INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL SCIENTIFIC AND CULTURAL ORGANIZATION

INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS

34100 TRIESTE (ITALY) - P.O.B. 586 - MIRAMARE - STRADA COSTIERA 11 - TELEPHONES: 224281/2/3/4/5/6
CABLE: CENTRATOM - TELEX 460392 - I

SMR.201/19

SECOND WORKSHOP ON MATHEMATICS IN INDUSTRY

(2 - 27 February 1987)

IDENTIFICATION OF LINEAR DYNAMICAL SYSTEMS. - II

M. HAZEWINKEL
CWI
P.O. Box 4057
1009 AB Amsterdam
The Netherlands

6 An off line identification algorithm

The algorithm to be described below is a ~~small~~ generalization of one described by Guidorzi [3] in that one can use any nice selection ~~method~~ instead of just the Kronecker one. For certain systems this is an important advantage. The particular implementation which is used in the tutorials and demonstrations ~~was~~ is based on the Kronecker selection only. (computer) The implementation was done by F.-J. Pfreundt of the Technomathematik Group in Kaiserslautern

So suppose we have input-output data $u(0), u(1), \ldots, u(N)$; $y(0), y(1), \ldots, y(N)$ where N is some sufficiently large number. We are looking for some system (A,B,C) and initial condition $x(0)$ which explains the data ~~in terms~~, i.e. such that the relations (1.5) hold:

$$(6.1) \qquad y(n) = CA^n x(0) + CA^{n-1}Bu(0) + \cdots + CABu(n-2) + CBu(n-1)$$

Of course the real test of the model would be to use ~~there~~ say half the data to identify the model and then to use the other half as a comparison to what is generated by the model to check how good the model is

It can be assumed that (A,B,C) is completely observable (because the outputs will never see the unobservable part). It can not be really assumed that (A,B,C) is completely reachable because a priori the initial state $x(0)$ could have been in a part of state space which is not reachable from zero

Thus we know that there is a nice output selection κ such that $\tilde{O}(A,C)_\kappa$ is invertible. Let κ be $(\kappa_1, \ldots, \kappa_p)$. For instance $\kappa = (1,3,2)$

$$\begin{array}{ccc} \times & \times & \times \\ \cdot & \times & \times \\ \cdot & \times & \cdot \\ \vdots & \vdots & \vdots \end{array}$$

At this stage $A, C, Q(A,C)$ and K (and also n) are of course all still unknown. Now ~~if~~ because K is a nice selection with $Q(A,C)_K$ invertible we know that we know that the "successor vectors" $(CA^{\kappa_i})_i$ ~~are linear com~~, the i-th row of CA^{κ_i} (same i!) are linear combinations of the row vectors labelled by K, i.e. the $(CA^k)_j$ with $k \leq \kappa_j - 1$. In the illustration below the "successor vectors" have been indicated by $*$'s

$$\begin{array}{ccc} \times & \times & \times \\ * & \times & \times \\ \cdot & \times & * \\ \cdot & * & \cdot \\ \cdot & \cdot & \cdot \end{array}$$

In other words we have relations

$$(6.2) \qquad (CA^{\kappa_i})_i = \sum_{j=1}^{p} \sum_{k=0}^{\kappa_j - 1} \alpha_{ijk} (CA^k)_j$$

Now consider the i-th component of $y(t+\kappa_i)$ for all i

$$(6.3) \qquad y_i(t+\kappa_i) = (CA^{t+\kappa_i} x(0))_i + (CA^{t+\kappa_i-1} B u(0))_i + \cdots$$

$$+ (CA^{\kappa_i} B u(t-1))_i + (CA^{\kappa_i - 1} B u(t))_i + \cdots + (CBu(t+\kappa_i-1))_i$$

Now by postmultiplication with A^τ and $A^\tau B$ it follows from (6.2) that

$$(6.4) \qquad (CA^{\kappa_i+\tau})_i = \sum_{j=1}^{p} \sum_{k=0}^{\kappa_j - 1} \alpha_{ijk} (CA^{k+\tau})_j$$

$$(6.5) \qquad (CA^{\kappa_i+\tau}B)_i = \sum_{j=1}^{p} \sum_{k=0}^{\kappa_j - 1} \alpha_{ijk} (CA^{k+\tau}B)_j$$

and by further postmultiplication with the $x(0)$ and $u(t-\tau-1)$ this gives

$$(6\,6) \qquad (CA^{K_i+\tau}x(0))_i = \sum_{j=1}^{p}\sum_{k=0}^{\nu_j-1} \alpha_{ijk}(CA^{k+\tau}x(0))_j$$

$$(6\,7) \qquad (CA^{K_i+\tau}Bu(t-1-\tau))_i = \sum_{j=1}^{p}\sum_{k=0}^{\nu_j-1}(CA^{k+\tau}Bu(t-1-\tau))_i$$

Now substitute (66) and (67) in (63) in all the terms of the right hand side except the last K_i. Then collect terms again using (63) for the values $t-1, t-2, \dots t-K_i$. The result is

$$(6\,8) \qquad y_i(t+K_i) = \sum_{j=1}^{p}\sum_{k=0}^{K_j-1} \alpha_{ijk}\, y_j(t+k) + \sum_{j=1}^{m}\sum_{k=0}^{K_i-1} \beta_{ijk}\, u_j(t+k)$$

Here of course the α_{ijk} are the coefficients occurring in (62) and the β_{ijk} are the coefficients occurring in the matrices CA^kB, $k=0,\dots,K_i-1$. Note that the relations (68) hold for all t! Now given enough data the fact that there must be some relations (68) can of course be used to determine the α_{ijk} and, β_{ijk} and middle K_j directly from the input-output data. This can e.g. be done as follows. Form the vectors

$$y^i(t)(L) = \begin{pmatrix} y_i(t) \\ y_i(t+1) \\ \vdots \\ y_i(t+L) \end{pmatrix}, \quad u^i(t)(L) = \begin{pmatrix} u_i(t) \\ u_i(t+1) \\ \vdots \\ u_i(t+L) \end{pmatrix}$$

where L is some sufficiently large number. Because (68) holds for the components of the $y^i(t)(L)$ and $u^i(t)(L)$ the same relations hold between the vectors, i.e.

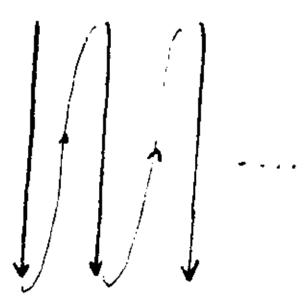
$$(6.9)\qquad y^i(t+K_i) = \sum_{j=1}^{p}\sum_{k=0}^{K_j-1} \alpha_{ijk}\, y^j(t+k) + \sum_{j=1}^{m}\sum_{k=0}^{K_i-1} \beta_{ijk}\, u^j(t+k)$$

So what we have to do is to study the linear ~~depency~~ dependency relations between the vectors

$$(6.10)\qquad \begin{array}{lllll}
y^1(t) & y^1(t+1) & \cdots & y^1(t+n-p) & y^1(t+n-p+1) \\
\vdots & & & & \vdots \\
y^p(t) & y^p(t+1) & \cdots & y^p(t+n-p) & y^p(t+n-p+1) \\
u^1(t) & u^1(t+1) & \cdots & u^1(t+n-p) & \\
\vdots & & & & \\
u^m(t) & u^m(t+1) & \cdots & u^m(t+n-p) &
\end{array}$$

where n is the smallest integer (or any integer) such that the $y^i(t+n-p+1)$ are linearly dependant on the vectors on the left of the last column. Then there will be a nice output selection $K=(K_1,\dots,K_p)$ such that for each i there is a relation (6.9) for certain $\alpha_{ijk}, \beta_{ijk}$. This determines both ~~the~~ K and the $\alpha_{ijk}, \beta_{ijk}$.

There may of course be different K (and correspondingly different $\alpha_{ijk}, \beta_{ijk}$) which work. Some K will be better than others, ~~and~~ One possible K is the Kronecker output nice selection which will be obtained by studying the linear dependencies of the collection (6.10) in the order.



This may not be the best κ to use. In fact at least theoretically that particular κ may be one of the worst or even the worst to use, cf. below. We are testing this out experimentally at the moment.

Testing the linear dependency relations between vectors can e.g. be done as follows. Consider a sequence of vectors

$$z_1, z_2, z_3, \ldots$$

For $n = 1, 2, 3, \ldots$ form the matrix $M_1 = (z_1)$, $M_2 = (z_1\ z_2)$, ... and $S_1 = M_1^T M_1$, $S_2 = M_2^T M_2$, ... Then calculate the smallest eigenvalue of the S_i; if that one is zero (near zero) for the first time at $n = n_1$, z_{n_1} is dependent on the previous ones. Calculate the coefficients and throw out z_{n_1} and continue.

For an array like (6.10) one should in principle do this for all ~~[illegible]~~ potential ~~[illegible]~~ nice selections. There are easy ways of generating all possibilities.

Of course one needs a test for which nice selection to pick of the many possibilities which exist. This is by no means a settled matter. One thing to look out for is ~~that~~ the property that the matrix

$$\left(y^1(t), \ldots, y^1(t+\kappa_1-1), y^2(t), \ldots, y^2(t+\kappa_2-1), \ldots, y^p(t), \ldots, y^p(t+\kappa_p-1)\right)$$

contains an $n \times n$ submatrix which is numerically well invertible. In particular the determinant of this submatrix should not be close to zero.

It would be worthwhile to ~~do~~ try to find out whether the socalled Akaike information criterium could play a useful role here.

7. Comments on the algorithm

7.1. The algorithm as it is yields the α_{ijk} and β_{ijk} (and a nice κ). Then of course relations (6.8) can immediately be used to ~~pre~~ predict the future behaviour of the system in terms of new inputs. We do not yet have the matrices A, B, C themselves and for some purposes (such as compensator or stabilizer design or Kalman filtering or decoupling or . .) this would be desirable.

We shall again try to find those A, B, C which have the additional property that $Q(A,C)_\kappa = I_n$. Then the A-matrix is made up out of unit vectors and vectors formed from the α_{ijk} exactly as in section 5 above (after all the α_{ijk} come from the dependencies between the vectors of $Q(A,C)$, cf (6.2) above).

Further knowing the β_{ijk} we know the matrices

$$CA^kB \,, \quad k = 0, 1, \dots, \max(\kappa_i) \tag{7.2}$$

Because $Q(A,C)_\kappa = I_n$ this is sufficient to find B as well. Indeed

$$(Q(A,C)B)_\kappa =$$

consists only of rows which belong to the rows of the matrices in (7.2).

It remains to find C. This may in fact not be possible (because (A,B,C) is not necessarily completely observable). With a bit of luck there is nice input selection α_R for which $R(A,B)_{\alpha_R}$ is invertible which involves only the now known matrices CA^kB, $k \le \max(\kappa_i)$. If not things ~~may~~ become a bit more complicated, even in the completely reachable case.

7.3 An example

Consider again the example

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \eta & 0 & 0 \\ 1 & \varepsilon & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & \varepsilon & 0 & 0 & 0 \end{pmatrix}$$

Notice that the Kronecker output selection is $\kappa = (4, 1)$ and it picks out the indicated vectors in $Q(A, C)$

$$Q(A,C) = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \eta & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \eta & \varepsilon\eta & 0 & \eta & 0 \\ 1 & \varepsilon & 0 & 1 & 0 \\ 0 & 0 & 2\eta+\varepsilon\eta^2 & 0 & \eta \\ 0 & 0 & 2+\varepsilon\eta & 0 & 1 \\ & & \cdots & & \\ & & \cdot\cdot & & \end{pmatrix}$$

It follows that $\det Q(A,C)_\kappa = \eta^3$ which is very small if η is say 0.1 making this a badly invertible matrix. Yet it is more or less this matrix which has to be inverted to calculate the α_{ijk}. Indeed the dependency relations (6.2):

$$\text{Ok'22 (7.4)} \qquad (CA^{\kappa_i})_i = \sum_{j=1}^{p} \sum_{k=0}^{\kappa_j - 1} \alpha_{ijk} (CA^k)_j$$

precisely say that a suitable matrix with the α_{ijk} as entries is equal to the inverse of the matrix formed by the $(CA^k)_j$, $0 \le k \le \kappa_j - 1$ and this inverse is precisely $Q(A,C)_\kappa$. In the algorithm itself the α_{ijk} are calculated differently ~~but on~~ (which may help numerical stability) but that does not change the fact that the true α_{ijk} (for this κ) are very sensitive to small changes in η

On the other hand there is a very good nice selection viz $(1,4)$ for which the corresponding determinant is 1 and in fact the corresponding matrix is very near the identity (for small ε and η) making it ideal for inversion

Thus on theoretical grounds this ~~to~~ $(1,4)$ would seem to be a much better choice as a selection to work with. The generalization described above in section 6 of course permits this

7.5 Choice of output dependence

Consider again the example 7.3 but now with a different output matrix namely with the two output channels switched, i.e. with

$$C' = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Now the Kronecker selection is again $(4,1)$. But so to speak the 4 now refers to the other output channel so that we are really talking about a very different system. This is theoretically a good reason to suspect that the results of the algorithm may be quite dependent on which way the outputs are labelled (if one insists on using the Kronecker output selection) Simply using the "best" nice κ would remove this undesirable feature

7.6 Parameter sensitivity (again)

Consider once more example 7.3. Note that the Kronecker output selection is $(4,1)$ for $\eta \neq 0$ and it is $(1,4)$ for $\eta = 0$. Thus for small η, given numerical roundoff one might well obtain the wrong Kronecker (and $\eta = 0$) selection. In case $\kappa_Q(A,B,C)$ is not a continuous function of the entries of (A,B,C) which is likely to introduce severe discontinuities into any algorithm which is based on calculating these indices at some intermediate stage.

Experimental calculations to test whether these potential extreme numerical sensitivities do indeed ~~[illegible]~~ materialize are being done this week.