", "jj..j" indicate the number of the iteration cycle during which the estimates have been computed.

After computing new estimates of $\hat{a}_i^{jj}$, we can compose a new matrix G (see eq. 2.11) out of $\hat{a}_i^{jj}$ and we can choose to use the already existing $\underline{n}_{jl}$(with the elements $M_{j,l}(i)$) or compose a new $\underline{n}_{jl}$ out of the series $M^j(i)$.

According to J.B. KORTAS [8], who has implemented the GERTH algorithm for SISO systems, the composition of $\underline{n}_{jl}$ out of $M^j(i)$ is not a very stable method, because the start series $M^{jj.j}(i)$ can drift away from the optimal estimates very easy. We can explain this by considering $\underline{n}_{jl}^{jj.j}$ as a reference to the given noise corrupted Markov parameters.

So we have implemented the iteration scheme in which only the H_{rk} and G are adapted during each iteration cycle:

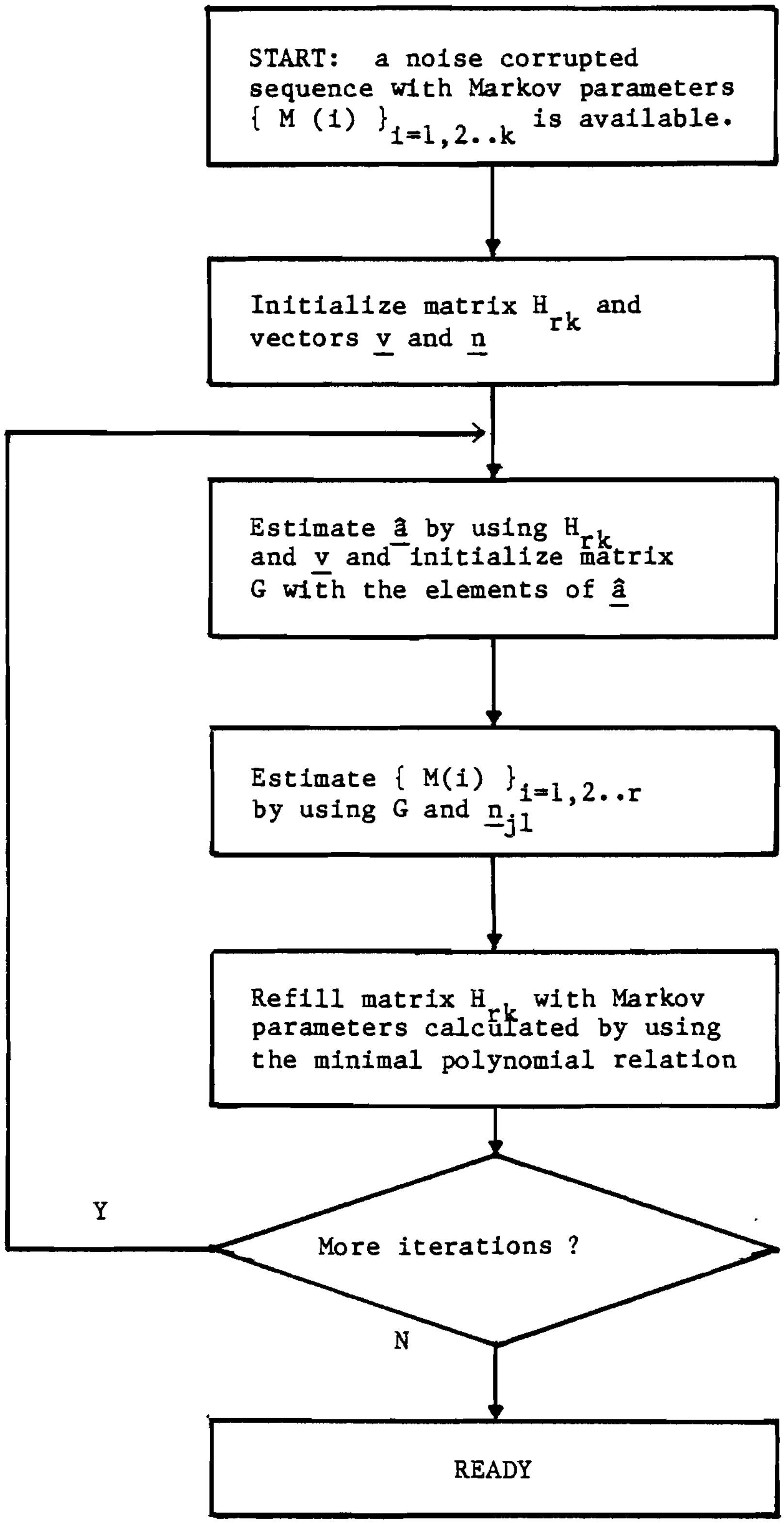


fig. 2: Implemented scheme of the iterative GERTH algorithme.

3 THE DIRECT METHOD.

3.1 An introduction to the DIRECT Method.

In the previous chapter, the GERTH method has been described. This procedure estimates a set of minimal polynomial coefficients and a start series of Markov parameters out of another set of Markov parameters. This last set usually will consist of the samples of estimated, truncated impulse responses.
The GERTH algorithm optimizes the fit of the (reconstructed) Markov parameters (calculated by extension with the coefficients of the minimal polynomial) on the truncated impulse responses. During this optimization we do not reconstruct any output samples from the input measurements, so no reference is made with input and output measurements of the system under consideration.
This means that the result of the GERTH estimation totally depends upon the quality of the already estimated Markov parameters!
In the next section we will derive a method, so called DIRECT method, which estimates minimal polynomial coefficients and a minimal set of Markov parameters from input and output data of the system.

3.2 Derivation of the DIRECT method.

We will use a model in describing the behaviour of the MIMO system that is called the HANKEL model [7: definition 12], which is based on the impulse response of the system. Out of the state space notation, we derive equation 3.1. (c.f. [11: chapter 2]).

$$\underline{y}(k) = C\,A^{k}\,\underline{x}(0) + \sum_{i=1}^{k} C\,A^{i-1}\,B\,\underline{u}(k-i) + D\,\underline{u}(k) \qquad 3.1$$

where: $\underline{x}(0)$ is the initial state of the system.
$\underline{u}(k)$ is the (px1) input vector $\underline{u}$ at time instant k
$\underline{y}(k)$ is the (qx1) output vector $\underline{y}$ at time instant k
A is the (nxn) system matrix, where n is the dimension of the system.
B is the (nxp) distribution matrix.
C is the (qxn) output matrix.
D is the (qxp) input,output matrix.

The matrix products $C.A^{i-1}.B$ are the multivariable impulse responses of our system. The part of equation 3.1 containing the summation may be regarded as a convolution sum of impulse response and all previous and present input signals.

These samples of the multivariable impulse response are called Markov parameters, defined as:

Markov parameter at time instant i:

$$M(i) = \begin{cases} 0 & , i < 0 \\ D & , i = 0 \\ C\,A^{i-1}\,B & , i > 0 \end{cases} \qquad 3.2$$

If we suppose that the initial conditions of the system are zero ($\underline{x}(0)=\underline{0}$) and we use the definitions of equation 3.2, we can rewrite eq. 3.1 to the form of eq. 3.3 .

$$\underline{y}(k) = \sum_{i=0}^{k} M(i) \cdot \underline{u}(k-i) \qquad k = 0\ ,\ 1,\ldots\ldots \qquad 3.3$$

We can rewrite expression 3.3 into a matrix notation:

$$\begin{vmatrix} \underline{y}^T(0) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \underline{y}^T(k) \end{vmatrix} = \begin{vmatrix} \underline{u}^T(0) & 0 & \ldots & \ldots & \ldots & 0 \\ \cdot & & 0 & \ldots & \ldots & 0 \\ \cdot & & & 0 & \ldots & 0 \\ \cdot & & & & 0 \ldots & 0 \\ \cdot & & & & 0 \ldots & 0 \\ \cdot & & & & & 0 \\ \underline{u}^T(k) & \ldots & \ldots & \ldots & \ldots & \underline{u}^T(0) \end{vmatrix} \begin{vmatrix} M^T(0) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ M^T(k) \end{vmatrix} \qquad 3.4$$

Normally (for stable systems) impulse responses always will decrease to zero, if sufficient sample intervals are taken into account. Using this property, we can abstract part of matrices of equation 3.4 and rewrite this like eq. 3.5 (so we will neglect the influence of markov parameters M(i) for i > m).

$$\begin{vmatrix} \underline{y}^T(h) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \underline{y}^T(h+m) \end{vmatrix} = \begin{vmatrix} \underline{u}^T(h) & \ldots & \underline{u}^T(h-m) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \underline{u}^T(h+m) & \ldots & \underline{u}^T(h) \end{vmatrix} \begin{vmatrix} M^T(0) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ M^T(m) \end{vmatrix} \qquad 3.5$$

If we use the minimal polynomial expansion instead of the Markov parameters M(i) (i>r where r is the degree of the minimal polynomial), and rename these parameters to functions of the set $\{a_i, M(i)\}_{i=1,2\ldots r}$ (eq. 3.6: we use a recurrent relation), we will find the matrix expression of eq. 3.7.

$$F^T(1) = \sum_{i=1}^{r} a_i \cdot M^T(r-i+1) = M^T(r+1)$$

$$F^T(j) = \sum_{i=1}^{j-1} a_i \cdot F^T(j-i) + \sum_{i=j}^{r} a_i \cdot M^T(r-i+j) = M^T(r+j) \qquad 3.6$$

$$(\text{for } 1 \leq j \leq r)$$

$$F^T(j) = \sum_{i=1}^{r} a_i \cdot F^T(j-i) = M^T(r+j)$$

$$(\text{for } j > r)$$

$$\underbrace{\begin{vmatrix} \underline{y}^T(h) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \underline{y}^T(h+m) \end{vmatrix}}_{Y} = \underbrace{\begin{vmatrix} \underline{u}^T(h) & \cdots & \underline{u}^T(h-m) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \underline{u}^T(h+m) & \cdots & \underline{u}^T(h) \end{vmatrix}}_{S_m^T} \underbrace{\begin{vmatrix} M^T(0) \\ M^T(1) \\ \cdot \\ M^T(r) \\ F^T(1) \\ \cdot \\ F^T(m-r) \end{vmatrix}}_{M} \qquad 3.7$$

The problem we deal with is the estimation of minimal polynomial coefficients and the first r+1 Markov parameters (M(0) is the input/output matrix). We suppose that we have given a set of input/output data, that may be corrupted with measurement noise, that is not correlated with the input signals and which is assumed to be white and channel independent. So:

$$Y = S_m^T \cdot M + E \qquad 3.8$$

How can we find an estimate $\hat{M}$ which is as close as possible to the real parameters M ? This $\hat{M}$ will satisfy the expression in eq. 3.9 ($\hat{Y}$ consists of the reconstructed output data).

$$\hat{Y} = S_m^T \cdot \hat{M} \qquad 3.9$$

Our target now is to find a matrix $\hat{M}$ which minimizes the error $Y - \hat{Y}$ (in some chosen sense). We will use the least squares criterion to find the minimum.

The idea behind this criterion is that the projection $\underline{\hat{y}}$ of the noise corrupted vector $\underline{y}$ on the Euclidean space formed by the input vectors $\underline{u}$ is the best approximation to the vector $\underline{y}$ (the parameter vector M is mapped into this space by the transformation by S_m^T).
This means that the Euclidean distance between $\underline{y}$ and $\underline{\hat{y}}$ is minimized. In case of a single output system, the loss function would be given by eq. 3.10.

$$V = \sum_{i=1}^{m} (y(i)-\hat{y}(i))^2 = (\underline{y} - \underline{\hat{y}})^T \cdot (\underline{y} - \underline{\hat{y}}) \qquad 3.10$$

But in the multi-input multi-output case, Y and $\hat{Y}$ are matrices built up out of 'q' output signals $\underline{y}$, so the loss function will be:

$$V = \sum_{j=1}^{q} \sum_{i=1}^{m} e_j^{\,2}(l+i) = \text{trace}(E^T \cdot E) = \qquad 3.11$$

$$\text{trace}[(Y - \hat{Y})^T \cdot (Y - \hat{Y})]$$

Thus to find the least squares estimate of the Matrix M, it is necessary to minimize the loss function V with respect to M, containing Markov parameters and higher order functions of Markov parameters and minimal polynomial coefficients.
If we subsitute equation 3.9 into eq. 3.11, we will find:

$$V = \text{trace}[(Y - S_m^T.\hat{M})^T.(Y - S_m^T.\hat{M})] =$$

$$\text{trace}[Y^T.Y] - 2.\text{trace}[Y^T.S_m^T.\hat{M}] + \text{trace}[\hat{M}^T.S_m.S_m^T.\hat{M}] \qquad 3.12$$

In Appendix A the loss function of eq. 3.12 is rewritten to an expression, which can be computed very easily.
The equation error $E = Y - S_m^T\hat{M}$ is not linear in a_i, so V is not square in parameters a_i. This means that we have to apply an optimization method to estimate a_i and M(i).
Most of these methods need at the least first derivatives of the object function (few even ask for second derivatives or calculate estimates of them).
The first partial derivatives of the loss function to the parameters a_i and M(i) have also been derived in Appendix A.

4 SIMULATIONS FOR TESTING THE PROPERTIES OF THE GERTH ALGORITHM.

In order to test the properties of the iterative GERTH algorithm, different systems have been simulated under several conditions. Simulations show the advantage of giving the freedom to influence certain quantities relevant for the GERTH algorithm, such as:

1. The number of estimated Markov parameters $\{\tilde{M}(i)\}$.

2. The level and the kind of the noise disturbances.

3. The eigenvalues of the system.

Fig. 3 shows a block diagram of the operations. Before the GERTH algorithm can be run, we have to estimate a sequence of Markov parameters $\{\tilde{M}(i)\}$. The program EXACTMARK [11 : section 4.2] , an explicit iterative algorithm for the estimation of Markov parameters, is used to find this sequence $\{\tilde{M}(i)\}$.
The input of the estimation program is a sequence of input and output samples of the system under consideration. This sequence can be constructed out of measurements, but in this case, the samples have been generated by a simulation program called SYSSIMUL [10].
In figure 3 we see that we can generate a sequence of output samples with a certain signal-to-noise ratio (the user can select this SN-ratio). We can also define an output noise system {F, G, H}, so the output samples can be corrupted by coloured Gaussian additive noise instead of by white noise.
The input signal is also a white noise sequence (normal distribution) with a zero mean value and a standard deviation of 1. The output noise has the same characteristics, but is not correlated with the input noise. During our analysis, we have generated sequences of 1000 samples for each input and output of the system.

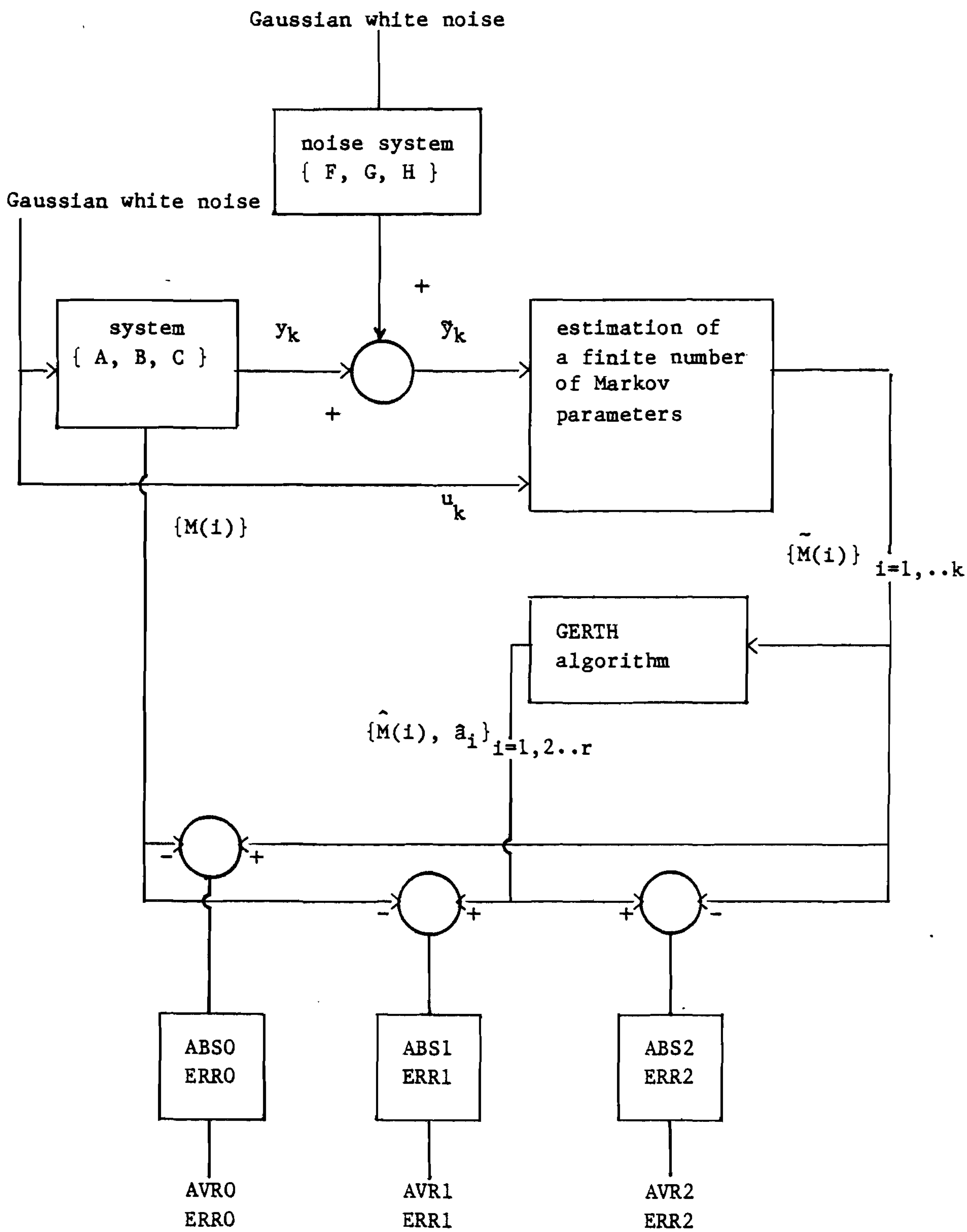


fig 3. Simulation, processing of Markov parameters and reprocessing of minimal polynomial coefficients and a start sequence of Markov parameters.

4.1 Systems and parameter values, used during the tests of the GERTH algorithm.

The results for two systems will be presented, both having 3 inputs, 2 outputs and dimension 4. The state space descriptions of both systems are given in eq. 4.1, eq. 4.2 and eq. 4.3 These systems are also used by VAESSEN [11]. For a clear view on the eigenvalues of the systems, the A matrices are chosen in a diagonal way. The applied noise system is presented in eq. 4.4. Both input-output Matrices D and DN are equal to zero, this means that we have restricted ourselves to strictly proper systems.

$$B=\begin{bmatrix} 1.0 & 0.0 & 1.0 \\ -1.0 & 0.5 & 0.5 \\ 0.0 & 1.0 & -0.5 \\ 0.0 & 0.5 & -1.0 \end{bmatrix} \qquad C=\begin{bmatrix} 0.5 & 0.0 & 1.0 & 1.0 \\ 1.0 & -0.5 & 0.5 & -0.5 \end{bmatrix} \qquad 4.1$$

$$\text{SYSTEM 2:} \quad A = \text{diag}(\ 0.7,\ 0.6,\ 0.2,\ 0.1) \qquad 4.2$$

$$\text{SYSTEM 3:} \quad A = \text{diag}(\ 0.9,\ 0.8,\ 0.3,\ 0.2) \qquad 4.3$$

$$F=\begin{bmatrix} 0.85 & 0.0 \\ 0.0 & 0.75 \end{bmatrix} \qquad G=H=\begin{bmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{bmatrix} \qquad 4.4$$

For both systems we have chosen 2 large and 2 small eigenvalues to be able to look what happens to the smallest eigenvalues after estimation and applying the GERTH algorithm. The largest eigenvalues of SYSTEM 3 are so large that we deal with a very slowly decreasing impulse response.

This implies that we should take into consideration many Markov parameters to avoid any influence of the neglection of the 'tail' of the truncated impulse response. So the number of 8 Markov parameters will be too small to guaranty a neglectible truncation error. We will estimate this value in case of system 3 in appendix D. We will check whether the GERTH algorithm can improve the effects of the truncation. The noise system is a combination of 2 decoupled SISO systems in

parallel and is used in both test cases.
Further we have constructed sequences with output signal-to-noise ratio's (SN) of:

1. 30 dB.
2. 10 dB.
3. 0 dB.

Out of these sequences, EXACTMARK has estimated 8 Markov parameters $\{\tilde{M}(i)\}$.
The GERTH algorithm estimates an optimal(in least squares sense) set of startparameters of the Markov series and a optimal set of minimal polynomial coefficients. Out of these parameters, the algorithm calculates a certain number of reconstructed Markov parameters $\{\hat{M}(i)\}$.
In our simulation, we have chosen a number of 30 parameters, so the most significant part of the tail of the impulse responses will be described. The number of iterations in the GERTH algorithm has also been varied during our reprocessing of the Markov parameters and the minimal polynomial coefficients.
We will examine the behaviour of the algorithm by calculating the relative errors ERR0, ERR1 and ERR2. ERR0 and ERR1 are defined as the quotients of the maximal values of the energy in the equation error and the deterministic sequence of Markov parameters (see eq. 4.5 and eq. 4.6). ERR2 is the quotient of the maximal energy in the error and the available (noise corrupted) sequence of Markov parameters $\{\tilde{M}(i)\}$ (see eq. 4.7.). We mention that ERR0 is a measure for the maximal disturbance of the available Markov parameters, ERR1 is a measure for the largest error in the results of the iterative GERTH algorithm.
The deterministic sequence $\{M(i)\}$ has been calculated from the state space description presented by matrices {A, B, C, D}. In eq. 4.5, 4.6, 4.7, 4.8, 4.9 and 4.10, $|\ |_{\infty}$ stands for the maximum norm.

$$\text{ERR0} = \frac{\sum_{i=1}^{N_1} | \tilde{M}(i) - M(i) |_\infty^2}{\sum_{i=1}^{N_1} | M(i) |_\infty^2} \qquad 4.5$$

$$\text{ERR1} = \frac{\sum_{i=1}^{N_2} | \hat{M}(i) - M(i) |_\infty^2}{\sum_{i=1}^{N_2} | M(i) |_\infty^2} \qquad 4.6$$

$$\text{ERR2} = \frac{\sum_{i=1}^{N_1} | \hat{M}(i) - \tilde{M}(i) |_\infty^2}{\sum_{i=1}^{N_1} | \tilde{M}(i) |_\infty^2} \qquad 4.7$$

We mention that ERR2 shows the maximal distance between the noise corrupted sequence of Markov parameters of EXACTMARK $\{\tilde{M}(i)\}$ and the extended series of Markov parameters of the GERTH algorithm $\{\hat{M}(i)\}$. This is the error value we want to minimize during the iterations in the GERTH algorithm, but minimizing ERR2 does not mean that we also minimize the distance between the sequence of exact Markov parameters and the GERTH extension (= ERR1)!
We also will calculated a number of absolute error values over a constant number of parameters N_1, so that we will be able to compare the results of the GERTH algorithm $\{\hat{M}(i)\}$ with the output of the estimation algorithm EXACTMARK $\{\tilde{M}(i)\}$. The definitions of these absolute error values are given in eq. 4.8, eq. 4.9 and eq. 4.10, and are according to the definitions in Van den HOF [16], but Van den HOF has used the Euclidean norm and we have calculated the maximum norm! By using the maximum norm we will get an idea of the worst case situation, instead of the mean distance.

$$AVR0 = \sum_{i=1}^{N_1} | \tilde{M}(i) - M(i) |^2_\infty \qquad 4.8$$

$$AVR1 = \sum_{i=1}^{N_1} | \hat{M}(i) - M(i) |^2_\infty \qquad 4.9$$

$$AVR2 = \sum_{i=1}^{N_1} | \hat{M}(i) - \tilde{M}(i) |^2_\infty \qquad 4.10$$

4.2 The results of the simulations on the GERTH algorithm.

During the simulations, we have generated 20 different input noise data sets. In the GERTH algorithm a test is executed to check whether one more iteration still improved the fit of the extended sequence Markov parameters on the estimated set (out of EXACTMARK). We have calculated the average values of the error taking the error values of the results of equivalent tests with different input data. During the calculation of the average values of the errors only the useful results have been used.

If all twenty runs take place, the mean error value in the next tables will be followed by a ´*´ notation.

In these tables, a ´-´ notation indicates that during all calculations on a system, the desired number of iterations of the GERTH algorithm did not reduce the value of the error function ERR2 any further.

The notation $ERR0_{30}$ indicates the mean value of ERR0 during simulations with signals with signal-to-noise ratio SN of 30 dB.

As mentioned in previous chapter: N_1=8 and N_2=30!

4.2.1 The results of the simulations, using System 2.

		DEGREE=2		DEGREE=3		DEGREE=4	
		ERR1	ERR2	ERR1	ERR2	ERR1	ERR2
	it=1	.13 E-1*	.25 E-1*	.14 E-1*	.17 E-1*	.18 E-1*	.13 E-1*
	2	.12 E-1*	.24 E-1*	.15 E-1*	.17 E-1*	.19 E-1*	.12 E-1*
SN=00	3	.10 E-1	.22 E-1	.24 E-1	.17 E-1	.92 E+0	.12 E-1
	4	.10 E-1	.22 E-1	.22 E-1	.15 E-1	.23 E-1	.11 E-1
	5	.97 E-2	.21 E-1	.15 E-1	.15 E-1	.30 E-1	.11 E-1
	it=1	.20 E-2*	.29 E-2*	.18 E-2*	.20 E-2*	.20 E-2*	.15 E-2*
	2	.21 E-2	.29 E-2	.19 E-2*	.20 E-2*	.21 E-2*	.14 E-2*
SN=10	3	.19 E-2	.29 E-2	.19 E-2	.20 E-2	.23 E-2	.14 E-2
	4	.18 E-2	.28 E-2	.18 E-2	.18 E-2	.30 E-2	.14 E-2
	5	.18 E-2	.28 E-2	.19 E-2	.18 E-2	.26 E-2	.16 E-2
	it=1	.89 E-3*	.77 E-3*	.14 E-3*	.90 E-4*	.10 E-3*	.40 E-4*
	2	.86 E-3*	.74 E-3*	.84 E-3	.73 E-3	.11 E-3*	.44 E-4*
SN=30	3	.85 E-3*	.74 E-3*	-	-	.82 E-4	.23 E-4
	4	.85 E-3*	.74 E-3*	-	-	-	-
	5	.84 E-3	.73 E-3	-	-	-	-

Remarks: * :all 20 different runs make sense.
- :no runs make sense.

table 1: Relative errors ERR1 and ERR2 during simulations on SYSTEM 2 and no noise-colouring:

$ERR0_{30}$= 0.87 E-4 $ERR0_{10}$= 0.30 E-2 $ERR0_{00}$= 0.29 E-1

		DEGREE=2	DEGREE=3	DEGREE=4
		AVR1	AVR1	AVR1
	it=1	0.52 E-1*	0.57 E-1*	0.70 E-1*
	2	0.45 E-1*	0.53 E-1*	0.67 E-1*
SN=00	3	0.39 E-1	0.52 E-1	0.64 E-1
	4	0.39 E-1	0.52 E-1	0.63 E-1
	5	0.38 E-1	0.52 E-1	0.58 E-1
	it=1	0.76 E-2*	0.70 E-2*	0.78 E-2*
	2	0.78 E-2	0.70 E-2*	0.76 E-2*
SN=10	3	0.70 E-2	0.71 E-2	0.78 E-2
	4	0.68 E-2	0.69 E-2	0.85 E-2
	5	0.69 E-2	0.72 E-2	0.91 E-2
	it=1	0.33 E-2*	0.45 E-3*	0.33 E-3*
	2	0.32 E-2*	0.52 E-3	0.34 E-3*
SN=30	3	0.32 E-2*	-	0.24 E-3
	4	0.32 E-2*	-	-
	5	0.31 E-2	-	-

Remarks: * :all 20 different runs make sense.
- :no runs make sense.

table 2: Absolute error AVR1 during simulations on SYSTEM 2 and no noise colouring:
$AVRO_{30}$= 0.35 E-3 $AVRO_{10}$= 0.12 E-1 $AVRO_{00}$= 0.12 E+0

		DEGREE=2		DEGREE=3		DEGREE=4	
		ERR1	ERR2	ERR1	ERR2	ERR1	ERR2
	it=1	.24 E-2*	.15 E-2*	.21 E-2*	.80 E-3*	.23 E-2*	.49 E-3*
	2	.24 E-2	.15 E-2	.21 E-2	.84 E-3	.23 E-2	.50 E-3
SN=10	3	.24 E-2	.14 E-2	.22 E-2	.62 E-3	.13 E-1	.42 E-3
	4	.23 E-2	.14 E-2	.23 E-2	.49 E-3	.29 E-2	.40 E-3
	5	.23 E-2	.14 E-2	.24 E-2	.36 E-3	.95 E-2	.32 E-3
	it=1	.91 E-3*	.75 E-3*	.14 E-3*	.50 E-4*	.93 E-4*	.18 E-4*
	2	.88 E-3*	.73 E-3*	.15 E-3	.66 E-4	.10 E-3	.22 E-4
SN=30	3	.87 E-3*	.72 E-3*	-	-	.13 E-3	.25 E-4
	4	.86 E-3*	.72 E-3*	-	-	-	-
	5	.86 E-3*	.71 E-3	-	-	-	-

Remarks: * :all 20 different runs make sense.

\- :no runs make sense.

table 3: Relative errors ERR1 and ERR2 during simulations on SYSTEM 2 and NOISE SYSTEM 1: $ERRO_{30}$= 0.81 E-4 $ERRO_{10}$= 0.26 E-2

		DEGREE=2	DEGREE=3	DEGREE=4
		AVR1	AVR1	AVR1
	it=1	0.94 E-2*	0.80 E-2*	0.86 E-2*
	2	0.93 E-2	0.80 E-2	0.84 E-2
SN=10	3	0.94 E-2	0.83 E-2	0.81 E-2
	4	0.89 E-2	0.84 E-2	0.83 E-2
	5	0.90 E-2	0.85 E-2	0.90 E-2
	it=1	0.33 E-2*	0.43 E-3*	0.32 E-3*
	2	0.33 E-2*	0.50 E-3	0.33 E-3
SN=30	3	0.33 E-2*	-	0.42 E-3
	4	0.32 E-2*	-	-
	5	0.32 E-2*	-	-

Remarks: * :all 20 different runs make sense.

- :no runs make sense.

table 4: Absolute error AVR1 during simulations on SYSTEM 2 and NOISE SYSTEM 1: $AVR0_{30}$= 0.33 E-3 $AVR0_{10}$=0.10 E-2

4.2.2 The results of the simulations, using System 3.

		DEGREE=2		DEGREE=3		DEGREE=4	
		ERR1	ERR2	ERR1	ERR2	ERR1	ERR2
	it=1	.53 E+0*	.28 E-1*	.66 E+0*	.18 E-1*	.69 E+0*	.12 E-1*
	2	.66 E+0*	.25 E-1*	.72 E+0*	.18 E-1*	.71 E+0*	.12 E-1*
SN=00	3	.70 E+0	.24 E-1	.71 E+0	.17 E-1	.75 E+0	.12 E-1
	4	.73 E+0	.24 E-1	.76 E+0	.17 E-1	.73 E+0	.10 E-1
	5	.67 E+0	.22 E-1	.84 E+0	.16 E-1	.71 E+0	.10 E-1
	it=1	.66 E+0*	.41 E-2*	.66 E+0*	.31 E-2*	.65 E+0*	.22 E-2*
	2	.69 E+0	.42 E-2	.68 E+0	.31 E-2	.66 E+0*	.21 E-2*
SN=10	3	.73 E+0	.43 E-2	.67 E+0	.33 E-2	.12 E+2	.23 E-2
	4	.73 E+0	.38 E-2	.71 E+0	.35 E-2	.72 E+0	.21 E-2
	5	.66 E+0	.40 E-2	.76 E+0	.29 E-2	.73 E+0	.21 E-2
	it=1	.68 E+0*	.16 E-2*	.67 E+0*	.84 E-3*	.66 E+0*	.46 E-3*
	2	.69 E+0*	.16 E-2*	.67 E+0	.10 E-2	.66 E+0	.52 E-3
SN=30	3	.63 E+0	.13 E-2	.63 E+0	.31 E-3	.66 E+0	.41 E-3
	4	.57 E+0	.20 E-2	-	-	.66 E+0	.40 E-3
	5	.57 E+0	.19 E-2	-	-	.66 E+0	.40 E-3

Remarks: * :all 20 different runs make sense.

- :no runs make sense.

table 5: Relative errors ERR1 and ERR2 during simulations on SYSTEM 3 and no noise colouring:

$ERRO_{30}$= 0.43 E+0 $ERRO_{10}$= 0.43 E+0 $ERRO_{00}$= 0.47 E+0

		DEGREE=2	DEGREE=3	DEGREE=4
		AVR1	AVR1	AVR1
	it=1	0.17 E+1*	0.18 E+1*	0.19 E+1*
	2	0.18 E+1*	0.19 E+1*	0.19 E+1*
SN=00	3	0.18 E+1	0.18 E+1	0.19 E+1
	4	0.18 E+1	0.19 E+1	0.20 E+1
	5	0.17 E+1	0.19 E+1	0.20 E+1
	it=1	0.18 E+1*	0.18 E+1*	0.18 E+1*
	2	0.18 E+1	0.18 E+1	0.18 E+1*
SN=10	3	0.19 E+1	0.18 E+1	0.17 E+1
	4	0.19 E+1	0.19 E+1	0.20 E+1
	5	0.17 E+1	0.20 E+1	0.20 E+1
	it=1	0.18 E+1*	0.18 E+1*	0.18 E+1*
	2	0.18 E+1*	0.18 E+1	0.18 E+1
SN=30	3	0.17 E+1	0.17 E+1	0.17 E+1
	4	0.15 E+1	-	0.17 E+1
	5	0.15 E+1	-	0.17 E+1

Remarks: * :all 20 different runs make sense.
- :no runs make sense.

table 6: Absolute error AVR1 during simulations on SYSTEM 3 and no noise colouring:

$AVR0_{30}$= 0.18 E+1 $AVR0_{10}$=0.17 E+1 $AVR0_{00}$= 0.19 E+1

		DEGREE=2		DEGREE=3		DEGREE=4	
		ERR1	ERR2	ERR1	ERR2	ERR1	ERR2
	it=1	.66 E+0*	.38 E-2*	.67 E+0*	.29 E-2*	.67 E+0*	.22 E-2*
	2	.68 E+0	.38 E-2	.68 E+0	.29 E-2	.68 E+0*	.21 E-2*
SN=10	3	.69 E+0	.37 E-2	.73 E+0	.28 E-2	.13 E+1	.23 E-2
	4	.59 E+0	.45 E-2	.76 E+0	.21 E-2	.76 E+0	.21 E-2
	5	.50 E+0	.41 E-2	.76 E+0	.21 E-2	.77 E+0	.21 E-2
	it=1	.78 E+0*	.22 E-2*	.74 E+0*	.16 E-2*	.72 E+0*	.83 E-3*
	2	.68 E+0	.21 E-2	.74 E+0	.17 E-2	.73 E+0	.11 E-2
SN=30	3	.13 E+1	.21 E-2	.70 E+0	.19 E-2	-	-
	4	.68 E+0	.21 E-2	.66 E+0	.22 E-2	-	-
	5	.70 E+0	.16 E-2	.66 E+0	.20 E-2	-	-

Remarks: * :all 20 different runs make sense.

- :no runs make sense.

table 7: Absolute error AVR1 during simulations on SYSTEM 3 and NOISE SYSTEM 1: $ERRO_{30}$= 0.43 E+0 $ERRO_{10}$= 0.43 E+0

		DEGREE=2	DEGREE=3	DEGREE=4
		AVR1	AVR1	AVR1
	it=1	0.18 E+1*	0.18 E+1*	0.18 E+1*
	2	0.18 E+1	0.18 E+1	0.18 E+1*
SN=10	3	0.18 E+1	0.19 E+1	0.19 E+1
	4	0.16 E+1	0.20 E+1	0.18 E+1
	5	0.14 E+1	0.20 E+1	0.18 E+1
	it=1	0.19 E+1*	0.19 E+1*	0.19 E+1*
	2	0.19 E+1	0.19 E+1	0.19 E+1
SN=30	3	0.18 E+1	-	0.20 E+1
	4	0.17 E+1	-	0.20 E+1
	5	0.17 E+1	-	0.20 E+1

Remarks: * :all 20 different runs make sense.

- :no runs make sense.

table 8: Absolute error AVR1 during simulations on SYSTEM 3 and NOISE SYSTEM 1: $AVRO_{30}$= 0.19 E+1 $AVRO_{10}$=0.17 E+1

4.2.3 Some remarks on the results of all simulations.

A general remark we can make, based on these tests, is that applying the implemented iteration process, does not make any sense! In fact, we do not know what happens during the substitutions of the estimated values $\{\hat{a}_i, \hat{M}(i)\}_{i=1,..r}$. Some times a very small improvement is found by using the iteration process.

It is quite obvious that the best fit of $\{\hat{M}(i)\}$ on $\{\tilde{M}(i)\}$ is found by increasing the degree of the minimal polynomial because by increasing this degree we will introduce a larger space for all possible solutions.

If we compare the deviations of the tests using system 2, we see that the noise colouring of the additive output noise has a neglectible influence on the results of the GERTH algorithm as well as on the results of the estimation program, EXACTMARK (we compare the situations of white output noise and coloured output noise). We will confine ourselves by examining the results of the simulations without noise colouring (we did not execute the simulations with output noise colouring during the tests with signals with SN = 0 dB).

Comparing the measurements of the fit of $\{\tilde{M}(i)\}$ and $\{\hat{M}(i)\}$ on the exact values of the Markov parameters, we have to note that the relative error values ERR0 and ERR2 are calculated out of the first 8 Markov parameters, while the calculations of ERR1 take the first 30 parameters in consideration. The computation of the absolute error value is executed by using only eight parameters, so we can use these error values to compare the output of the GERTH algorithm with the results of the estimation program EXACTMARK.

<u>The choice of the degree of the minimal polynomial of SYSTEM 2 (tabel 1 and tabel 2).</u>

As we have said before, increasing the degree of the minimal polynomial implies a better fit of $\{\hat{M}(i)\}$ on $\{\tilde{M}(i)\}$, expressed by ERR2 and AVR2. But this does not automatically implies a better fit of $\{\hat{M}(i)\}$ on the exact parameters $\{M(i)\}$.

In case of well conditioned measurements (high values of the signal--to-noise ratio), the distance between $\{\tilde{M}(i)\}$ and $\{M(i)\}$ is rather small, so the deviation between $\{\hat{M}(i)\}$ and $\{M(i)\}$ will also be rather small in this case. In the results of the tests with SN = 30 dB, we find indeed that the minimal value of ERR1 and AVR1 will occur in the cases with a minimal polynomial of degree=4. The order of both systems under consideration is 4!

By applying output noise with a higher energy contents (SN = 10 dB and SN = 0 dB), the optimal fit on the exact parameters has occurred in simulations with degree = 3.

The results of SYSTEM 2 presented in tabel 1 and tabel 2.

In the situation with SN = 30 dB, we have registrated (taken the mean value over 20 different simulations) AVR0 = $0.35\ 10^{-3}$ and AVR1 ≃ $0.33\ 10^{-3}$ (number of iterations it=1 and degree of the minimal polynomial r=4). This implies that we have found only a small improvement of the absolute error value over the first eight Markov parameters.

In cases of SN = 10 dB, we have found values AVR0 = $0.12\ 10^{-1}$ and AVR1 ≃ $0.76\ 10^{-2}$ (it=1, r=2), this implies an error reduction of almost 40%. During the simulations with SN = 0 dB, the value AVR1 ≃ $0.52\ 10^{-1}$ (it=1, r=1) has been calculated. AVR0 is 0.12 during these tests. So we have found a large improvement (50% !) by executing the GERTH algorithm on the results of the estimation program EXACTMARK.

We also mention that we have calculated mean values ERR0 = $0.30\ 10^{-2}$ and ERR1 = $0.20\ 10^{-2}$ during the simulations with SN = 10 dB (and it=1, r=2), and values ERR0 = $0.29\ 10^{-1}$ and ERR1 =$0.13\ 10^{-1}$ for SN = 0 dB (it=1, r=2). So the relative error of $\{\tilde{M}(i)\}$, over 8 Markov parameters is much higher than the relative error in $\{\hat{M}(i)\}$, computed over 30 parameters. In case of SN = 30 dB, we found the following values: ERR0 = $0.87\ 10^{-4}$ and ERR1 = $1.03\ 10^{-4}$ (it=1, r=4).

The choice of the degree of the minimal polynomial of SYSTEM 3 (tabel 5 and tabel 6).

Now we will have a look on the results of simulations with SYSTEM 3. The placed remark on the output noise colouring can also be placed in these situations: we will only consider white noise corrupted measurements.

As we have found during simulations with SYSTEM 2, the best fit of $\{\hat{M}(i)\}$ on $\{\tilde{M}(i)\}$ will occur in the situations with a minimal polynomial of degree 4. In case of SN = 30 dB, this choice of the degree of the minimal polynomial also proved to be the best fit of $\{\hat{M}(i)\}$ on the exact Markov parameters.

We remark that in these cases (SN = 30 dB), allowing one extra iteration the values ERR1 and ERR2 are still improving!
During the simulations with an output SN-ratio of 10 dB and 0 dB, we have computed the smallest deviation between $\{\hat{M}(i)\}$ and $\{M(i)\}$ if degree has been chosen 2. However the differences between ERR1(r=2), ERR1(r=3) and ERR1(r=4) are very small.

The results of SYSTEM 3 presented in tabel 5 and tabel 6.

Comparing the various values of the relative and absolute errors during the tests with SN = 30 dB with the corresponding values from of the simulations with SN = 10 dB and 0 dB, we note that there are only small deviations between the error values. This property occures during simulations with white output noise as well as with output noise colouring.
In appendix D, we will compare the influence of the additive output noise with the effects of the truncation of the impulse response during the execution of EXACTMARK by estimating only 8 Markov parameters.
If we have a look at the absolute error values, we will find in case of SN = 30 dB an AVR0 value of 1.75, while the optimal value of AVR1 is 1.76 (it=1, r=4).
If SN = 10 dB, AVR0 = 1.74 and AVR1 = 1.75 (it=1, r=4) are registrated.
In both situations applying the GERTH algorithm reduces the fit of the first eight estimated Markov parameters on the exact values!
Despite of this slight worsening, it still is useful to apply the GERTH algorithm. If we do not use this algorithm, we have at our disposal only an finite number of Markov parameters (8 in this case). This means that the middle and low frequency responses of a system under consideration can not or hardly be expressed. If the GERTH algorithm can make an extension of the finite set of parameters without a substantial reduction of the accuracy of the original finite set, we have gained a lot: we are able to make an reasonable estimation for the tail of the impulse response of a system.
During the simulations with SN = 0 dB, the mean values AVR0 = 1.91 and AVR1 = 1.67 (it=1) are found while the degree of the minimal polynomial has been taken 2. In this situation a improvement of ≈ 20% has taken place.

CONCLUSION:

In general, by applying the GERTH algorithm we are able to make a reasonable estimation of the complete impulse response of a system. If the start series $\{\tilde{M}(i)\}$ have a good fit on the exact values, we can use a degree of the minimal polynomial which is nearby the real order of the system. In cases with a low signal-to-noise ratio, it is recommended to use a lower order system to estimate the values $\{\hat{a}_i, \hat{M}(i)\}_{i=1,..r}$. Also it makes no sense to apply more than one iteration, because the fit of $\{\hat{M}(i)\}$ on $\{\tilde{M}(i)\}$ is mostly reducing although the fit of $\{\hat{M}(i)\}$ on the exact parameters is sometimes improving. In noisy situations, the GERTH estimation is also smoothing the somewhat stochastic pattern of the estimated Markov parameters. This can be explained by the fact that during the estimation of the parameters in EXACTMARK (or any other estimation program implemented by J.VAESSEN [11]) all entries of Markov parameters are considered as independent parameters (so no attention is paid to the mutual relation of the Markov parameters of a certain system).

5 SIMULATIONS FOR TESTING THE DIRECT METHOD.

During the tests on the properties of the DIRECT method several systems have been taken under consideration. Some tests on SYSTEM 2 and 3 (see chapter 4) have been performed. The DIRECT method estimates the output signals by convolving preceding and actual input measurements with a sequence of Markov parameters, calculated by applying the minimal polynomial relation with estimates of $\{a_i, M(i)\}_{i=1,2..r}$.
The number of measurements in the past that should be taken into account, depends upon the eigenvalues of the system (so on the rate of decrease of the impulse response).
Using systems like SYSTEM 3 with the largest eigenvalue of about 0.9, the impulse responses will decrease slowly. Almost 400 samples will be needed to have a "truncation" error of the same order as the accuracy of the computer.
The optimization methods also ask for the partial derivatives of the loss function to the minimal polynomial coefficients and to the start sequence of Markov parameters. During these calculations the whole data set has to be used so the iterations will need a lot of computational efforts. For example the process time needed to execute only <u>one</u> iteration (with 500 preceding input signal samples) is almost 15 minutes. The average value of the number of iterations during <u>one</u> minimization is ≈ 400. So the total process time will be almost 4 days!
It will not be possible (using this algorithm) to execute a lot of tests on systems with large eigenvalues.
We also want to test the properties of the DIRECT method on systems with eigenvalues that lie close to each other, and that represent an equal part of energy in the impulse response. We have chosen SYSTEM 4:

$$A = \text{diag}(0.08, 0.60, 0.10, 0.55, 0.50)$$

$$B = \begin{vmatrix} 1.0 & 0.0 & -1.0 \\ 1.0 & 1.7 & 0.0 \\ 0.0 & 1.0 & -1.5 \\ 7.0 & 0.0 & 8.0 \\ 0.0 & 7.0 & 5.0 \end{vmatrix}$$

$$C = \begin{vmatrix} 4.0 & 3.0 & 0.0 & 0.0 & 0.0 \\ 9.0 & 1.0 & 8.0 & -0.5 & 0.5 \end{vmatrix} \qquad 5.1$$

$$D = \begin{vmatrix} 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \end{vmatrix}$$

The largest eigenvalues of this (strictly proper) system are rather small; we will only need 200 Markov parameters to have a truncation error of the same order as the accuracy of the VAX-computer. Using this system one iteration takes about 20 seconds processor time, so for one minimization run the computer will need about one hour.
We have used two minimization methods:

1. Conjugate gradient method [20]

2. Comprehensive quasi Newton method [21]

The loss function (the trace of the output error matrix) will decrease very slowly and can have several areas with a small gradient. In most tests the conjugate gradient method has not been able to exceed these "terraces". The comprehensive quasi Newton method will perform a local search if the conditions for an optimum have been met (or almost met). The term "comprehensive" implies in this case that the second derivatives have been estimated instead of calculated. The Newton minimization could reach the global minimimum in most of the performed tests. So we will only discuss tests using the comprehensive quasi Newton method.
We want to compare the results of the test of the DIRECT method with the results of the estimation method EXACTMARK [11] and the GERTH algorithm. We have used the estimates $\{\hat{a}_i, \hat{M}(i)\}_{i=1,2..r}$ of the GERTH method as start parameters in the DIRECT method. By inserting these start parameters we will reduce the time needed.

So the "flow chart" of fig. 4.1 (chapter 4) will be extended with an extra block containing the DIRECT method (simplified):

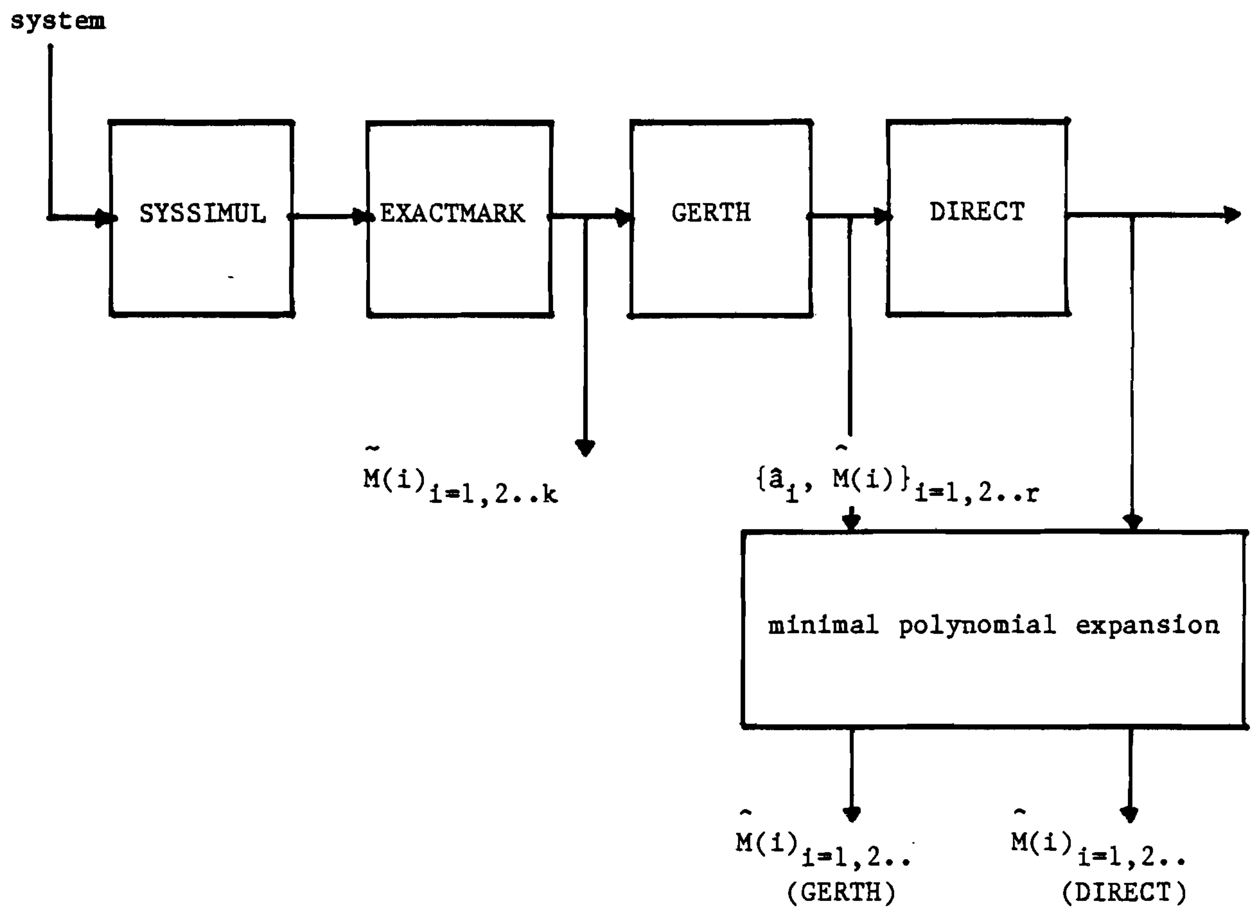


Fig.5.1 Simulations, processing of noise corrupted Markov parameters $\tilde{M}(i)$, the GERTH approximation and the DIRECT estimation of the minimal polynomial coefficients and the start sequence of Markov parameters.

We have used data sequences corrupted with additive output noise consisting of white channel independent noise samples and output signal-to-noise ratio´s (SN) of:

1. 100 dB.
2. 80 dB.
3. 60 dB. 5.2
4. 40 dB.
5. 20 dB.

During our tests with the GERTH algorithm we also have used data with SN less than 20 dB. But during these tests the data set consisted of 1000 measurements instead of 400 samples during the tests with the

DIRECT method.

Using data sets of different SN's we have calculated the energies of the additive output noise. These energies will be the minima of the trace of the output error matrix. In case of SN is 20 dB, the optimal value of the trace almost has been found during the first iteration of the DIRECT method (so by applying the GERTH estimates in the calculation of the trace of the output error matrix). So it makes no sense (using SYSTEM 4) to apply the DIRECT method (as it is implemented here!) on data sets with signal-to-noise ratios less than 20 dB.

Like we have done in chapter 4, we will calculate the relative squared distance between the deterministic and the estimated Markov parameters. So we have calculated ERR0 (eq. 4.5), ERR1 (eq. 4.6) with $\hat{M}(i)$ from the GERTH algorithm and ERR1 with $\hat{M}(i)$ from the DIRECT method.

In appendix F the error values of the same tests will be presented but then we will use the Euclidean norm instead of the Maximum norm, so these values can be used in the comparison with other estimation methods. We have to remark that the differences between the results expressed in Maximum norm and the results in Euclidean norm are rather small.

During our tests the estimation algorithm EXACTMARK has been used to calculate only the first 8 Markov parameters (M(0) has not being taken in account). First we will present the error values of the results with N_1 (number of Markov parameters in ERR0) and N_2 (number of Markov parameters in ERR1) of 8.

The order of SYSTEM 4 is 5 so we have to apply a minimal polynomial extension to calculate Markov parameter 6,7 and 8 by using the estimates $\{\hat{a}_i,\hat{M}(i)\}_{r=1,2..r}$ of the GERTH algorithm and the DIRECT estimation.

In the second part we will present the error values of the results with N_1 and N_2 both 50 (after 50 samples the impulse response of SYSTEM 4 is $\approx 10^{-6}$). The estimates of minimal polynomial coefficients and start sequence of Markov parameters of the GERTH method and the DIRECT estimation will be used to calculate these 45 Markov parameters (50 - the degree of the minimal polynomial). But in EXACTMARK we have estimated only 8 Markov parameters, so the impulse response has been truncated after 8 samples. So in the calculation of ERR0 we will have to assume that the Markov parameters "9" till "50" are zero!

Remark: The degree of the minimal polynomial of SYSTEM 4 is 5 and this value has been used during all tests and in the GERTH algorithm only one iteration has been made.
Like the tests on the GERTH algorithm during the simulations we have generated 20 different noise data sets (for the excitating input noise and for the additive output noise) for the tests with the same signal-to-noise ratio.
In the following bar graphs we will present the mean value of the error values ERR0, ERR1 (GERTH) and ERR1 (DIRECT), averaged over the results of all 20 tests with different excitation and additive output noise corruption. The standard deviation of the error signals also will be presented in the same graphs!
During the calculation of the mean values and standard deviations of the error values, the standard deviation of the results of the simulations on signals with signal-to-noise ratios of 20 dB , 100 dB (DIRECT method) and 20 dB (GERTH algorithm) was very large. During the DIRECT estimation (and both noise levels) the trace after one minimization was significant larger (x 10^{+5} !) than the trace values of the other (19) minimizations. The GERTH method (SN = 20 dB) proposes a set of minimal polynomial coefficients belonging to a system that is not stable!
The reason of this deviation during the test with SN = 100dB can be the restart of the minimization after the restart of the computer system after a "lock" situation. In case of a noise corruption of 20 dB the deviations have been found during the estimations on the same data set!
In the calculation of the mean values and the standard deviations, these 3 "bad" cases have not been considered so the statistic calculations have been executed over the results of 19 minimizations.
Remark: the DIRECT method only uses the second half of the output signals while EXACTMARK (and so GERTH algorithm) examines the complete set of output measurements. The noise reduction during the DIRECT estimation will be less than during simular tests with both other methods. During tests with much noise corruption on the data, the variances of the error values of the DIRECT method will be greater than the variances of the error in the results of EXACTMARK and GERTH algorithm.
We will present the results of minimization on data sequences with a signal-to-noise ratio of 20 dB first, because the results of EXACTMARK

(and so the results of the GERTH method) on signals with SN = 60dB, SN = 80dB and SN = 100dB will remain the same, while the results of the DIRECT method will still decrease very much by increasing SN's.
If we want to present these small error values, the errors of EXACTMARK and the GERTH method will exceed the plots!
In the legends on the bar graphs "s" indicates the standard deviation.

5.1 Results of the tests on signals with SN = 20 dB.

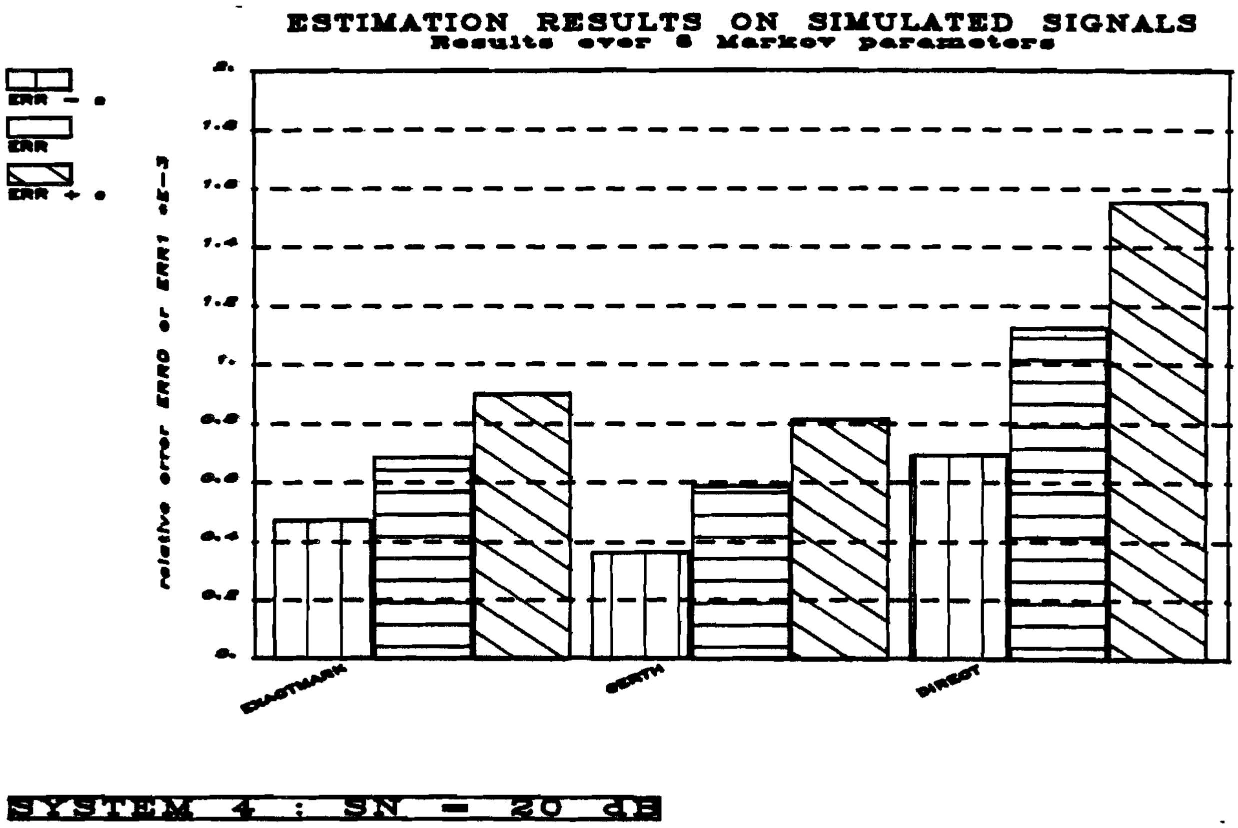


Fig. 5.2 : Estimation results of SYSTEM 4 (8 Markov parameters)

From the results presented in fig. 5.2, we can conclude that the first 8 Markov parameters estimated by EXACTMARK and the GERTH algorithm are simular, while the relative error of the results of the DIRECT method is almost twice the deviation of the estimates of the other methods.

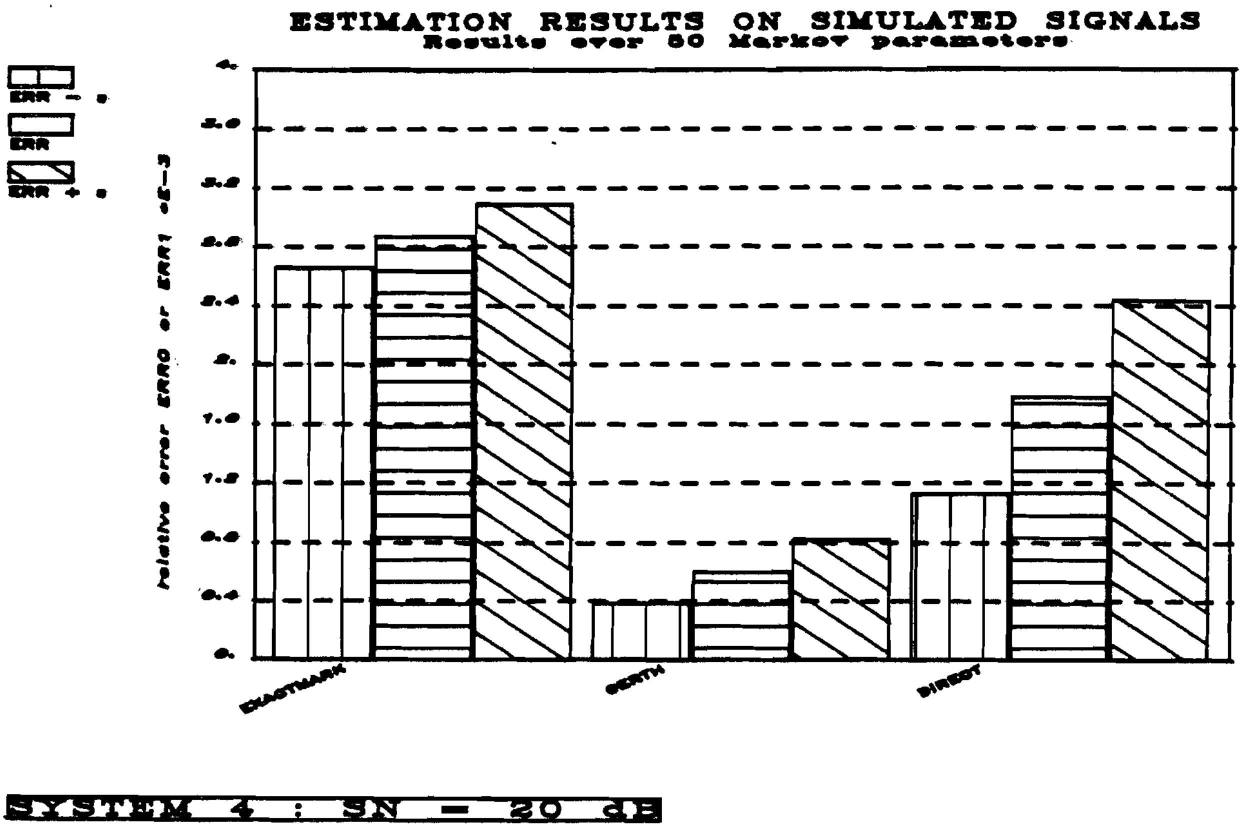


Fig. 5.3 : Estimation results of SYSTEM 4 (50 Markov parameters)

If we take a look at the results on fig. 5.3, we mention that the error of the complete estimated impulse response (--> 50 Markov parameters) of the DIRECT method is almost 40% lower than the error in the estimates of EXACTMARK.

The GERTH method will result in estimates that have a much better performance (60%) than the results of the DIRECT estimation.

5.2 Results of the tests on signals with SN = 40 dB.

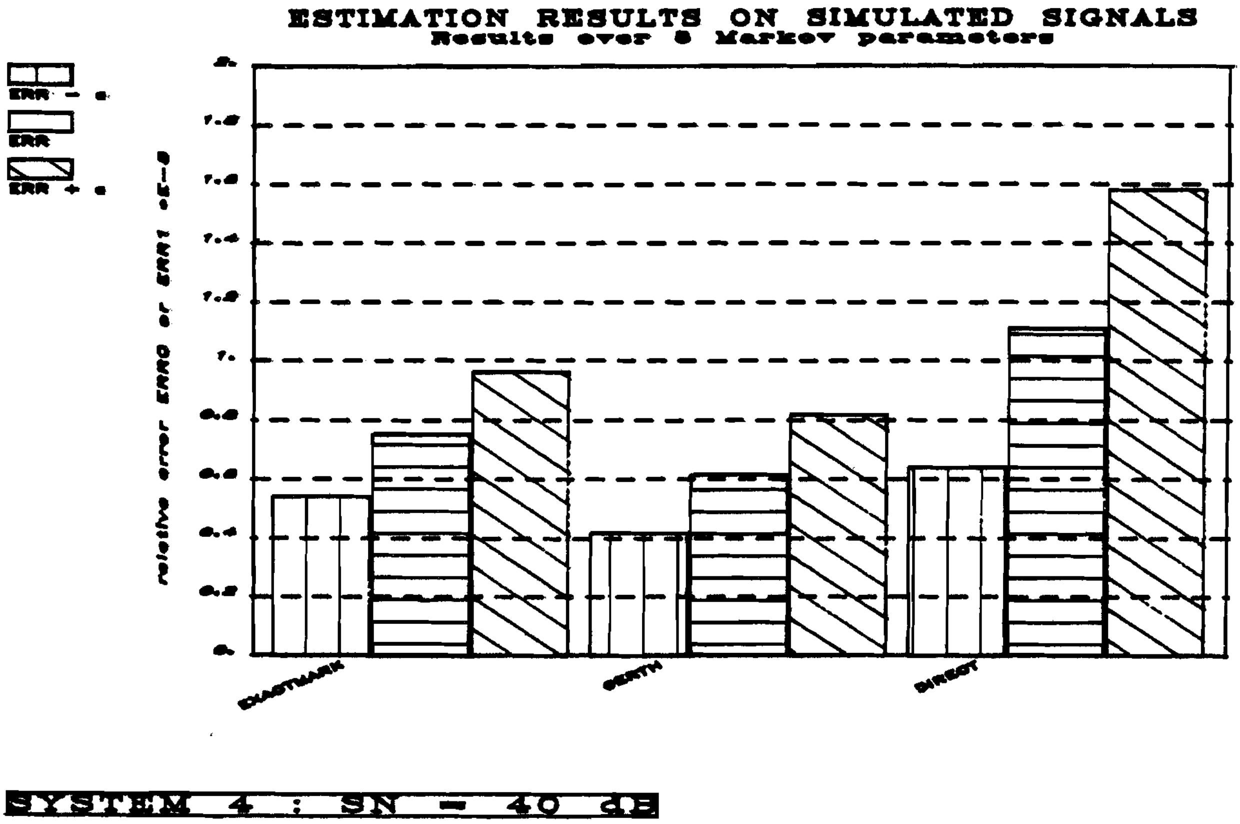


Fig. 5.4 : Estimation results of SYSTEM 4 (8 Markov parameters)

In fig. 5.4 we have presented the results of the different methods by a signal-to-noise ratio of 40 dB. All error values are of the order 10^{-5} while we have found values of about 10^{-3} during the tests on data with SN = 20dB, but the differences between the methods are the same.

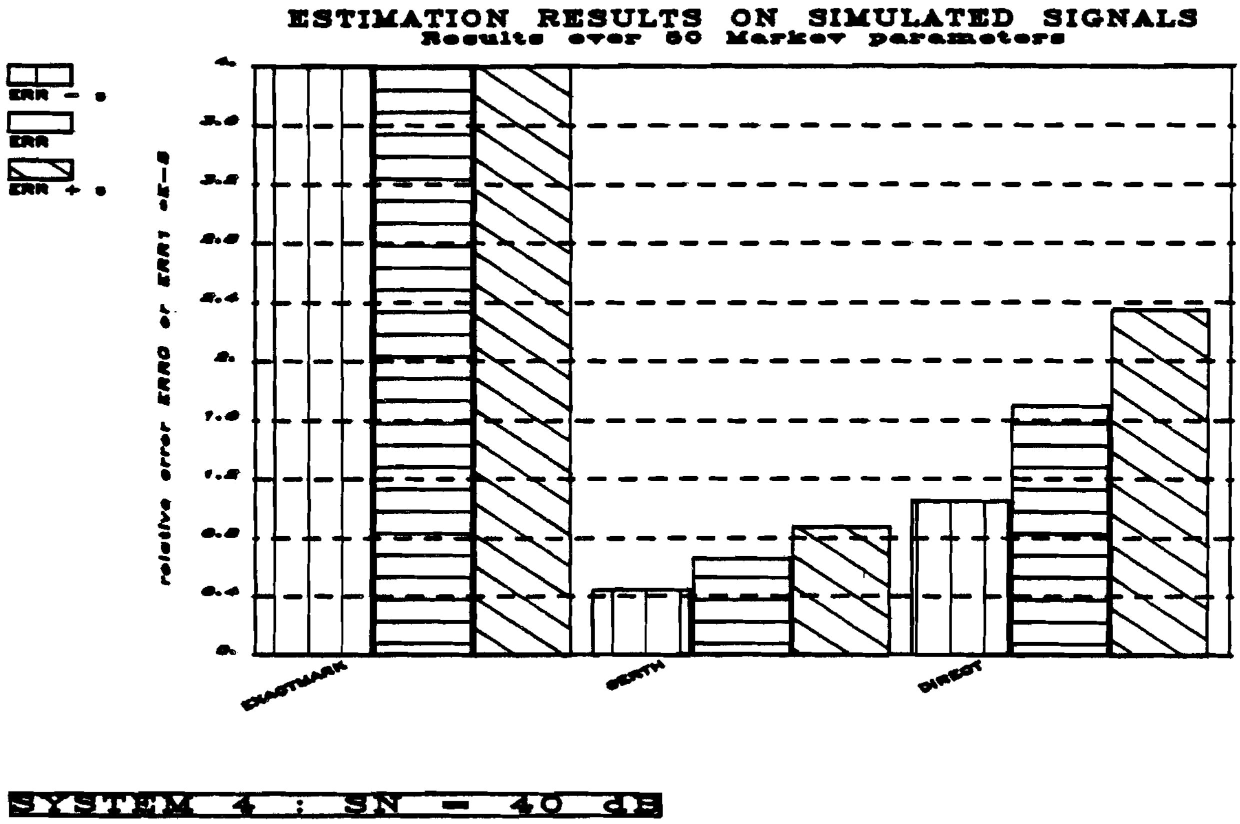
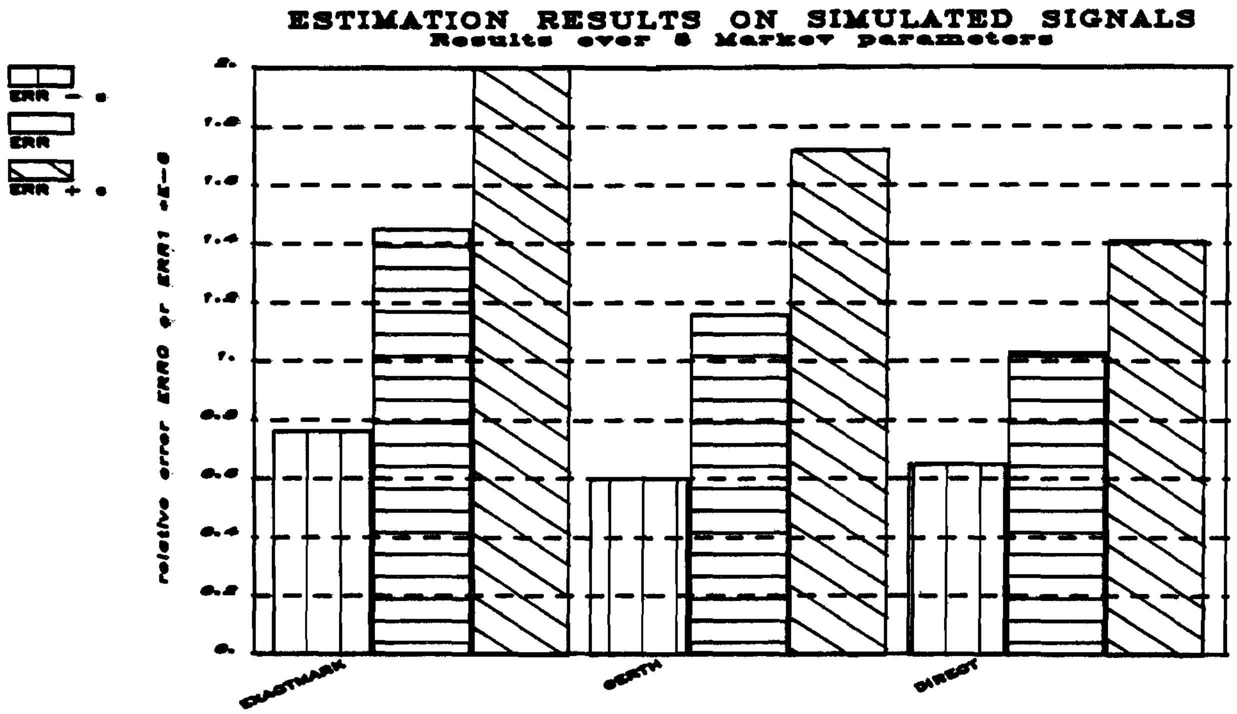


Fig. 5.5 : Estimation results of SYSTEM 4 (50 Markov parameters)

The bars belonging to the results of EXACTMARK exceed the upperbound of the plot: $ERRO_{40} \approx 2.2\ 10^{-3}$. This value is large due to the truncation of the impulse response after 8 samples.
The GERTH algorithm estimates the Markov parameters much better than the DIRECT method.

5.3 Results of the tests on signals with SN = 60 dB.



SYSTEM 4 : SN = 60 dB

Fig. 5.6 : Estimation results of SYSTEM 4 (8 Markov parameters)

The information presented in fig. 5.6 shows that the application of the GERTH algorithm improves the relative error over the start samples of the impulse response (an improvement of about 20%).
The fit of the estimates of the DIRECT method even improves the results of EXACTMARK even with a factor of 30%!

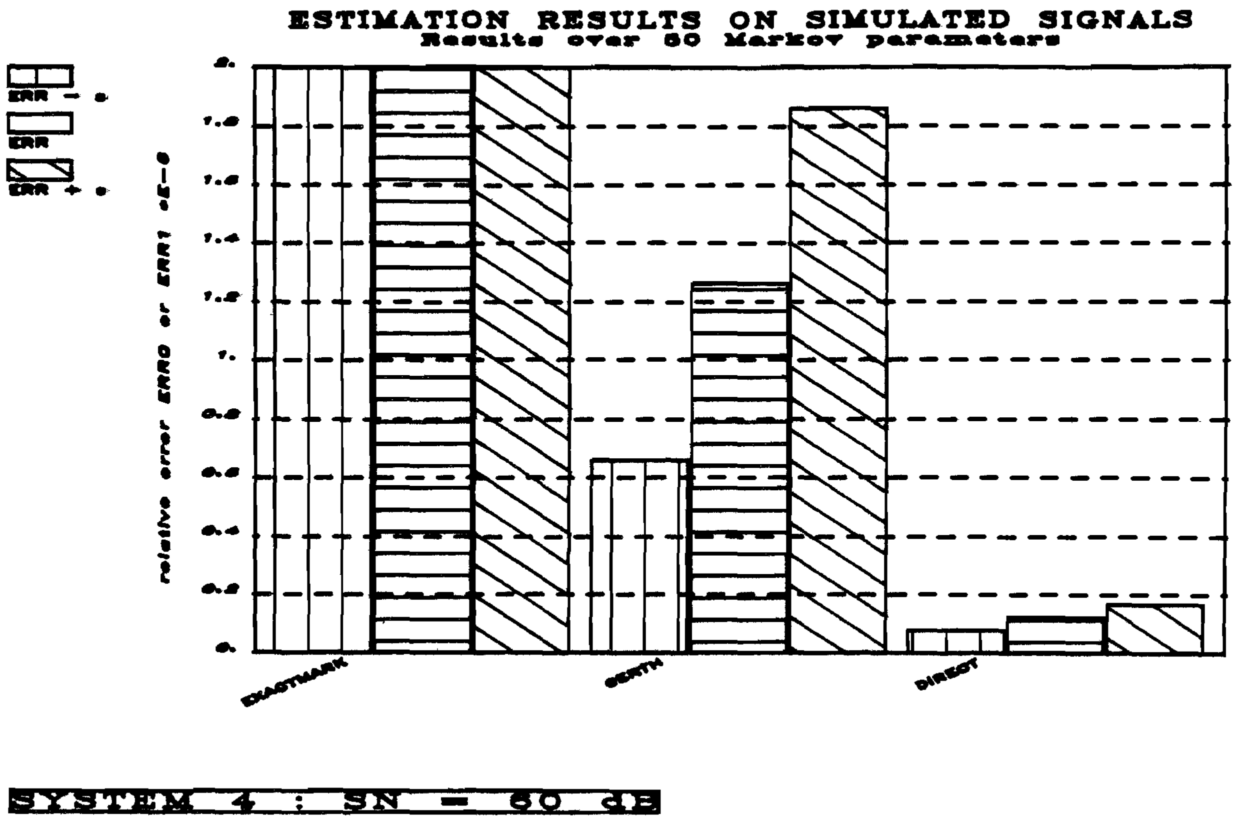


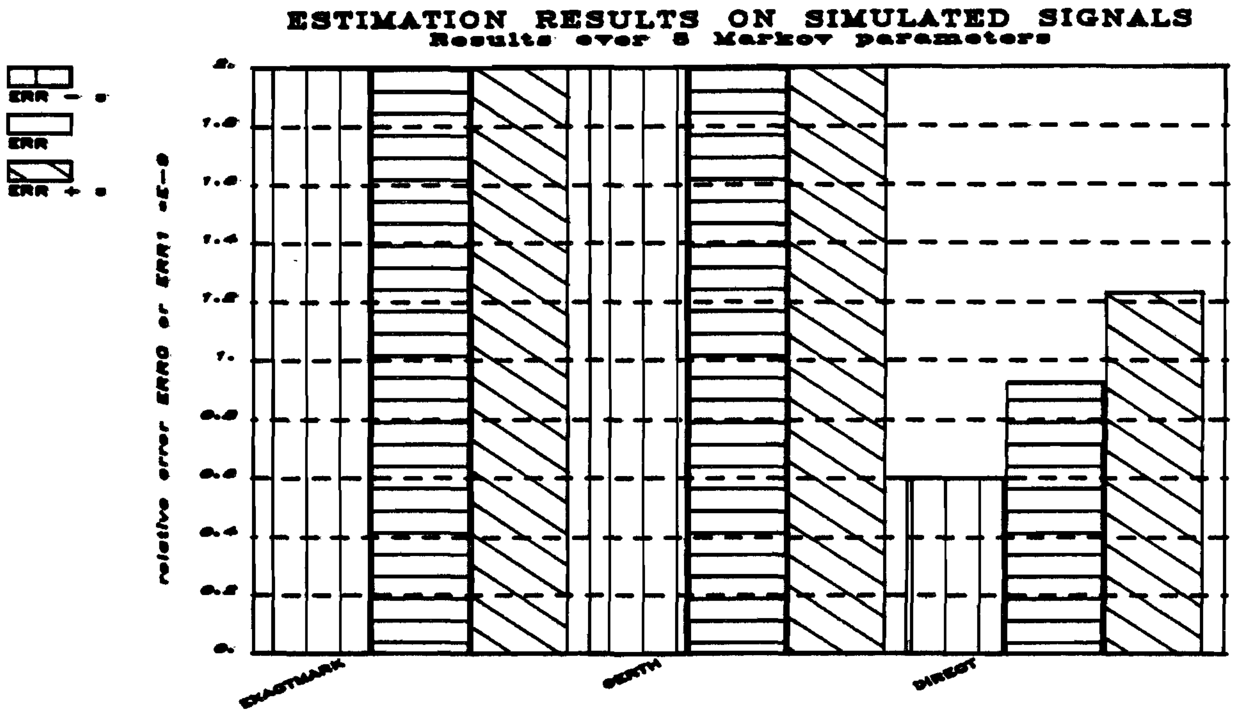
Fig. 5.7 : Estimation results of SYSTEM 4 (50 Markov parameters)

If we take a look at fig. 5.7, we see that the accurcy of the "whole" impulse response (--> 50 Markov parameters) of the DIRECT estimation is almost 90% better then the fit of the expanded sequence of Markov parameters on the deterministic sequence using the minimal polynomial coefficients and start Markov parameters of the GERTH method.

In the following paragraphs the results will be presented of the estimations based on data sets with SN's of 80 and 100 dB.

The quality of the first 8 Markov parameters estimated by EXACTMARK remains the same as during the tests with SN = 60dB (the results presented in fig. 5.6 and fig. 5.7). The GERTH algorithm only uses the output data of EXACTMARK, so the results of the GERTH method will not change either.

5.4 Results of the tests on signals with SN = 80 dB.



SYSTEM 4 : SN = 80 dB

Fig. 5.8 : Estimation results of SYSTEM 4 (8 Markov parameters)

The mean value of the error of the results of the DIRECT estimation is $\approx 0.9\ 10^{-9}$ while the corresponding error value of the GERTH algorithm is $\approx 1.1\ 10^{-6}$!

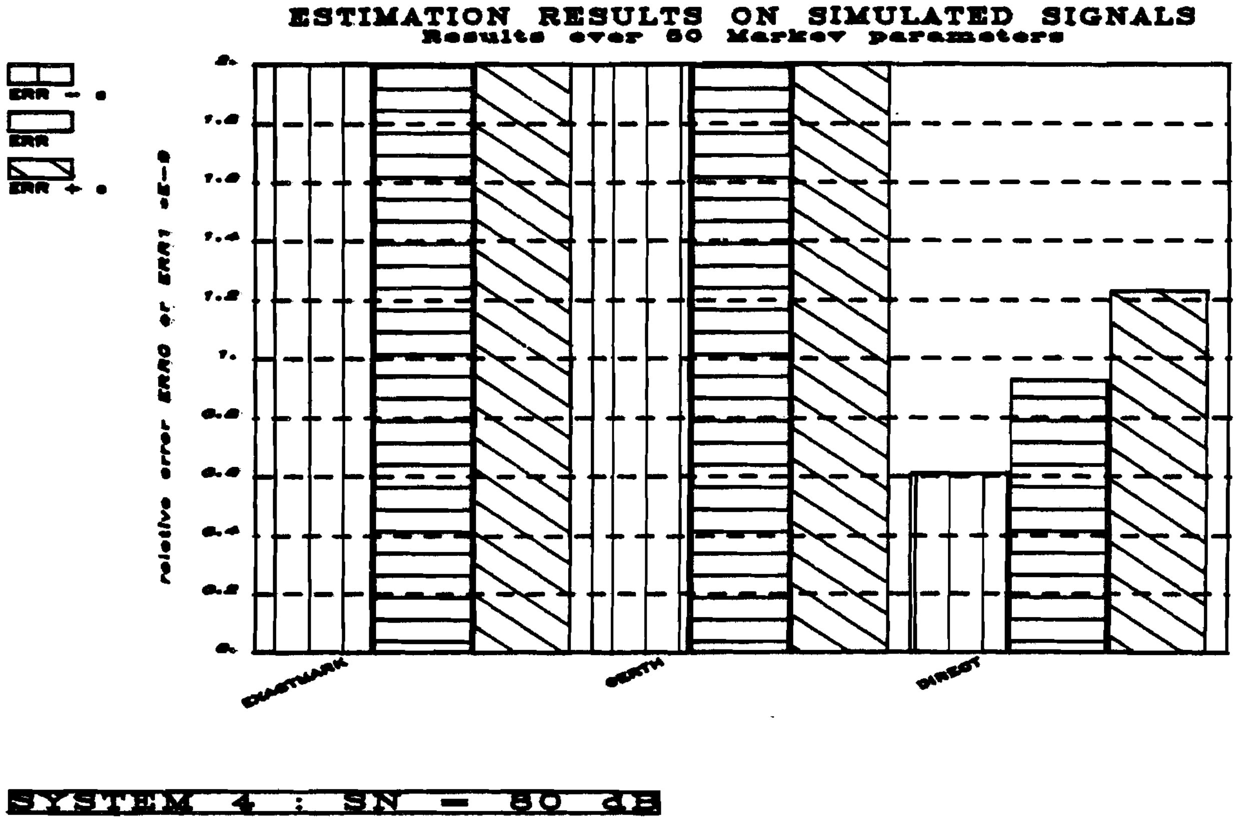


Fig. 5.9 : Estimation results of SYSTEM 4 (50 Markov parameters)

In the preceding figure we see that the ratio of the error (over 50 Markov parameters) of the GERTH method and the error value of the DIRECT estimation is almost 10^{+3}!. This ratio also has been found from the results in fig. 5.8.

5.5 Results of the tests on signals with SN = 100 dB.

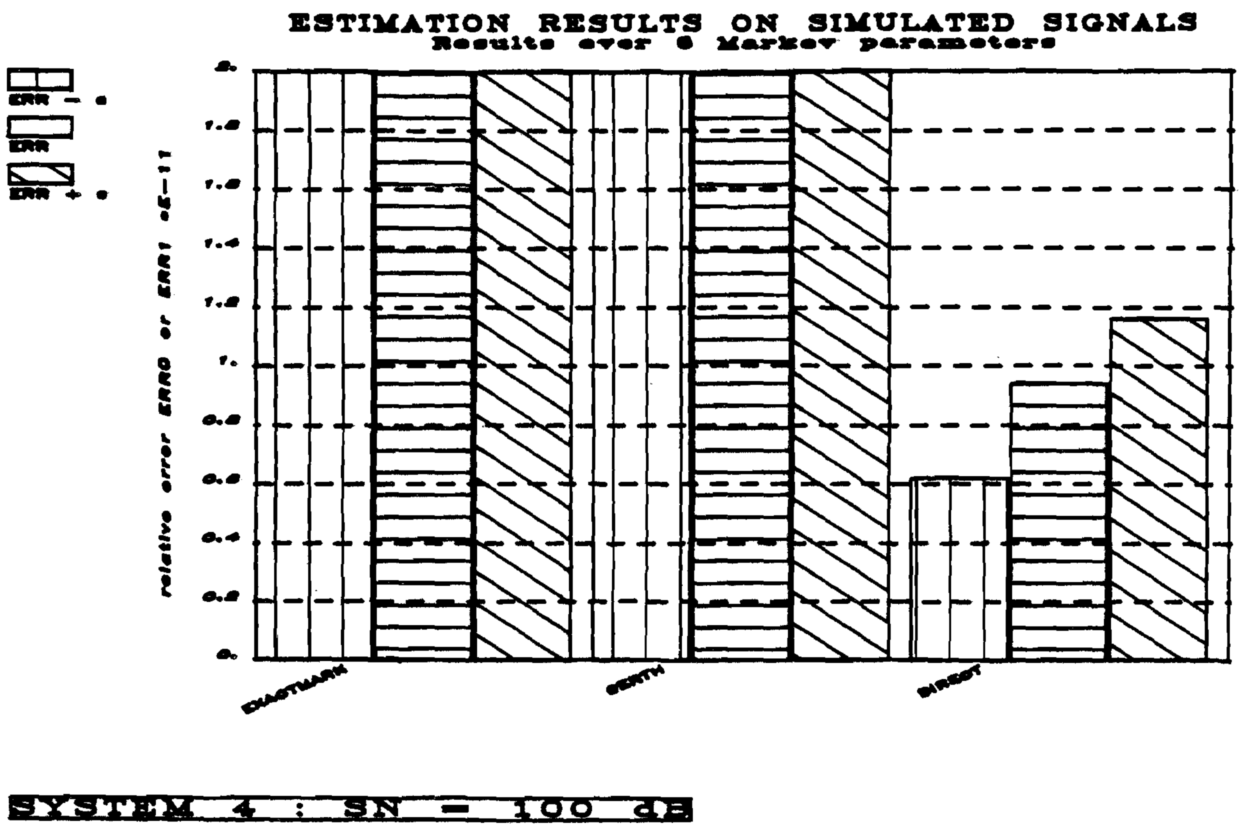


Fig. 5.10 : Estimation results of SYSTEM 4 (8 Markov parameters)

As mentioned before the relative errors of the results of the GERTH method and EXACTMARK are the same as during the tests with signal-to-noise ratios of 60 and 80 dB.

The average value of the error of the DIRECT estimation decreases to $\approx 0.9\ 10^{-11}$ by taking 8 Markov parameters into consideration and $\approx 1.0\ 10^{-11}$ if we look at the "whole" impulse response (see fig. 5.11).

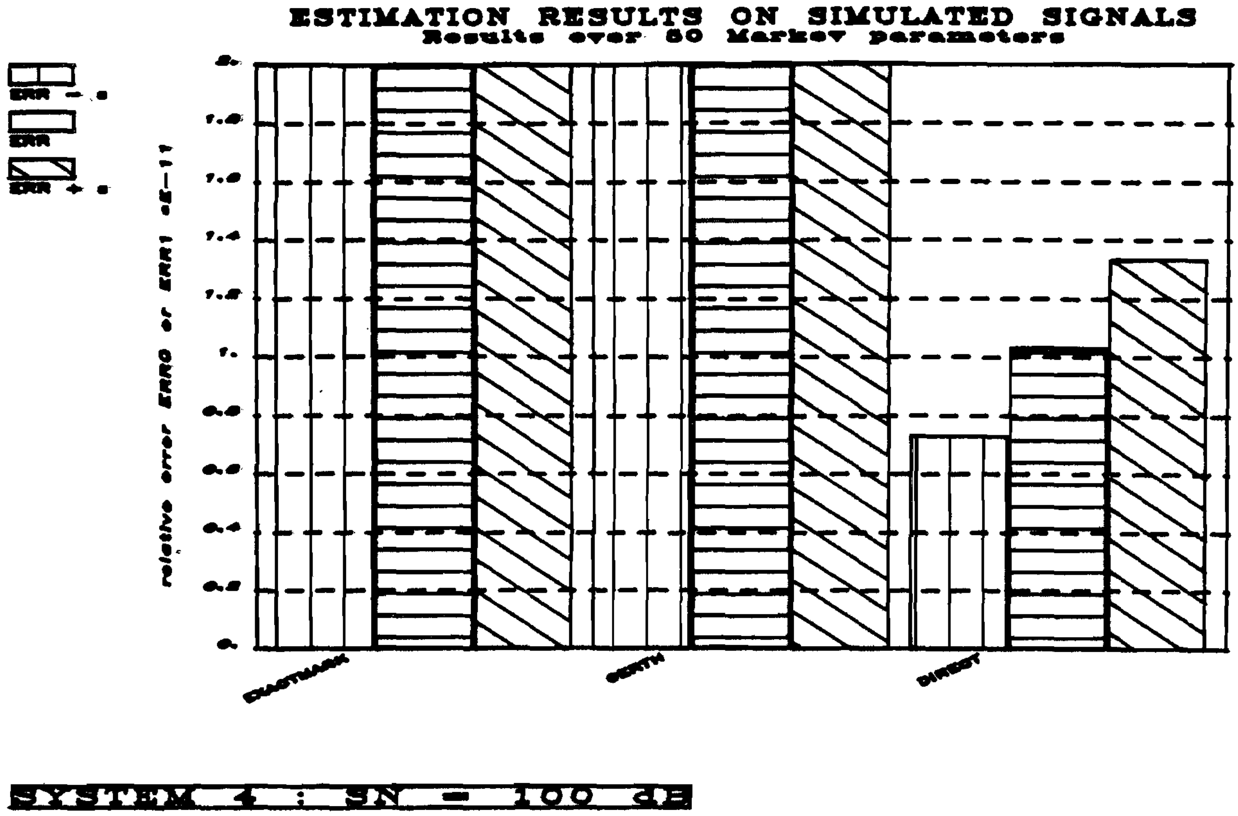


Fig. 5.11 : Estimation results of SYSTEM 4 (50 Markov parameters)

5.6 conclusions

As we mentioned before the DIRECT method needs a lot of computational effort, especially if many input samples of the past should be taken into consideration (so if the impulse response is decreasing slowly). The loss-function is not a very pleasant one: the trace decreases little by little and can have several "flat" areas that could be seen as local optima. So we will have to use a robust minimization method that passes these flat areas!

We mention that the loss-function in the minimization process is the squared distance between the measured output signals (noise corrupted data) and the reconstructed outputs (by using the estimated model). But during the comparison with other methods we have used the distance of the estimated impulse respons to the deterministic impulse response of the system.

The minimal value of the trace will be the total energy in the additive output noise. The DIRECT method is able to estimate the model in such a way that this minimal trace will be found during all test on SYSTEM 4.
The number of Markov parameters estimated with EXACTMARK has been chosen the sam as during our tests on the GERTH algorithm. By applying the derivation of appendix D we can calculate how many Markov parameters should be estimated to make the truncation error less than the influence of the additive output noise.
The number of parameters in EXACTMARK is (8 x 3 x 2 =) 48 while the number of parameters in the DIRECT estimation is the number of minimal polynomial coefficients r plus (r+1) times the entries of Markov parameters (we also will estimate M(0) in the DIRECT method). So this last number has been 41 during our tests on SYSTEM 4.

"Summary" of the results of the several tests:
If we deal with bad conditioned signals (a lot of noise), the GERTH algorithm smoothes the impulse response (see chapter 4) and will give good results. The output equation error expressed by the trace of the error matrix still will be improved by the DIRECT estimation but the improvement is rather small.
With increasing signal-to-noise ratio's the results of the DIRECT method will exceed the GERTH estimates rather fast (starting at about 50 dB).
In case of very well conditioned measurements (SN = 100dB) the estimates of the minimal polynomial coefficients and the start sequence of Markov parameters is very good (an improvement of a factor 10^{+5} in comparison with the results of the GERTH algorithm). The reconstructed eigenvalues calculated by using the estimated minimal polynomial coefficients will very well approximate the exact values.

6 CONCLUSIONS.

In this report two methods have been developed for the estimation of minimal polynomial coefficients and a start squence of Markov parameters. The problem during the estimation of these parameters $\{\hat{a}_i,\hat{M}(i)\}_{i=1,2..r}$ is that the extended sequence of Markov parameters, calculated by applying the minimal polynomial, is not linear in the minimal polynomial coefficients.
The iterative GERTH algorithm will lead to an approximate solution while the DIRECT method will find the exact solution using hill climbing techniques.
The properties of both algorithms have been investigated by performing tests with simulations of several systems. From the results of these tests we can summarise the following conclusions:

-The implemented iterative GERTH algorithm minimizes the squared distance of a sequence of Markov parameters $\{\hat{M}(i)\}_{i=1,2..k}$ (calculated by applying the minimal polynomial expression) to the noise corrupted set $\{\tilde{M}(i)\}_{i=1,2..k}$. During the execution of several iterations this distance will be reduced in most of the tests. But only if the estimated (noise corrupted) Markov parameters are close to the deterministic Markov parameters (no truncation effects and well conditioned signals) the distance between the extended sequence and the deterministic set will be minimized by executing more than one iteration.

-If we deal with estimates of $\{\tilde{M}(i)\}_{i=1,2,..k}$ based on data with a rather small signal-to-noise ratio, it is very useful to decrease the estimated degree of the minimal polynomial (in the GERTH algorithm as well as in DIRECT method). So we will "reduce" the image of the input/output projection. We have used only short data sets and so parts of process dynamics will get lost by noise corruption.

-The DIRECT method needs a lot of computational effort, especially if many Markov parameters have to be taken into consideration. In those cases the calculation of the output signals will be based on many input samples in the past.
The calculation of the partial derivatives of the loss function will use the same number of data samples. The time needed for the calculation of these derivatives is ≈ 0.9 of the total processor time. So if we can estimate the partial derivatives the minimization process

will ask less computational efforts. The properties of an estimation have to be investigated.

-Dealing with signals with large signal-to-noise ratios ($\gtrsim$ 50 dB) the DIRECT method will lead to estimates of minimal polynomial coefficients and a start sequence of Markov parameters that are much better than the estimates found with the GERTH algorithm.

-During all tests on SYSTEM 4 the DIRECT estimation minimizes the reconstruction error almost to the theorictical lowest bound, being the energy of the additive output noise. During the tests on data with a high noise corruption, these bounds almost have been reached by applying the GERTH method only . The DIRECT method is able to reduce the trace of the output error matrix to this lowest bound, but this reduction is not substantial.

-It is very useful to take the GERTH estimates as start parameters of the DIRECT method. This way it is possible to reduce the number of iterations during the minimization process.

-In the minimal polynomial description the order of the model will be min(p,q)xr. The extra model space compared to the input (output-) companian form will lead to multiple eigenvalues (see Appendix B). In case of large SN ratios we easily can extract the eigenvalues with substantial influence from the complete set by applying a singular value decomposition of the Hankel matrix, composed out of the extended sequence of Markov parameters. By neglecting the multiple eigenvalues with a relative low energy contents, we can make an exact realization of the system (from the impulse response model to a state space representation).

The DIRECT method calculates the output signals based on ´m-1´preceding and one actual value of the input samples. In our implementation ´m´ has be chosen half the number of samples in the available file with the measurements. So we will use only the second <u>half</u> of the output measurements. EXACTMARK uses the <u>complete</u> set of measurements so the noise reduction during the estimation with this method (and so with the GERTH method) will be better than during the DIRECT estimation. Especially during estimations on data sets with a lot of noise corruption this property will be shown.

Appendix A DERIVATION OF THE EQUATIONS FOR THE DIRECT METHOD.

In this appendix, belonging to the chapter 3 we will derive expressions for the minimization of a loss function with respect to a set of Markov parameters and the series expansion of this set by means of the minimal polynomial. In the chapter 'DIRECT METHOD' we have found a expression for the loss-function. We will recall these equations (eq. A.1-3).

$$Y = S_m^T \cdot M + E \qquad \text{A.1}$$

$$\hat{Y} = S_m^T \cdot \hat{M} \qquad \text{A.2}$$

or:

$$\begin{bmatrix} \underline{y}^T(h) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \underline{y}^T(h+m) \end{bmatrix} = \begin{bmatrix} \underline{u}^T(h) & \cdots & \underline{u}^T(h-m) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \underline{u}^T(h+m) & \cdots & \underline{u}^T(h) \end{bmatrix} \begin{bmatrix} M^T(0) \\ M^T(1) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ M^T(m) \end{bmatrix} + E$$

We define an equation-error:

$$E \triangleq Y - S_m^T \cdot M = Y - \hat{Y} \qquad \text{A.3}$$

Then the loss function will be:

$$V = \text{trace}(E^T \cdot E) = \text{trace}[(Y - \hat{Y})^T \cdot (Y - \hat{Y})] =$$
$$\text{trace}[Y^T.Y] - 2\,\text{trace}[Y^T.S_m^T.M] + \text{trace}[M^T.S_m.S_m^T.M] \qquad \text{A.4}$$

The matrix M will consist of Markov parameters (see eq. A.1). We will use a series expansion by the minimal polynomial for the Markov parameters M(i) ($i>r$: r is the degree of the minimal polynomial), and we will rename these parameters to functions (eq. A.6: we use a recurrent relation), we will find the matrix-expression of eq. A.5.

$$\underbrace{\begin{bmatrix} \underline{y}^T(h) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \underline{y}^T(h+m) \end{bmatrix}}_{Y} = \underbrace{\begin{bmatrix} \underline{u}^T(h) & \cdots & \underline{u}^T(h-m) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \underline{u}^T(h+m) & \cdots & \underline{u}^T(h) \end{bmatrix}}_{S_m^T} \underbrace{\begin{bmatrix} M^T(0) \\ M^T(1) \\ \cdot \\ M^T(r) \\ F^T(1) \\ \cdot \\ F^T(m-r) \end{bmatrix}}_{M} \qquad \text{A.5}$$

where:

$$F^T(1) = \sum_{i=1}^{r} a_i \cdot M^T(r-i+1)$$

$$F^T(j) = \sum_{i=1}^{j-1} a_i \cdot F^T(j-i) + \sum_{i=j}^{r} a_i \cdot M^T(r-i+j) \qquad (\text{for } 2 \leq j \leq r) \qquad \text{A.6}$$

$$F^T(j) = \sum_{i=1}^{r} a_i \cdot F^T(j-i) \qquad (\text{for } j > r)$$

We will derive reccurent-expressions for all three parts of eq. A.4, starting with first taking h=0:

$$Y^T = \begin{bmatrix} y_1(0) & : & \cdots\cdots & : & y_1(m) \\ \cdot & : & \cdots\cdots & : & \cdot \\ \cdot & : & \cdots\cdots & : & \cdot \\ \cdot & : & \cdots\cdots & : & \cdot \\ y_q(0) & : & \cdots\cdots & : & y_q(m) \end{bmatrix} \qquad A.7$$

So it is easy to see that:

$$\text{trace}\ \{\ Y^T.Y\ \} = \sum_{n=1}^{q} \sum_{k=0}^{m} \{\ y_n(k)\}^2 \qquad A.8$$

The second part will now be derived:

$$S_m^T . M = \qquad \text{A.9}$$

$$\begin{bmatrix} u_1(0) & u_2(0) & \cdots & u_p(0) & : & \cdots & : & u_1(-m) & \cdots & u_p(-m) \\ u_1(1) & u_2(1) & \cdots & u_p(1) & : & \cdots & : & u_1(1-m) & \cdots & u_p(1-m) \\ \vdots & \vdots & & \vdots & : & \cdots & : & \vdots & & \vdots \\ u_1(m) & u_2(m) & \cdots & u_p(m) & : & \cdots & : & u_1(0) & \cdots & u_p(0) \end{bmatrix} .$$

$$\begin{bmatrix} M_{11}(0) & \cdots & M_{q1}(0) \\ \vdots & & \vdots \\ M_{1p}(0) & \cdots & M_{qp}(0) \\ \hline \vdots & & \vdots \\ \hline M_{11}(r) & \cdots & M_{q1}(r) \\ \vdots & & \vdots \\ M_{1p}(r) & \cdots & M_{qp}(r) \\ \hline F_{11}(1) & \cdots & F_{q1}(1) \\ \vdots & & \vdots \\ F_{1p}(1) & \cdots & F_{qp}(1) \\ \hline \vdots & & \vdots \\ \hline F_{11}(m-r) & \cdots & F_{q1}(m-r) \\ \vdots & & \vdots \\ F_{1p}(m-r) & \cdots & F_{qp}(m-r) \end{bmatrix} =$$

$$\begin{bmatrix} \sum_{j=0}^{r}\sum_{i=1}^{p} u_i(-j)\cdot M_{1i}(j) + \sum_{j=r+1}^{m}\sum_{i=1}^{p} u_i(-j)\cdot F_{1i}(j-r) & \vdots & \cdots & \vdots & \sum_{j=0}^{r}\sum_{i=1}^{p} u_i(-j)\cdot M_{qi}(j) + \sum_{j=r+1}^{m}\sum_{i=1}^{p} u_i(-j)\cdot F_{qi}(j-r) \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \sum_{j=0}^{r}\sum_{i=1}^{p} u_i(m-j)\cdot M_{1i}(j) + \sum_{j=r+1}^{m}\sum_{i=1}^{p} u_i(m-j)\cdot F_{1i}(j-r) & \vdots & \cdots & \vdots & \sum_{j=0}^{r}\sum_{i=1}^{p} u_i(m-j)\cdot M_{qi}(j) + \sum_{j=r+1}^{m}\sum_{i=1}^{p} u_i(m-j)\cdot F_{qi}(j-r) \end{bmatrix}$$

with:

$$Y^T = \begin{bmatrix} y_1(0) & : & \cdots\cdots & : & y_1(m) \\ \cdot & : & \cdots\cdots & : & \cdot \\ \cdot & : & \cdots\cdots & : & \cdot \\ \cdot & : & \cdots\cdots & : & \cdot \\ y_q(0) & : & \cdots\cdots & : & y_q(m) \end{bmatrix} \qquad \text{A.10}$$

then: $$Y^T . S_m^T . M = \tag{A.11}$$

$$\begin{bmatrix}
\sum_{k=0}^{m} y_1(k)\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{1i}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j).F_{1i}(j-r)\} & \cdots & \sum_{k=0}^{m} y_1(k)\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{qi}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j).F_{qi}(j-r)\} \\
\vdots & & \vdots \\
\sum_{k=0}^{m} y_q(k)\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{1i}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j).F_{1i}(j-r)\} & \cdots & \sum_{k=0}^{m} y_q(k)\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{qi}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(m-j).F_{qi}(j-r)\}
\end{bmatrix}$$

So the trace $[Y^T . S_m^T . M]$ can be expressed by:

A.12

$$\sum_{n=1}^{q} \sum_{k=0}^{m} y_n(k) \{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j) . M_{ni}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j) . F_{ni}(j-r) \}$$

From matrix equation A.8, we calculate the third part of the loss-function:

$$M^T . S_m . S_m^T . M(1,1) = \tag{A.13}$$

$$\sum_{k=0}^{m} \{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j) . M_{1i}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j) . F_{1i}(j-r) \}^2$$

$$M^T . S_m . S_m^T . M(q,q) = \tag{A.14}$$

$$\sum_{k=0}^{m} \{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j) . M_{qi}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j) . F_{qi}(j-r) \}^2$$

$$M^T.S_m.S_m^T.M(s,t)= \qquad A.15$$

$$\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(-j).M_{si}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(-j).\ F_{si}(j-r)\} \ .$$

$$\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(-j).M_{ti}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(-j).\ F_{ti}(j-r)\} + \ . \ . \ .$$

$$\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(m-j).M_{si}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(m-j).\ F_{si}(j-r)\} \ .$$

$$\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(m-j).M_{ti}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(m-j).\ F_{ti}(j-r)\} =$$

$$\sum_{k=0}^{m} \{\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{si}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j).\ F_{si}(j-r)\} \ .$$
$$\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{ti}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j).\ F_{ti}(j-r)\}\}$$

(We can use this last general expression if we have to calculate the determinant instead of the trace). So:

$$\text{trace } \{M^T.S_m.S_m^T.M \} = \qquad A.16$$

$$\sum_{n=1}^{q} \sum_{k=0}^{m} \{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{ni}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j).\ F_{ni}(j-r)\}^2$$

Now we are able to give an expression for the loss-function V:

$$\text{trace}\ \{ E^T . E \} = \qquad \text{A.17}$$

$$\sum_{n=1}^{q} \sum_{k=0}^{m} [\{ y_n(k) \}^2 + \{ -2\, y_n(k) + \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j) . M_{ni}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j) . F_{ni}(j-r) \} .$$

$$\{ \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j) . M_{ni}(j) + \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j) . F_{ni}(j-r) \}]$$

The partial derivatives of the loss function with respect to the Markov parameters will be:

$$\frac{d(\text{trace}\ \{ E^T . E \})}{d(M_{ni'}(j'))} = \qquad \text{A.18}$$

$$\sum_{k=0}^{m} [\{ -2\, y_n(k) + 2 \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j) . M_{ni}(j) + 2 \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j) . F_{ni}(j-r) \} .$$

$$\{ u_{i'}(k-j') + \sum_{j=r+1}^{m} u_{i'}(k-j) . \frac{d(F_{ni'}(j-r))}{d(M_{ni'}(j'))} \}]$$

M(0) has no contribution to the minimal polynomial expansion, so :

$$\frac{d(M_{ni'}(j'\neq 0))}{d(M_{ni'}(0))} = \frac{d(F_{ni'}(j'))}{d(M_{ni'}(0))} = 0$$

Applying this in eq. A.18 than:

If $j'=0$ then:

$$\frac{d(\mathrm{trace}\ \{ E^T.E \})}{d(M_{ni'}(j'))} = \tag{A.19}$$

$$\sum_{k=0}^{m} [\ \{-2\ y_n(k) + 2 \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{ni}(j) + 2 \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j).\ F_{ni}(j-r)\} \ .\ \{ u_{i'}(k) \}]$$

The partial derivatives of the loss function with respect to the minimal polynomial coefficients will be:

$$\frac{d(\mathrm{trace}\ \{ E^T.E \})}{d(a_{j'})} = \tag{A.20}$$

$$\sum_{n=1}^{q} \sum_{k=0}^{m} [\ \{-2\ y_n(k) + 2 \sum_{j=0}^{r} \sum_{i=1}^{p} u_i(k-j).M_{ni}(j) + 2 \sum_{j=r+1}^{m} \sum_{i=1}^{p} u_i(k-j).\ F_{ni}(j-r)\} \ .$$

$$\{ \sum_{i=1}^{p} \sum_{j=r+1}^{m} u_i(k-j). \frac{d(\ F_{ni}(j-r))}{d(\ a_{j'}\ )} \}]$$

For the calculation of A.18, A.19 and A.20, we need expressions for:

$$\frac{d(F_{ni'}(j-r))}{d(M_{ni'}(j'))} \quad \text{and} \quad \frac{d(F_{ni'}(j-r))}{d(a_{j'})}$$

Note that:

$$\frac{d(M_{ni'}(j-r))}{d(a_{j'})} = 0$$

We are going to use the expressions of eq. A.6 for the calculation of the partial derivatives. We can derive that:

A.21

$$\frac{d(F_{ni'}(j))}{d(M_{ni'}(j'))} =$$

1. If $j = 1$ then:

$$\longrightarrow a_{r+1-j'}$$

2. If $2 \leq j \leq r$ and $j \leq j' \leq r$ then:

$$\longrightarrow \sum_{t=1}^{j-1} a_t \cdot \frac{d(F_{ni'}(j-t))}{d(M_{ni'}(j'))} + a_{r+j-j'}$$

3. If $2 \leq j \leq r$ and $1 \leq j' < j$ then:

$$\longrightarrow \sum_{t=1}^{j-1} a_t \cdot \frac{d(F_{ni'}(j-t))}{d(M_{ni'}(j'))}$$

4. If $j > r$ then:

$$\longrightarrow \sum_{t=1}^{r} a_t \cdot \frac{d(F_{ni'}(j-t))}{d(M_{ni'}(j'))}$$

And for the partial derivatives to minimal polynomial coefficients:

A.22

$$\frac{d(F_{ni'}(j))}{d(a_{j'})} =$$

1. If $j = 1$ then:

$$\longrightarrow M_{ni'}(r+1-j')$$

2. If $2 \leq j \leq r$ and $j \leq j' \leq r$ then:

$$\longrightarrow \sum_{t=1}^{j-1} a_t \cdot \frac{d(F_{ni'}(j-t))}{d(a_{j'})} + M_{ni'}(r+j-j')$$

3. If $2 \leq j \leq r$ and $1 \leq j' \leq j-1$ then:

$$\longrightarrow \sum_{t=1}^{j-1} a_t \cdot \frac{d(F_{ni'}(j-t))}{d(a_{j'})} + F_{ni'}(j-j')$$

4. If $j > r$ then:

$$\longrightarrow \sum_{t=1}^{r} a_t \cdot \frac{d(F_{ni'}(j-t))}{d(a_{j'})} + F_{ni'}(j-j')$$

Now we are able to compute equations A.18 and A.19 by using the expressions of equations A.21. By substituting A.22 in equation A.20, it is possible to calculate the partial derivatives of the loss-function with respect to the minimal polynomial coefficients.

Appendix B THE DIMENSION VERSUS THE DEGREE OF THE MINIMAL POLYNOMIAL

We will look for the dimension of a MIMO system, identified using a model consisting of coefficients of the minimal polynomial and a start sequence of Markov parameters, $\{a_i, M(i)\}_{i=1,2,..r}$.
The number of parameters of this model N_m is given by expression B.1 (M(0) is not included):

$$N_m = r\ (\ p \times q + 1\ ) \qquad \text{B.1}$$

The dimension of this model is not known in advance: the description allows models with minimal dimension n=r, r+1, ...
Can we find an upperbound for the minimal dimension of the model?
If we will extend the Hankel matrix H_r (definition eq. 2.1), with Markov parameters calculated by applying the minimal polynomial relation, to Hankel matrices H_{r+j} (j>0), it is easy to see that:

$$\text{rank}\{\ H_{r+j}\ \} = \text{rank}\{\ H_r\ \} \qquad (j>0) \qquad \text{B.2}$$

because the extension will depend linearly on the original part H_r of H_{r+j}.
So we have found an upperbound of the minimal dimension of a model $\{a_i, M(i)\}_{i=1,2,..r}$, given in equation B.3 [7: theorem 2].

$$n_{max} = \max(\text{rank}\{\ H_r\ \}) = r \times \min(p,q) \qquad \text{B.3}$$

where r x min(p,q) is the minimum of the number of columns and rows of the Hankel matrix H_r.

A parametrization $\{a_i, M(i)\}_{i=1,2..r}$ includes models with minimal dimension n = r, r+1, .. r x min(p,q).
During an identification based on the minimal polynomial model, the parameters will be estimated independently, without any restrictions. This means that the identified system will have the highest rank possible due to errors during the calculations and/or due to noise corruption of the measurements. The dimension of the model during the identification will therefore always be the upperbound of eq. B.3!

The number of independent parameters in a canonical state space description N_c will be [5]:

$$N_c = n (p + q) \qquad B.4$$

with n = r, r+1, .. r x min(p,q) we will find that the number of parameters will be:

$$\begin{aligned} &1.\ \text{if } p \geq q \text{ then } N_c|_{max} = (r \times \min(p,q) \times (p + q) \\ &\qquad = r \times q \times (p + q) \\ &\qquad = r \times p \times q + r \times q^2 \\ &\qquad\qquad\qquad\qquad\qquad\qquad B.5 \\ &2.\ \text{if } q > p \text{ then } N_c|_{max} = (r \times \min(p,q) \times (p + q) \\ &\qquad = r \times p \times (p + q) \\ &\qquad = r \times p \times q + r \times p^2 \end{aligned}$$

So in both situations the number of independent parameters of the minimal polynomial model N_m is less than or equal to the number of parameters in the canonical state space representation N_c. This implies that the model $\{a_i, M(i)\}_{i=1,2..r}$ can not represent all possible systems of order r x min(p,q): the minimal polynomial has to be of degree r.
What will be the restrictions on these systems?
For n>r, the model will have multiple poles ('distinct' poles). We will prove this for n = r x min(p,q).
This last situation will generally occur during an identification. So the Hankel matrix H_r will be of full rank; we can look at two cases (as done in eq. B.5)(c.f. [2] and [7 : p.p. 36 - 45]):

1. if $p \geq q$ ==> rank{ H_r } = r x q : we will find partial Kronecker row indices n_r

$$n_{ri} = r \text{ for all } i=1,2 .. q$$

B.6

2. if $q > p$ ==> rank{ H_r } = r x p : we will find partial Kronecker column indices n_c

$$n_{ci} = r \text{ for all } i=1,2 .. p$$

In the second situation, there have to be static dependencies between the outputs, so the question is why the model is chosen that way. We will only take a look at situation 1.

In the situation that the number of inputs of the system is greater than or equal to the number of outputs ($p \geq q$), we can transform the model in the canonical observable form with a system matrix A of a known structure.

The first block part (with 'r' rows) of this matrix A is given in eq. B.7.

$$
A_1 = \left[\begin{array}{cccccc:ccccc:}
0 & 1 & 0 & \cdot & \cdot & 0 & 0 & \cdot & \cdot & \cdot & 0 \\
0 & 0 & 1 & 0 & \cdot & 0 & \cdot & & & & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & & & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & & & & \cdot \\
0 & 0 & 0 & \cdot\cdot & \cdot & 1 & 0 & & & & 0 \\
x_{11r} & & \cdot & \cdot & x_{112} & x_{111} & x_{12r} & \cdot & \cdot & x_{122} & x_{121}
\end{array} \right.
$$

$$
\left. \begin{array}{:cccc:cccc:cccc}
0 & \cdot & \cdot & 0 & \cdot & \cdot & \cdot & \cdot & 0 & \cdot & \cdot & 0 \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
0 & \cdot & \cdot & 0 & \cdot & \cdot & \cdot & \cdot & 0 & \cdot & \cdot & 0 \\
x_{13r} & \cdot & \cdot & x_{131} & \cdot & \cdot & \cdot & \cdot & x_{1qr} & \cdot & \cdot & x_{1q1}
\end{array} \right] \qquad B.7
$$

This block part structure can be derived from the Hankel matrix H_{r+1} (see [2]). The x parameters (eq. B.7)indicate the linear dependence between the first row of the last block part (with 'r' rows) of H_{r+1} and the preceding rows of H_r.

The dependence of the first row of the last part with the rows excisting of corresponding entries of Markov parameters is expressed by the parameters x_{11r}, .. x_{112}, x_{111}; the dependences with the remaining entries are shown by x_{12r},.. x_{121}, . . . , x_{1qr},.. x_{1q1}.

The degree of the minimal polynomial is 'r', so the row 'q x r + 1' will depend only on the corresponding entries of the Markov parameters $M(i)_{i=1,2..r}$ in H_r. This means that:

$$x_{1js} = 0 \quad \text{for all } j=2,3..\, q \ \underline{\text{and}}\ s=1,2..\, r \tag{B.8}$$

A same derivation can also be made for the entries 2,3 .. q. So:

$$x_{ijs} = 0 \quad \text{for all } \{i=1,2..\, q,\ j=1,2..\, q\}_{i \neq j} \ \underline{\text{and}} \ s=1,2..\, r \tag{B.9}$$

These structural zero´s, q x [(q-1) x r] = (n-r) x q in number, are restrictions and so the number of degrees of freedom will be:

$$N_c = n \times (p+q) - q \times (n-r) = n \times p + r \times q \tag{B.10}$$

Using the parameter values of equation B.8, we will find a system matrix A in canonical observable state space notation of the following form:

$$A = \begin{bmatrix} A_1 & 0 & 0 & . & 0 \\ 0 & A_2 & 0 & . & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ 0 & 0 & . & 0 & A_{min(p,q)} \end{bmatrix} \tag{B.11}$$

$$\text{with } A_i = \begin{bmatrix} 0 & 1 & 0 & . & . & 0 \\ 0 & 0 & 1 & 0 & . & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & . & 1 \\ x_{iir} & . & . & . & x_{ii2} & x_{ii1} \end{bmatrix}$$

The matrix A has to have a minimal polynomial of degree ´r´, so:

$$A^r - a_1A^{r-1} - a_2A^{r-2} . \,.- a_rI_{r \times min(p,q)} = NULL_{r \times min(p,q)} \tag{B.12}$$

where: $I_{r \times \min(p,q)}$ is a identity matrix with dimension $[\, r \times \min(p,q) \, , \, r \times \min(p,q)]$

$NULL_{r \times \min(p,q)}$ is a square zero matrix with dimension $[\, r \times \min(p,q) \, , \, r \times \min(p,q)]$

If we calculate powers of a block diagonal matrix, the results will also be of a block diagonal structure. This implies that the minimal polynomial of equation B.10 can also be applied to A_i instead of A! So each part A_i will have the same parameters $x_{ii1}, x_{ii2}, \ldots x_{iir}$. see [2] and these parameters will be the coefficients of the minimal polynomial:

$$x_{iij} = a_j \text{ for all } i=1,2.. \min(p,q) \text{ \underline{and} for all } j=1,2.. r \qquad \text{B.13}$$

The block matrices on the diagonal of A will have to be the same and so we have proven that the roots of the minimal polynomial will form a set of q multiple ('distinct') eigenvalues of system matrix A.
Another consequence of eq. B.12 is that there are not left 'q x r' degrees of freedom in the canonical observable system matrix A but only 'r'. So the last expression of eq. B.10 is 'r' instead of 'q x r'!
The remaining number of parameters will be:

$$N_c = n \times p + r = r \times (p \times q + 1) = N_m \ ! \qquad \text{B.14}$$

An equivalent derivation can be made if q>p; in that case we should use a canonical controlable state space notation.
We will illustrate the previous theoretical part by an example (worked out by P. Van den Hof in a summary of a MIMO meeting).

B.1 Example to illustrate Appendix B.

Given 2 Markov parameters of a '2'input - '2'output system:

$$M(1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad M(2) = \begin{bmatrix} 2 & 3 \\ 1 & 4 \end{bmatrix} \qquad \text{B.15}$$

and the degree of the minimal polynomial is 2 (r=2).
A model $\{a_i, M(i)\}_{i=1,2}$ with random M(i) and a_i can have a minimal

dimension n=2, 3 or 4 (according to the previous theory).
The information about the dimension of the model is contained in $\{a_i, M(i)\}_{i=1,2}$; the Hankel matrix can be filled in an unique way using the available parameters. The rank of the Hankel matrix expresses the dimension of the model.
We will take a look at several possibilities:

a. A second order representation.
b. A third order representation.
c. A fourth order representation.

B.1.1 A second order representation.

The number of parameters in the minimal polynomial model is $N_m = r(p \times q + 1) = 10$. A second order model with Markov parameters as presented in equation B.15 will be represented in a unique way by $\{M_{ij}(k),\ k \leq n_{ri} + n_{cj}\}_{i=1,2\ j=1,2}$ with $n_{r1} = n_{r2} = n_{c1} = n_{c2} = 1$ [2].
So we have a unique representation $\{M_{ij}(k),\ k \leq 2\}_{i=1,2\ j=1,2}$. This means that M(1) <u>and</u> M(2) are necessary and sufficient to represent a second order model. The a_i parameters will depend on the Markov parameters!
The number of independent parameters in the minimal polynomial model will be $N_m|_{indep} = N_m - 2 = 8$; the same number of parameters as needed in the canonical state space description ($N_c = n(p+q) = 8$).
We will calculate the minimal polynomial coefficients from the partial behaviour matrix [3], (--> ´incomplete Hankel matrix´ in [2]):

$$B_2 = \begin{bmatrix} 1 & 0 & : & 2 & 3 \\ 0 & 1 & : & 1 & 4 \\ - & - & - & - & - \\ 2 & 3 & : & ? & ? \\ 1 & 4 & : & ? & ? \end{bmatrix} \qquad \text{B.16}$$

The rank of the Hankel matrix H_2 has to be 2, so by applying the Main lemma [3] we can find an unique solution for M(3):

$$M(3) = \begin{bmatrix} 7 & 18 \\ 6 & 19 \end{bmatrix} \qquad B.17$$

Now the a_i parameters can be calculated by means of the minimal polynomial. We will find the unique values $a_1 = 6$ and $a_2 = -5$. In cases where the minimal polynomial coefficients will be known, we will find two constraints on the entries of M(1) and/or M(2) to guarantee a dimension 2 of the model.

B.1.2 A third order representation.

In the given situation and a third order model, the Kronecker row indices will be:

$$\begin{matrix} n_{r1} = 2 \\ n_{r2} = 1 \end{matrix} \qquad \text{or} \qquad \begin{matrix} n_{r1} = 1 \\ n_{r2} = 2 \end{matrix} \qquad B.18$$

We will take a look at the situation with $n_{r1}=2$ and $n_{r2}=1$.
With the Markov parameters M(1) and M(2) (eq. B.15) and minimal polynomial coefficients a_1 and a_2 we can make a minimal realization (in a canonical observable form!):

$$A = \left[\begin{array}{cc:c} 0 & 1 & 0 \\ x_{112} & x_{111} & x_{121} \\ \hdashline x_{212} & x_{211} & x_{221} \end{array}\right] \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad B.19$$

We mention that there are still 12 parameters in this realization (while in the minimal polynomial model there are only 10 degrees of freedom!).
We will construct the partial behaviour matrix (eq. B.16) and we will determine the x parameters.

$$B_2 = \left[\begin{array}{cc:cc:cc} 1 & 0 & 2 & 3 & M_{11}(3) & M_{12}(3) \\ 0 & 1 & 1 & 4 & M_{21}(3) & M_{22}(3) \\ \hdashline 2 & 3 & M_{11}(3) & M_{12}(3) & & \\ 1 & 4 & M_{21}(3) & M_{22}(3) & & \\ \hdashline M_{11}(3) & M_{12}(3) & & & & \end{array}\right] \qquad \text{B.20}$$

In [2] and [7 : par. 2.2] we have found that that the $x_{1..}$ parameters express the linear dependence:

$$\begin{bmatrix} M_{11}(3) \\ M_{12}(3) \end{bmatrix} = x_{112} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + x_{111} \begin{bmatrix} 2 \\ 3 \end{bmatrix} + x_{121} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad \text{B.21}$$

But we have also stated that the degree of the minimal polynomial is 2! So:

$$\begin{bmatrix} M_{11}(3) \\ M_{12}(3) \end{bmatrix} = a_2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + a_1 \begin{bmatrix} 2 \\ 3 \end{bmatrix} \qquad \text{B.22}$$

The restrictions B.21 and B.22 indicate that the parameter x_{121} is a <u>structural zero</u> and x_{112} and x_{111} are uniquely determined by resp.a_2 and a_1.

Now we are looking for the remaining x parameters. We will start with a partial behaviour matrix B_3:

$$B_3 = \left[\begin{array}{cc:cc:cc} 1 & 0 & 2 & 3 & 2a_1+a_2 & 3a_1 \\ 0 & 1 & 1 & 4 & a_1 & 4a_1+a_2 \\ \hdashline 2 & 3 & 2a_1+a_2 & 3a_1 & & \\ 1 & 4 & a_1 & 4a_1+a_2 & & \\ \hdashline 2a_1+a_2 & 3a_1 & & & & \\ a_1 & 4a_1+a_2 & & & & \end{array}\right] \qquad \text{B.23}$$

And with the definition of the x parameters:

$$\begin{bmatrix} 1 \\ 4 \\ a_1 \\ 4a_1+a_2 \end{bmatrix} = x_{212} \begin{bmatrix} 1 \\ 0 \\ 2 \\ 3 \end{bmatrix} + x_{211} \begin{bmatrix} 2 \\ 3 \\ 2a_1+a_2 \\ 3a_1 \end{bmatrix} + x_{221} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 4 \end{bmatrix} \qquad \text{B.24}$$

So we have found 4 equations with 3 unknowns. We will have to show that there is one dependent relation so we will have to prove that:

$$\text{DET}\left(\begin{bmatrix} 1 & 1 & 2 & 0 \\ 4 & 0 & 3 & 1 \\ a_1 & 2 & 2a_1+a_2 & 1 \\ 4a_1+a_2 & 3 & 3a_1 & 4 \end{bmatrix} \right) \overset{?}{=} 0 \qquad \text{B.25}$$

We have assumed that the order of the model is 3, so (using eq. B.16, B.23):

$$\text{DET}(B_2) = \text{DET}\left(\begin{bmatrix} 1 & 0 & 2 & 3 \\ 0 & 1 & 1 & 4 \\ 2 & 3 & 2a_1+a_2 & 3a_1 \\ 1 & 4 & a_1 & 4a_1+a_2 \end{bmatrix} \right) := 0 \qquad \text{B.26}$$

$$==> 5a_1^2 + 6a_1a_2 + a_2^2 - 30a_1 - 26a_2 + 25 = 0 \qquad \text{B.27}$$

This restriction automatically implies that equality B.25 is true (for the determinant of a matrix is equal to or only of different sign as the determinant of a transposed matrix with some rows changed!).
The set of linear equations (expression B.24) will have a unique solution if and only if the rank of matrix B_2 is 3. This implies that the determinants of all minors of B_2 are not equal to zero (see GANTMACHER [14 : p.p.239, def.4]):

$$\mathrm{DET}(\begin{bmatrix} 1 \end{bmatrix}) = 1$$

$$\mathrm{DET}(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}) = 1 \qquad \text{B.28}$$

$$\mathrm{DET}(\begin{bmatrix} 1 & 0 & 2 \\ 0 & 1 & 1 \\ 2 & 3 & 2a_1+a_2 \end{bmatrix}) = 2a_1 + a_2 - 7 \neq 0 \ (?)$$

By substituting $a_2 = 7 - 2a_1$ in equation B.27, we will see that:

$$a_2 + 2a_1 - 7 = 0 \quad \Longleftrightarrow \quad a_1 = 6 \text{ and } a_2 = -5 \qquad \text{B.29}$$

So the determinant of the third minor of B_2 will only be 0 if a_1=6 and a_2=-5: this is the unique set of minimal polynomial coefficients belonging to the second order model!

The unique solution of the set of linear equations B.24 will be:

$$x_{212} = \frac{2a_2 + 5}{2a_1 + a_2 - 7}$$

$$x_{211} = \frac{a_1 - 6}{2a_1 + a_2 - 7} \qquad \text{B.30}$$

$$x_{221} = \frac{5a_1 + 4a_2 - 10}{2a_1 + a_2 - 7}$$

Does this mean that the x parameters in eq. B.30 are independent? We did not yet use the minimal polynomial restriction r=2.

$$A = \begin{bmatrix} 0 & 1 & 0 \\ a_2 & a_1 & 0 \\ x_{212} & x_{211} & x_{221} \end{bmatrix} \qquad \text{B.31}$$

$$A^2 = \begin{bmatrix} a_2 & a_1 & 0 \\ a_1 a_2 & a_2 + a_1^2 & 0 \\ x_{211}a_2 + x_{221}x_{212} & x_{212} + x_{211}a_1 + x_{211}x_{221} & x_{221}^2 \end{bmatrix}$$

By applying the minimal polynomial relation (see eq. B.12), we will find the next restrictions:

$$x_{211}a_2 + x_{212}x_{221} - x_{212}a_1 = 0 \quad (1)$$
$$x_{212} + x_{211}x_{221} = 0 \quad (2) \qquad \text{B.32}$$
$$x_{221}^2 - x_{221}a_1 - a_2 = 0 \quad (3)$$

Out of eq. B.32 (3) we will calculate a value for x_{221} (two possibilities!).

By substituting B.32 (2) $x_{212} = -x_{211}x_{221}$ in B.32 (1):

$$\Longrightarrow \; x_{211}\,(x_{221}^2 - x_{221}a_1 - a_2) = 0 \qquad \text{B.33}$$

With B.32 (3): $\Longrightarrow$ x_{211} can be chosen at random and x_{212} is linearly dependent on x_{211}!

This means that the number of parameters in the canonical state space description N_c is equal to the number of parameters in the minimal polynomial description N_m (=10), for x_{121} is a structural zero and x_{212} depends on x_{211}.
An equivalent derivation can be made for the situation with Kronecker row indices n_{r1}=1 and n_{r2}=2!

B.1.3 A fourth order representation.

In the given situation and a fourth order model, the Kronecker row indices will always be (see equation B.6):

$$n_{r1} = 2 \qquad \text{B.34}$$
$$n_{r2} = 2$$

The canonical observable state space description will be:

$$A = \left[\begin{array}{cc:cc} 0 & 1 & 0 & 0 \\ x_{112} & x_{111} & x_{122} & x_{121} \\ \hdashline 0 & 0 & 0 & 1 \\ x_{212} & x_{211} & x_{222} & x_{221} \end{array}\right] \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \\ b_{41} & b_{42} \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad \text{B.35}$$

Applying the restriction of the minimal polynomial with degree 2 (see eq. B.9):

$$x_{121} = x_{122} = 0 \qquad \text{B.36}$$
$$x_{211} = x_{212} = 0 \qquad (\text{ 4 restrictions!})$$

Using matrix A in expression B.35:

$$A^2 = \begin{bmatrix} x_{112} & x_{111} & 0 & 0 \\ x_{111}x_{112} & x_{112}+x_{121}^2 & 0 & 0 \\ 0 & 0 & x_{222} & x_{221} \\ 0 & 0 & x_{221}x_{222} & x_{222}+x_{221}^2 \end{bmatrix} \qquad \text{B.37}$$

And by using the expression of eq. B.37 in the minimal polynomial (see eq. B.12):

$$A^2 - a_1 A - a_2 I_4 = NULL_4 \qquad \text{B.38}$$

Substituting the expressions for A and A^2 (of B.36 and B.37) in eq. B.38, we will find the next restrictions and conditions for the Kronecker indices:

$$x_{111} = x_{221} = a_1 \qquad \text{B.39}$$
$$x_{112} = x_{222} = a_2$$

Generally the parametrization of a fourth order model (2 inputs and 2 outputs!) in a canonical observable form consist of 16 parameters.
We have proven that there are 6 restrictions on the parameter choice, so there are only 10 independent parameters left: the number of parameters in the canonical state space description is equal to the number of parameters in the minimal polynomial model!
These results are according to the previous theoretical part of this appendix.

Appendix C OVERRATING THE DEGREE OF THE MINIMAL POLYNOMIAL.

For real systems the degree of the minimal polynomial r will be unequal to minimal order of the system n. If the r' of the estimation model will be put equal to this n, the final dimension of the model woll be n x min(p,q), but no extra poles will be added: the existing poles will become multiple distinct.
Now the question arises, what happens if the r' of the estimation model is chosen too high.
Let the actual r of the system be 2 and the r' of the estimation model be 3 then:
So in this case we will find relation C.1 for this situation (r=2):

$$M(i) = a_1 M(i-1) + a_2 M(i-2) \qquad \text{C.1}$$

But we have forced a degree of the minimal polynomial equal to 3 (r'=3):

$$M(i) = a'_1 M(i-1) + a'_2 M(i-2) + a'_3 M(i-3) \qquad \text{C.2}$$

By applying eq. C.1 once again (with a delay of one sample), we can find an expression for M(i-3):

$$M(i-3) = \frac{1}{a_2} M(i-1) - \frac{a_1}{a_2} M(i-2) \qquad \text{C.3}$$

Substituting C.3 in equation C.2:

$$M(i) = \left(a'_1 + \frac{a'_3}{a_2} \right) M(i-1) + \left(a'_2 - \frac{a'_3}{a_2} a_1 \right) M(i-2) \qquad \text{C.4}$$

So the a'-parameters can not be chosen arbitrarily: A combination of equation C.1 and C.4 results in the restrictions in eq. C.5.

$$a'_1 + \frac{a'_3}{a_2} = a_1 \qquad a'_2 - \frac{a'_3}{a_2} a_1 = a_2 \qquad \text{C.5}$$

This third order polynomial can be divided in a second order part and a first order part:

$$z^3 - a_1' z^2 - a_2' z - a_3' =$$

$$(z^2 - a_1 z - a_2) (z + \frac{a_3'}{a_2}) \qquad \text{C.6}$$

This means that by overrating the degree of the minimal polynomial as 3 instead of 2, an extra root will be introduced:

$$z = - \frac{a_3'}{a_2} \qquad \text{C.7}$$

In eq. C.5 we have stated restrictions on a_1', a_2' and a_3', but the choice of one minimal polynomial coefficient a_i' can be chosen at random. So the place of the introduced extra root will also be arbitrary!

In section 2.2 of [11], a model has been derived from the Hankel model which is suited for the estimation of Markov parameters.
If we suppose that the initial conditions of the system are zero i.e. $\underline{X}(0) = \underline{0}$, we can represent the system with the matrix equation D.1

$$\begin{bmatrix} \underline{y}^T(h) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \underline{y}^T(h+m) \end{bmatrix} = \begin{bmatrix} \underline{u}^T(h) & \underline{u}^T(h-1) & \cdot & \cdot & \underline{u}^T(h-m) \\ \underline{u}^T(h+1) & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \underline{u}^T(h+m) & \cdot & \cdot & \cdot & \underline{u}^T(h) \end{bmatrix} \cdot \begin{bmatrix} M^T(0) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ M^T(m) \end{bmatrix} + N \qquad D.1$$

So the output measurements are corrupted with noise. If we want to truncate the impulse response after k samples, we can rewrite equation D.1 in a different form, shown in eq. D.2.

$$\begin{bmatrix} \underline{y}^T(h) \\ \cdot \\ \cdot \\ \cdot \\ \underline{y}^T(h+m) \end{bmatrix} = \begin{bmatrix} \underline{u}^T(h) & \underline{u}^T(h-1) & \cdot & \cdot & \underline{u}^T(h-k) \\ \underline{u}^T(h+1) & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \underline{u}^T(h+m) & \cdot & \cdot & \cdot & \underline{u}^T(h+m-k) \end{bmatrix} \cdot \begin{bmatrix} M^T(0) \\ \cdot \\ \cdot \\ \cdot \\ M^T(k) \end{bmatrix} +$$

$$\begin{bmatrix} \underline{u}^T(h-k-1) & \underline{u}^T(h-k-2) & \cdot & \underline{u}^T(h-m) \\ \underline{u}^T(h-k) & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \underline{u}^T(h+m-k-1) & \cdot & \cdot & \underline{u}^T(h) \end{bmatrix} \cdot \begin{bmatrix} M^T(k+1) \\ \cdot \\ \cdot \\ \cdot \\ M^T(m) \end{bmatrix} + \begin{bmatrix} \underline{n}^T(h) \\ \cdot \\ \cdot \\ \cdot \\ \underline{n}^T(h+m) \end{bmatrix} \qquad D.2$$

or: $Y = S_k^T M_k + S_i^T M_i + N$

In equation D.2, $S_i^T M_i$ represents the influence of what is called the "tail" of the Markov parameters, so the part of the impulse response that represents the middle and low frequency behaviour. If we estimate only k Markov parameters, the contribution of the latter part causes a truncation error.

We are looking for an estimate of M_k in such a way that the equation error E is minimal (in least squares sence) with E :

$$E = Y - S_k^T \hat{M}_k \qquad \text{D.3}$$

A brief review on least squares solutions has been given in chapter 3. Using these results we find following expression:

$$\begin{aligned} \hat{M}_k &= (S_k S_k^T)^{-1} S_k Y \\ &= (S_k S_k^T)^{-1} S_k (S_k^T M_k + S_i^T M_i + N) \qquad \text{D.4} \\ &= M_k + (S_k S_k^T)^{-1} S_k \{(S_i^T M_i) + N)\} \\ &\triangleq M_k + \Theta_i \end{aligned}$$

The error Θ_i indicates the difference between the expression M_k with exact Markov parameters and an estimate of M_k. During our simulations we have used a noise input signal from a Gaussian white noise source with zero mean value and a standard deviation of σ_u (a normal distribution!). The additive output noise is also assumed to be white and channel independent with zero mean value and standard deviation σ_n.

We will compare the part of Θ_i caused by the truncation of the Markov parameters with the contribution of the output noise to the error Θ_i. So we will have to compare elements of the products of matrices $S_k S_i^T M_i$ and $S_k N$.

We will first take a look at the the elements of $S_k S_i^T M_i$.

Part of the first matrix multiplication is shown in eq. 3.17:

$$S_k\, S_i^T =$$

$$\begin{bmatrix} \underline{u}(h) & \cdot & \cdot & \cdot & \underline{u}(h+m) \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \underline{u}(h-k) & \cdot & \cdot & \cdot & \underline{u}(h+m-k) \end{bmatrix} \cdot \begin{bmatrix} \underline{u}^T(h-k-1) & \cdot & \cdot & \cdot & \underline{u}^T(h-m) \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \underline{u}^T(h-k+m-1) & \cdot & \cdot & \cdot & \underline{u}^T(h) \end{bmatrix} = \qquad \text{D.5}$$

$$\left[\begin{array}{c:c:c:c:c:c} \sum_{s=0}^{m} \underline{u}(h+s)\underline{u}^T(h-k-1+s) & \sum_{s=0}^{m} \underline{u}(h+s)\underline{u}^T(h-k-2+s) & & \cdot & & \cdot & \sum_{s=0}^{m} \underline{u}(h+s)\underline{u}^T(h-m+s) \\ \sum_{s=0}^{m} \underline{u}(h+s-1)\underline{u}^T(h-k-1+s) & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \sum_{s=0}^{m} \underline{u}(h+s-k)\underline{u}^T(h-k-1+s) & \cdot & \cdot & \cdot & \sum_{s=0}^{m} \underline{u}(h+s-k)\underline{u}^T(h-m+s) \end{array}\right] =$$

$$
\left[\begin{array}{c:c:c:}
\sum_{s=0}^{m} u_1(h+s)u_1(h-k-1+s) & \sum_{s=0}^{m} u_1(h+s)u_2(h-k-1+s) & \cdot \quad \cdot \\
\sum_{s=0}^{m} u_2(h+s)u_1(h-k-1+s) & \sum_{s=0}^{m} u_2(h+s)u_2(h-k-1+s) & \cdot \quad \cdot \\
\cdot & \cdot & \cdot \quad \cdot \\
\sum_{s=0}^{m} u_q(h+s)u_1(h-k-1+s) & \sum_{s=0}^{m} u_q(h+s)u_2(h-k-1+s) & \cdot \quad \cdot \\
\cdot & \cdot & \cdot \quad \cdot \\
\cdot & \cdot & \cdot \quad \cdot \\
\cdot & \cdot & \cdot \quad \cdot \\
\sum_{s=0}^{m} u_q(h-k+s)u_1(h-k-1+s) & \sum_{s=0}^{m} u_q(h-k+s)u_2(h-k-1+s) & \cdot \quad \cdot
\end{array}\right.
$$

$$
\left.\begin{array}{:c:c:c}
\sum_{s=0}^{m} u_1(h+s)u_1(h-m+s) & \cdot \quad \cdot & \sum_{s=0}^{m} u_1(h+s)u_q(h-m+s) \\
\sum_{s=0}^{m} u_2(h+s)u_1(h-m+s) & \cdot \quad \cdot & \sum_{s=0}^{m} u_2(h+s)u_q(h-m+s) \\
\cdot & \cdot \quad \cdot & \cdot \\
\cdot & \cdot \quad \cdot & \cdot \\
\sum_{s=0}^{m} u_q(h-k+s)u_1(h-m+s) & \cdot \quad \cdot & \sum_{s=0}^{m} u_q(h-k+s)u_q(h-m+s)
\end{array}\right]
$$

As mentioned before $\{u_i\}_{i=1,2..p}$ has an standard normal distribution. The probability density function is expressed by equation D.6:

$$P(u_i) = \frac{1}{\sqrt{2\pi}\sigma_u} \exp\left(\frac{-u_i^2}{2\sigma_u^2}\right) \quad \text{with } i=1,2..\ p \qquad \text{D.6}$$

We will try to find the expectation and the mean value of the multiplication $z_{ij} = u_i\ u_j$, so we will have to derive the distribution function of z_{ij}. Suppose that u_i and u_j (with i=1,2..p, j=1,2..p <u>and</u> $i \neq j$, for all t)are independent and also that $u_i(t_1)$ and $u_i(t_2)$ (with i=1,2..p <u>and</u> $t_1 \neq t_2$) are independent.
This means that the probability of z_{ij} is:

$$P(z_{ij}) = P(\ u_i(t_1)\ u_j(t_2)\ ) = P(u_i(t_1))\ P(u_j(t_2)) \qquad \text{D.7}$$

$$= \frac{1}{2\pi\sigma_u^2} \exp\left(\frac{u_i^2(t_1) + u_j^2(t_2)}{-2\sigma_u^2}\right) \quad \text{with } i=1,2..\ p,\ j=1,2..p$$

<u>and</u> $t_1 \neq t_2$ if $i = j$

The expectation of z_{ij} will be (by using eq. D.7):

$$E(z_{ij}) = \int_{-\infty}^{\infty} z_{ij}\ P(z_{ij})\ dz_{ij} \qquad \text{D.8}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} u_i(t_1)\ u_j(t_2)\ P(u_i(t_1))\ P(u_j(t_2))\ du_i\ du_j$$

$$= \int_{-\infty}^{\infty} u_i(t_1)\ P(u_i(t_1))\{\int_{-\infty}^{\infty} u_j(t_2)\ P(u_j(t_2))\ du_j\}\ du_i$$

$$= \int_{-\infty}^{\infty} u_i(t_1)\ P(u_i(t_1))\{\ E(\ u_j(t_2))\}\ du_i = 0$$

So the standard deviation of z_{ij} will be (by using eq. D.7 and eq. D.8):

$$\sigma^2_{z_{ij}} = E(z^2_{ij}) + [E(z_{ij})]^2 = E(z^2_{zij})$$

$$= \int_{-\infty}^{\infty} z^2_{ij} \; P(z_{ij}) \, dz_{ij}$$

D.9

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u^2_i(t_1) \; u^2_j(t_2)P(u_i(t_1)) \; P(u_j(t_2)) \; du_i du_j$$

$$= \int_{-\infty}^{\infty} u^2_i(t_1) \; P(u_i(t_1))\{ \int_{-\infty}^{\infty} u^2_j(t_2) \; P(u_j(t_2))du_j\}du_i$$

$$= \int_{-\infty}^{\infty} u^2_i(t_1) \; P(u_i(t_1))\{ E(u^2_j(t_2))\} \; du_i$$

$$= \sigma^2_u \int_{-\infty}^{\infty} u^2_i(t_1) \; P(u_i(t_1)) \; du_i$$

$$= \sigma^2_u \, E(u^2_i(t_1)) = \sigma^4_u$$

Now we are able to compute the standard deviation of a summation of m+1 <u>independent</u> z_{ij} values:

$$E((\sum_{s=0}^{m} z_{ij}(s))^2) = E((\sum_{s=0}^{m} u_i(t_1+s)u_j(t_2+s))^2)$$

D.10

$$= (m+1) \, E(z^2_{ij})$$

In the simulations during the tests of the GERTH algorithm we have used data files with 1000 samples, so in those cases $m+1 = 1000$.
But after all we are interested in the standard deviation of the elements of the matrix multiplication $S_k \, S_i^T \, M_i$. The upperbound will be:

$$
\sigma_u^2 \sqrt{1000} \left| \begin{array}{cccc} 1 & 1 & \cdot & 1 \\ 1 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ 1 & \cdot & \cdot & 1 \end{array} \right| \cdot \left| \begin{array}{cccc} M_{11}(k+1) & M_{21}(k+1) & \cdot & M_{q1}(k+1) \\ M_{12}(k+1) & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ M_{1p}(k+1) & \cdot & \cdot & M_{qp}(k+1) \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ M_{11}(m) & M_{21}(m) & \cdot & M_{q1}(m) \\ M_{12}(m) & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ M_{1p}(m) & \cdot & \cdot & M_{qp}(m) \end{array} \right| =
$$

D.11

$$
\sigma_u^2 \sqrt{1000} \left| \begin{array}{cccc} \sum_{s=k+1}^{m} \sum_{i=1}^{p} |M_{1i}(s)| & \cdot & \cdot & \sum_{s=k+1}^{m} \sum_{i=1}^{p} |M_{qi}(s)| \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \sum_{s=k+1}^{m} \sum_{i=1}^{p} |M_{1i}(s)| & \cdot & \cdot & \sum_{s=k+1}^{m} \sum_{i=1}^{p} |M_{qi}(s)| \end{array} \right|
$$

Output noise matrix N also consists of white noise samples with zero mean value and standard deviation σ_n. So the preceding derivation (eq. D.6 till eq. D.11) of the standard deviation of the elements of matrix $S_K \, S_i^T$ can also be applied to find the standard deviation of the elements of matrix $S_K \, N$.
It is easy to see that the mean value of these elements is also zero and the variance will be $(m+1) \, \sigma_n^2 \, \sigma_u^2$.

By inserting the standard deviation σ_u of the input signals and taking the Markov parameters of a certain system in eq. D.11, we can estimate the value of σ_n in such a way that the standard deviation of the elements of matrices $S_k\, S_i^T\, M_i$ and $S_k\, N$ are the same. In the following example we will estimate this value of σ_n.
In that case the contribution of the output noise to error Θ_i will have the same value as the part due to truncation of the impulse response.
We mention that in the case of rather slowly decreasing impulse responses the output signal to noise ratio has to be substantial to exceed the effects of the truncation error.

D.1 Example of the estimation of the standard deviation of the output noise.

We will excite SYSTEM 3 (see chapter 4), a 3-input/2-output system with inputsignals consisting of white noise with a standard normal distribution (so the average is 0 and the standard deviation σ_u is 1). By truncating the impulse response after k=8 Markov parameters (M(0) has not been considered!) and 1000 measurements (m+1=1000) we will find:

$$\sum_{s=k+1}^{1000} \sum_{i=1}^{3} | M_{1i}(s) | \simeq 4.3$$
$$\sum_{s=k+1}^{1000} \sum_{i=1}^{3} | M_{2i}(s) | \simeq 9.0 \qquad \text{D.12}$$

or:

$$\text{output 1:} \quad 4.3\, \sigma_u^2 \simeq \sigma_n \sigma_u$$
$$\text{output 2:} \quad 9.0\, \sigma_u^2 \simeq \sigma_n \sigma_u \qquad \text{D.13}$$

So we will compare the "cross-powers" of the truncation noise and the additive output noise with the energy of the truncation noise weighed by the Markov parameters.
So with a maximal standard deviation of the additive output noise is about 9 the contribution of the truncation will be comparible with the

part due to the output noise corruption.
In this particular case (using SYSTEM 3) the standard deviation of 9 will be found if the output signal to noise ratio is about 10 dB.
If we take a look at the results of the tests of the GERTH algorithm (tabel 5) we find:

$$ERRO_{30} = 0.43 \quad ERRO_{10} = 0.43 \quad ERRO_{00} = 0.47 \qquad \text{D.16}$$

The results are according to our estimation: the contribution of the additive output noise on the equation error Θ_i becomes significant during simulations with an output signal to noise ratio less or equal to 10 dB.

Appendix E ESTIMATION OF THE ORDER OF A SYSTEM

In the preceding part of this report, we have assumed that a finite set of noise corrupted Markov parameters is available. We also have assumed to know the degree of the minimal polynomial of the system under consideration. In general however, we do not know this degree and we have to estimate it. It is not possible to find the degree of the minimal polynomial from of a sequence of estimated Markov parameters. We only can estimate the order of the system from an available set of estimated Markov parameters $\{\tilde{M}_i\}_{i=1,2..k}$.

In recent publications several estimation methods for the order of the system have been described ([17], [18].

The order n of the system can be greater than the degree of the minimal polynomial r (see appendix B). Now we assume that this degree is equal to the order of the system. In case the order n is not equal to the degree r extra meaningless eigenvalues will be introduced (see appendix C).

First we will take a look at the deterministic situation (no noise on the available set of Markov parameters). We will compose a Hankel matrix from this set of Markov parameters.

The definition of the block Hankel matrix H_{st} is given in eq. E.1.

$$H_{ts} = \begin{bmatrix} M(1) & M(2) & \cdots & M(s) \\ M(2) & & & \cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ \cdot & & & \cdot \\ M(t) & \cdots & & M(t+s-1) \end{bmatrix} \tag{E.1}$$

Schwarz [19] has stated that in the deterministic case:

$$\text{rank } \{ H_{ts} \} = n \tag{E.2}$$

We have assumed that the order of the system is equal to the degree of the minimal polynomial. So by calculating the rank of the Hankel matrix, we can determine the order of the system and the degree of the minimal polynomial.

In noisy situations we assume that the Markov parameters are corrupted with white "entry independent" noise. In appendix B has been stated that in those cases the Hankel matrices always will be of full rank (the rank of the Hankel matrices will be equal to the minimal dimension of the matrix):

$$\text{rank}\{ H_{ts} \} = \min(ps , qt) \qquad \text{E.3}$$

The problem will be to determine the dimension of the space that is significant above the space (image) spanned by the noise.
One method to test the order of a system is based on a comparison of the energy of the additive noise to the energy of the basis of the Hankel matrix
We only will take a brief look at an order test based on the examination of the singular values of the Hankel matrix.

E.1 The Singular Value Decomposition of the Hankel matrix.

The Singular Value Decomposition (SVD) of the Hankel matrix can be used for a test of the order of a system. The SVD is an eigenvalue like concept with remarkable numerical qualities, that is based on the following theorem (c.f. [18 : eq.13]):
Theorem 1 For any gxh matrix A with rank{A} = n, there exists a factorization:

$$A = W . D . V^T \qquad \text{E.4}$$

where:

$j=\min(g,h)$,
W is a orthonormal matrix (i.e. $W^T.W=I_j$),
V is a orthonormal matrix (i.e. $V^T.V=I_j$),
$D=\text{diag}(\delta_1, \delta_2, \ldots, \delta_j)$ with:

$$\delta_1 \geq \delta_2 \geq \ldots \geq \delta_n > 0 \quad \text{and}$$
$$\delta_{n+1} = \delta_{n+2} = \ldots = \delta_j = 0$$

The diagonal elements δ_i are called the singular values of A and the columns of W and the columns of V, respectively are the left and the right singular vectors of A.
Because W and V are non-singular matrices, the rank of A is equal to the rank of D. This implies that the rank of A is equal to the number of non-zero singular values. We can rewrite this as:

$$A = A_n = W_n \cdot D_n \cdot V_n^T \qquad E.5$$

where for A_n only the first n nonzero singular values are used in D_n.
We also mention that the sum of the squares of all singular values of matrix A represents the total energy in the set of column vectors of A (see J. STAAR and J.VANDEWALLE [16]: eq. (13)).
In the noisy cases the Hankel matrix will be of full rank, so all singular values of this matrix will be unequal to 0.
We will think the noise corrupted Hankel matrix being divided in a deterministic and a part containing the noise (for illustration).

$$\tilde{H}_{ts} = H_{ts}\big|_{det} + H_{ts}\big|_{noise} \qquad E.6$$

Although the Markov parameters appear repeatedly in the Hankel marix, we assume that the noise on the elements of matrix H_{st} consists of white noise samples (mean value is 0 and standard deviation is σ_n).
This means that the noise power in each direction will be the same and equal to the square of the standard deviation σ_n.
We are interested in the distribution of the energy in the Hankel matrix, so we will take a look at $H_{ts} . H_{ts}^T$ if $qt \leq ps$ or at $H_{ts}^T . H_{ts}$ if $ps \leq qt$.
By using the singular value decomposition of $H_{ts}|_{det} = V.D.W^T$ then [18: appendix]:

$$E \{ \tilde{H}_{ts} \cdot \tilde{H}_{ts}^T \} = V (D + \sigma_n^2 \, ps \, I_{qt}) W \qquad \text{if } qt \leq ps$$

$$E \{ \tilde{H}_{ts}^T \cdot \tilde{H}_{ts} \} = V (D^2 + \sigma_n^2 \, qt \, I_{ps}) W \qquad \text{if } ps \leq qt \qquad E.7$$

Or generally if $\tilde{H}_{ts} = \tilde{V}\,\tilde{D}\,\tilde{W}^T$:

$$E\ \{\ \tilde{D}^2\ \} = D^2 + \sigma_n^2\ \max(qt,ps)\ I_{\min(qt,ps)} \qquad E.8$$

So the expectation of the squared singular values of H_{ts} will be equal to the sum of the expection of the squares of the singular values in the deterministic case and max(qt,ps) x σ_n^2.
So only the singular values δ_i of the noise corrupted matrix H_{st} satisfying expression E.7 are significant for the energy in the "signal" (without any noise).

$$\delta_i \geq \max(ps,qt)\ \sigma_n^2\ R_0 \qquad E.9$$

In expression E.7 R_0 is a measure for the "distance" between the noise level and the smallest singular value taken into account (see further [16 : chapter 7.2]).

Appendix F THE RESULTS OF THE TESTS ON THE DIRECT METHOD IN EUCLIDEAN AND MAXIMUM NORM.

F.1 The results of the simulations on SYSTEM 4 with SN = 20 dB.

	8 Markov parameters		50 Markov parameters	
	maximum norm	euclidean norm	maximum norm	euclidean norm
EXACTMARK	av= .687 E-3 sd= .215 E-3	av= .684 E-3 sd= .157 E-3	av= .287 E-2 sd= .022 E-2	av= .287 E-2 sd= .016 E-2
GERTH ALGORITHM	av= .559 E-3 sd= .203 E-3	av= .551 E-3 sd= .128 E-3	av= .601 E-3 sd= .214 E-3	av= .592 E-3 sd= .147 E-3
DIRECT METHOD	av= .112 E-2 sd= .043 E-2	av= .105 E-2 sd= .032 E-2	av= .178 E-2 sd= .065 E-2	av= .168 E-2 sd= .054 E-2

table 9 : The results of the simulations on SYSTEM 4 with SN = 20 dB.
Remark: ´av´ indicates the average while ´sd´ stands for standard deviation.

F.2 The results of the simulations on SYSTEM 4 with SN = 40 dB.

	8 Markov parameters		50 Markov parameters	
	maximum norm	euclidean norm	maximum norm	euclidean norm
EXACTMARK	av= .751 E-5 sd= .225 E-5	av= .824 E-5 sd= .158 E-5	av= .219 E-2 sd= .002 E-2	av= .219 E-2 sd= .002 E-2
GERTH ALGORITHM	av= .617 E-5 sd= .200 E-5	av= .670 E-5 sd= .144 E-5	av= .659 E-5 sd= .216 E-5	av= .716 E-5 sd= .161 E-5
DIRECT METHOD	av= .111 E-4 sd= .047 E-4	av= .106 E-4 sd= .035 E-4	av= .170 E-4 sd= .065 E-4	av= .161 E-4 sd= .054 E-4

table 10 : The results of the simulations on SYSTEM 4 with SN = 40 dB.

F.4 The results of the simulations on SYSTEM 4 with SN = 80 dB.

	8 Markov parameters		50 Markov parameters	
	maximum norm	euclidean norm	maximum norm	euclidean norm
EXACTMARK	av= .141 E-5 sd= .069 E-5	av= .145 E-5 sd= .145 E-5	av= .219 E-2 sd= .069 E-5	av= .219 E-2 sd= .145 E-5
GERTH ALGORITHM	av= .114 E-5 sd= .056 E-5	av= .120 E-5 sd= .056 E-5	av= .124 E-5 sd= 0.56 E-2	av= .133 E-3 sd= .056 E-5
DIRECT METHOD	av= .924 E-9 sd= .310 E-9	av= .914 E-9 sd= .280 E-9	av= .106 E-10 sd= .04 E-10	av= .107 E-10 sd= .04 E-10

table 12 : The results of the simulations on SYSTEM 4 with SN = 80 dB.

LIST OF LITERATURE REFERENCES.

1. Abramowitz M. and A. Stegun
HANDBOOK OF MATHEMATICAL FUNCTIONS
Dover 1970
ISBN 0-486-61272-4.

2. Bosgra O.H.
ON PARAMETRIZATIONS FOR THE MINIMAL PARTIAL REALIZATION PROBLEM
Systems and Control Letters, 1983, 3, p.p. 181 - 187

3. Damen A.A.H., R.P. Guidorzi, A.K. Hajdasiński and P.M.J. Van den Hof
ON MULTIVARIABLE PARTIAL REALIZATION
To be published in Intern. Journal of Control 1984

4. Gerth W.
ZUR IDENTIFIKATION UND MINIMALREALISIRUNG VON MEHRGRÖSSEN-ÜBERTRAGUNGSSYSTEMEN DURCH MARKOVPARAMETER
Ph. D. diss. Technischen Universität Hannover 1972

5. Guidorzi R.P.
CANONICAL STRUCTURES IN THE IDENTIFICATION OF MULTIVARIABLE SYSTEMS
Automatica, 1975, 11, p.p. 361 - 374

6. Hajdasiński A.K.
LINEAR MULTIVARIABLE SYSTEMS: Preliminary Problems in Mathematical Description, Modelling and Identification.
TH-Report 80-E-106. Eindhoven 1980
ISBN 90-6144-106-4.

7. Hajdasiński A.K., P. Eykhoff, A.J.W. van den Boom and A.A.H. Damen
THE CHOICE AND USE OF DIFFERENT MODEL SETS FOR SYSTEM IDENTIFICATION
Tutorial paper, 6th IFAC Symp. on Ident. and Syst. Param. Estim. Washington D.C. 1982

8. Kortas J.B.
BENADERDE REALISATIE GERTH-BENADERING MET SUCCESSIEVE SUBSTITUTIE (in Dutch)
Stage-verslag TH Eindhoven 1983

9. Numerical Algorithm Group Ltd.
FORTRAN LIBRARY MANUAL MARK 10 volume 3.
Oxford 1983

10. Vaessen J.
DOCUMENTATION OF THE SIMULATION PROGRAM SYSSIMUL
Report R-012 PICOS Philips Glass Eindhoven 1983

11. Vaessen J.
LEAST SQUARES AND MAXIMUM LIKELIHOOD ESTIMATION OF MARKOV PARAMETERS
M. Sc. Thesis University of Technology Eindhoven 1983

12. Weijden J.L.J.M. van der
DOCUMENTATION OF THE GERTH ALGORITHM
Report R-023 PICOS Philips Glass Eindhoven 1984

13. Golub G.H. and C. Reinsch
SINGULAR VALUE AND LEAST SQUARES SOLUTIONS
Numerische Mathematik, 1970, 14, p.p. 403 - 420

14. Gantmacher F.R.
THE THEORY OF MATRICES
New York 1977
ISBN 0-8284-0131-4

15. Kreyszig E.
INTRODUCTORY MATHEMATICAL STATISTICS
New York 1970
SBN 471-50730 X

16. Van den Hof P.M.J.
APPROXIMATE REALIZATION OF NOISY LINEAR MULTYVARIABLE SYSTEMS
Journal A, 1984, volume 25, no 1, p.p. 21 - 26

17. Staar J. and J. Vandewalle
SINGULAR VALUE DECOMPOSITION:
A reliable tool in the algorithmic analysis of linear systems
Journal A, 1982, volume 23, no 2, p.p. 69 - 74

18. Damen A.A.H., A.K. Hajdasiński and P.M.J. Van den Hof
APPROXIMATE REALIZATION BEASED UPON AN ALTERNATIVE TO THE HANKEL MATRIX: THE PAGE MATRIX
Systems and Control Letters, 1982, 2, p.p. 202 - 208

19. Schwarz H.
MEHRFACHREGELUNGEN
Springer Verlag,
Berlin 1971

20. Fletcher R. and C.M. Reeves
FUNCTION MINIMIZATION BY CONJUGATE GRADIENTS
Computer Journal, 1964, vol. 7, p.p. 149

21. Gill P.E. and W. Murray
QUASI-NEWTON METHODS FOR UNCONSTRAINED OPTIMIZATION
Journal of the Institute of Mathematics and its Applications, 1972, vol. 9, p.p. 91-108