Numerical Modelbuilding

Eklund, Karl

1968

Document Version:
Publisher's PDF, also known as Version of record

Link to publication

Citation for published version (APA):
Eklund, K. (1968). *Numerical Modelbuilding*. (Technical Reports TFRT-7004). Department of Automatic Control, Lund Institute of Technology (LTH).

Total number of authors:
1

LUND UNIVERSITY

PO Box 117
221 00 Lund
+46 46-222 00 00

NUMERICAL MODELBUILDING

K. EKLUND

REPORT 6808 NOVEMBER 1968
LUND INSTITUTE OF TECHNOLOGY
DIVISION OF AUTOMATIC CONTROL

NUMERICAL MODELBUILDING †

K. Eklund

ABSTRACT

This paper describes a systematic modelling technique which makes it possible to avoid many of the tedious elements that always are involved in the modelling of industrial processes. The technique is based on the nonlinear equations which are usually obtained from basic physical laws. It gives a systematic procedure to compute steady-state solutions and linearized equations from the nonlinear physical equations. The procedure is based on wellknown methods for solving nonlinear equations and numerical differentiation. The final result is a system description on standard form S(A,B,C,D). The essential difficulty is to find the smallest number of state variables and to assign these to the linearized model. A method for solving this problem is the main result of the paper. The paper also contains a FORTRAN program for the assignment of state variables as well as an application to the modelling of a boiler.

† This work has been supported by the Swedish Board for Technical Development under Contract 68-336-f.

TABLE OF CONTENTS

1. INTRODUCTION

It is a very tedious work to establish a linearized mathematical model of an industrial process of some complexity. Even if basic physical laws are applicable it can be extremely laborious to compute the steady-state values and to linearize the equations. In particular if we take into account that it is often highly desirable to develop linearized models for different operating conditions and to investigate the sensitivity of the model to basic physical parameters. It is also very difficult to check the linearized equations even if the basic equations themselves often are quite easy to check.

It thus seems rather attractive to develop an algorithm which enables a digital computer to perform all the tedious work. Such an algorithm is proposed in this report. We start with the basic nonlinear equations and proceed to compute steady-state values and linearize. The final result is a system description on standard form. The standard form is convenient because a large amount of control theory uses the state representation of the system as a starting point and since important physical variables can be retained as state variables in the final model. The procedure is based on wellknown methods for solving nonlinear equations and numerical differentiation. The only difficulty is to assign the smallest number of state variables to the linearized model. A method for solving this problem is the main result of the paper.

The procedure is outlined in section 2. In section 3 we give a procedure to find the smallest number of state variables for the linearized model. The main result theorem 1 is contained in this section. In section 4 we give an alternative formulation of some conditions of theorem 1. The assignment of state variables is not unique. In section 5 we discuss how this nonuniqueness can be exploited to choose state variables that are physically significant. A digital computer algorithm for the reduction is described in section 6. In section 7 we give an example of the application of the procedure to the modelling of a drum boiler. Some examples which illustrates the implications of the conditions of theorem 1 are given in section 8. Appendix A covers some material on pseudo inverses that are needed in the proof of theorem 1 and in Appendix B we give a FORTRAN listing of the reduction program.

2. A SYSTEMATIC MODEL REDUCTION TECHNIQUE

To establish a mathematical model of an industrial process it is often convenient to divide the process into a number of components. These components are treated separately. A set of equations which describe the dynamic and static relations between the inputs and outputs for each component are derived. The components are linked up with a number of internal variables which might be of secondary interest. This technique simplifies the derivation of the basic equations but introduces a number of auxiliary variables, ref {5}. The resulting mathematical model is usually a set of nonlinear equations which include both ordinary and partial differential equations. Partial differential equations are approximated by finite differences in the space coordinates. We also assume that the system has constant coefficients. The system behaviour for small disturbances about an equilibrium state is often of great interest. This behaviour might be described by the linearized system equations. Thus if we require the resulting model to be a set of linear ordinary differential equations the following systematic approach is proposed.

- The process is described by basic physical laws such as the laws of conservation of mass, energy and momentum. The resulting set of equations is

$$\begin{cases} f(\dot{v},v,u) = 0 & (2.1a) \\ g(y,v,u) = 0 & (2.1b) \end{cases}$$

  where f is an ℓ-vector whose components are nonlinear functions of the variables v, their time derivatives and the process input variables u. g is a k-vector whose components are nonlinear functions of the variables v, u and the process output variables y. The input vector u and the output vector y are identified and treated separately already at this stage. All other variables are included in v. The set of equations is consistent if the number of variables v equals ℓ and if y is a k-vector. We can always assume that the vector f does not depend on du/dt because we can then introduce a new input variable $u^* = du/dt$.

- The steady-state values are obtained if we put time derivatives equal to zero in equation (2.1) viz.

$$f(v,u) = 0 \tag{2.2a}$$

$$g(y,v,u) = 0 \tag{2.2b}$$

Given the steady-state values u^o of u equations (2.2a), (2.2b) determine the steady-state values v^o and y^o of v and y respectively. The zero solutions of the nonlinear equation (2.2) are obtained by standard techniques e.g. a Newton-Raphson method. Other methods are found in ref {1}, {2}, {6}, {7}.

- Linearize equation (2.1). We get

$$E\dot{v} + Fv + Gu = 0 \tag{2.3a}$$

$$Py + Qv + Ru = 0 \tag{2.3b}$$

where

$E = f_{\dot{v}}(0,v^o,u^o)$ $[\ell \times \ell]$

$F = f_v(0,v^o,u^o)$ $[\ell \times \ell]$

$G = f_u(0,v^o,u^o)$ $[\ell \times \ell]$

$P = g_y(y^o,v^o,u^o)$ $[k \times k]$

$Q = g_v(y^o,v^o,u^o)$ $[k \times \ell]$

$R = g_u(y^o,v^o,u^o)$ $[k \times m]$

The perturbed variables viz. the differences $v-v^o$, $y-y^o$ and $u-u^o$ are denoted as the variables themselves. The matrices of first partial derivatives of the vectors f and g with respect to $\dot{v}$, v and u and y,v and u respectively are obtained by numerical differences.

- The linearized set of equations (2.3) is reduced to standard form S(A,B,C,D) viz.

$$\begin{cases} \frac{dx}{dt} = Ax + Bu \\ y = Cx + Du \end{cases} \tag{2.4}$$

where x is the state n-vector, u is the input m-vector and y is the output k-vector. A,B,C and D are matrices of proper order.

The problem of reducing equation (2.3) to standard form is to introduce a proper number of state variables. A method which solves this problem is the main result of the paper and is presented in the following section.

- If the transfer function representations are needed these are easily obtained from equation (2.4) using standard methods, see e.g. ref {8}.

3. REDUCTION OF THE LINEARIZED SYSTEM EQUATIONS TO STANDARD FORM

Before the theorem is stated it is convenient to have a look at the structure of equation (2.3). It is immediately clear that the number of state variables at most equals the rank of E. If the rank of E equals ℓ the number of state variables is ℓ and equation (2.3a) can be solved directly using the inverse of E. The reduction to standard form in this case is trivial. However, in general the rank of E is less than ℓ and greater than zero. This is a consequence of two facts. Static relations between the variables v and u create rows of zeros in the matrix E.The number of time derivatives introduced might exceed the rank of E. That is we have additional static relations between the variables v and u which are not quite apparent.

If the inverse of P does not exist one or several of the outputs are a linear combination of the others. The linear dependent outputs may be excluded by computation of the linear independent rows of P. However, physical insight usually permit us to avoid this problem and there is no loss of generality if we assume that the inverse of P exists.

The proof of the theorem requires elementary knowledge of the concept of pseudo inverses. Some relevant properties of the pseudo inverse are given in Appendix A. A detailed presentation is found in ref {3}. Before the theorem is stated we introduce the following notations. R(T), N(T), $\rho(T)$ and T^+ denote the range space, the null space, the rank and the pseudo inverse of T respectively. The rank factorization of E used in the theorem gives E = KL and requires $\rho(E) > 0$. If $\rho(E) = 0$ there is no reduction problem and consequently we assume $\rho(E) > 0$.

<u>Theorem 1</u>

Given a linear dynamic system with constant coefficients described by

$$\begin{cases} E\dot{v} + Fv + Gu = 0 & (3.1a) \\ Py + Qv + Ru = 0 & (3.1b) \end{cases}$$

where v is an ℓ-vector, u is a m-vector and y is a k-vector and E,F,G,P,Q,R are matrices of proper order. If

$\rho((I-KK^{+})F) = \ell - n$ (i)

$R((I-KK^{+})F) \cap R(L) = 0$ (ii)

then the standard form of equation (3.1) becomes

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

where the state n-vector is

$$x = Lv$$

and

$$A = - K^{+}F\{L^{T}L + F^{T}(I-KK^{+})F\}^{-1}L^{T}$$
$$B = K^{+}\{F\{L^{T}L + F^{T}(I-KK^{+})F\}^{-1}F^{T}(I-KK^{+}) - I\}G$$
$$C = - P^{-1}Q\{L^{T}L + F^{T}(I-KK^{+})F\}^{-1}L^{T}$$
$$D = P^{-1}\{Q\{L^{T}L + F^{T}(I-KK^{+})F\}^{-1}F^{T}(I-KK^{+})G-R\}$$

Proof

The rank factorization of the matrix E is

$$E^{\{\ell x \ell\}} = K^{\{\ell x n\}} \cdot L^{\{n x \ell\}} \qquad (3.2)$$

where the matrices K and L both have rank n. Combining equations (3.2) and (3.1a) we get

$$KL\dot{v} + Fv + Gu = 0 \qquad (3.3)$$

Introduce the state n-vector x and set

$$x = Lv \qquad (3.4)$$

Notice that the rank of L is n. Equation (3.4) together with equation (3.3) yield

$$K\dot{x} + Fv + Gu = 0 \qquad (3.5)$$

If the vector -(Fv + Gu) lies in the column space of K a unique solution is obtained using the pseudo inverse of K.

$$\dot{x} = -K^{+}Fv - K^{+}Gu \qquad (3.6)$$

where

$$K^{+} = (K^{T}K)^{-1} K^{T}$$

Combining equations (3.5) and (3.6) we get

$$(I-KK^{+})Fv + (I-KK^{+})Gu = 0 \qquad (3.7)$$

Combine equation (3.4) and equation (3.7) to one equation. Hence

$$\begin{bmatrix} L \\ \hdashline (I-KK^+)F \end{bmatrix} v = \begin{bmatrix} x \\ \hdashline -(I-KK^+)Gu \end{bmatrix} \tag{3.8}$$

or

$$Tv = z \tag{3.9}$$

The vector x by definition lies in the column space of L. If condition (i) holds then

$$R(I-KK^+) = R((I-KK^+)F)$$

since the rank of $(I-KK^+)$ is ℓ-n (see Appendix A). Thus we have

$$R((I-KK^+)G) \subset R((I-KK^+)F)$$

Then it is clear that the vector z lies in the column space of T and a solution exists. If also condition (ii) is satisfied the rank of T is ℓ and the solution is unique. This implies in particular that there exists a unique solution to equation (3.7). Rewriting equation (3.7) we get

$$(I-KK^+)(Fv + Gu) = 0$$

then

$$(Fv + Gu) \in N(I - KK^+)$$

We also have

$$(I - KK^+)K = 0$$

then

$$k_i \in N(I - KK^+) \qquad i = 1, \ldots, n$$

where k_i is the column vectors of K. Thus the existence of a unique solution of equation (3.8) also implies the existence of a unique solution of equation (3.5). The solution of equation (3.8) is

$$v = \begin{bmatrix} L \\ \hdashline (I-KK^+)F \end{bmatrix}^+ \begin{bmatrix} x \\ -(I-KK^+)Gu \end{bmatrix}$$

Using that the matrix $(I-KK^+)$ is a symmetric projection (see 5 and 6 Appendix A) we get

$$v = \{L^TL + F^T(I-KK^+)F\}^{-1}L^Tx - \{L^TL + F^T(I-KK^+)F\}^{-1}F^T(I-KK^+)Gu \quad (3.10)$$

When equations (3.1b), (3.6) and (3.10) are combined the standard form of the original set of equations are obtained and the theorem is proved.

Notice that the theorem does only supply a solution of the reduction problem when the number of state variables equals the rank of E. This is not necessarily the case which is demonstrated by a simple example in section 8. All static relations between the variables v and u are given by the equation (3.7).

4. THE CHECK OF CONDITIONS

An algorithm which performs the rank factorization of the matrix E is a necessary subroutine of the reduction program. It is then convenient to formulate the conditions as rank conditions. Condition (i) is already on a suitable form. If condition (i) holds condition (ii) is equivalent to that the rank of T equals ℓ. Hence

$$\rho((I-KK^+)F) = \ell - n \qquad (i)$$

$$\rho(T) = \ell \qquad (ii)$$

The conditions are checked in this order. The implications of the conditions are illustrated in section 8 by simple numerical examples. The matrix T is defined by equation (3.9).

5. THE CHOICE OF STATE VARIABLES

The state variables are determined by

$$x = Lv$$

The matrix L is not uniquely determined but L is chosen to satisfy

$$E = KL$$

where the column vectors of K form a basis for the vector space generated of the column vectors of the matrix E. First let us consider the case when K is simply chosen as the n linear independent column vectors of the matrix E. Let E be arranged so that the n first columns are the linear independent ones. Then

$$E = KL = K[L_1|L_2] \qquad (5.1)$$

where

$L_1 = I \qquad n \times n$
$L_2 \qquad n \times \ell-n$

The matrix L_2 gives the coefficients in the expressions which constitute the ℓ-n linear dependent column vectors $e_{n+1},\dots,e_\ell$ of the matrix E. Hence if

$$e_{n+i} = \sum_{j=1}^{n} \lambda_{ji} e_j \qquad i = 1,\dots,\ell-n$$

then

$$(L_2)_i = [\lambda_{1i} \;\dots\dots\; \lambda_{ni}]^T \quad , \quad i = 1,\dots,\ell-n$$

where $(L_2)_i$ denotes the i:th column vector of the matrix L_2.

The variables v_i, $i = 1,\dots,\ell$ often have a simple physical interpretation. Then it is naturally of interest to retain the variables v_i as state variables if this is possible viz.

$$\begin{aligned} x_1 &= v_1 \\ x_2 &= v_2 \\ &\vdots \\ x_s &= v_s \qquad s \leq n \end{aligned}$$

The form of the matrix L then is

$$L = \left[\begin{array}{c|c} L_{11} & L_{12} \\ \hline L_{21} & L_{22} \end{array}\right]$$

where

$L_{11} = I \qquad s \times s$
$L_{12} = 0 \qquad s \times \ell-s$
$L_{21} \qquad n-s \times s$
$L_{22} \qquad n-s \times \ell-s$

However, in general it is not at all possible to have it this way. Suppose

$$E = [e_1 \dots e_n \quad e_{n+1} \dots e_\ell]$$

where

$$e_{n+i} = \sum_{j=1}^{n} \lambda_{ji}\, e_j \quad , \quad i = 1, \dots, \ell-n$$

and

$\lambda_{ji} \neq 0 \qquad \forall\ i,j$

Further no directions of vectors coincide. Let $k_1 \dots k_n$ be an arbitrary basis for the column space generated of $e_1 \dots e_n$.

Then we may write

$$\begin{bmatrix} e_1 & \dots & e_n & e_{n+1} & \dots & e_\ell \end{bmatrix} = \begin{bmatrix} k_1 & \dots & k_n \end{bmatrix} \left[\begin{array}{c|c} I & 0 \\ \hline L_{21} & L_{22} \end{array}\right] = \left[\begin{array}{c|c} K_{11} & K_{12} \\ \hline K_{21} & K_{22} \end{array}\right] \left[\begin{array}{c|c} I & 0 \\ \hline L_{21} & L_{22} \end{array}\right]$$

where

K_{11} s x s
K_{12} s x n-s
K_{21} ℓ-s x s
K_{22} ℓ-s x n-s

or

$$\left[\begin{array}{c|c} e_1 \dots e_s & e_{s+1} \dots e_n \quad e_{n+1} \dots e_\ell \end{array}\right] = \left[\begin{array}{c|c} K_{11} + K_{12}L_{21} & K_{12} \quad L_{22} \\ K_{21} + K_{22}L_{21} & K_{22} \quad L_{22} \end{array}\right]$$

Identify the last ℓ-s column vectors of the two matrices above. Hence

$$\begin{bmatrix} e_{s+1} & \dots & e_n & e_{n+1} & \dots & e_\ell \end{bmatrix} = \begin{bmatrix} k_{s+1} & \dots & k_n \end{bmatrix} L_{22}$$

Since we assumed that $\lambda_{ji} \neq 0 \ \ \forall\ i,j$ and that the vectors e_i are in pairs linearly independent, there are still n linear independent column vectors in the left hand matrix. The right hand matrix has n-s linear independent column vectors and thus the identity implies that s=0. Since in general no especial favorable choice of the state variables is available we may choose the form of the matrix L given by equation (5.1). If the coupling between the column vectors of the matrix E is not as elaborate as indicated above but e.g. only one vector, say e_1, is needed to establish the linear dependent ones, then this choice creates rows of zeros in the matrix L_2 except in the row corresponding to the vector e_1.

6. A DESCRIPTION OF THE REDUCTION PROGRAM

The reduction program RESTAF (Reduction to Standard Form) is presented in Appendix B. The program just computes the standard form matrices according to the formulas given in the theorem and checks the rank conditions. The subroutines used are MIART and DMATPROD. MIART computes the rank of matrices and inverts asymmetric matrices by the method of Gauss-Jordan with row-pivoting. DMATPROD performs matrix multiplication using a double precision scalar product. In order to avoid too cumbersome program listings auxilary subroutines concerned with standard matrix operations are omitted.

The program input is the linearized system matrices. The output is the result of the rank factorization, the standard form matrices and some intermediate results. For further details is referred to the program listing and the boiler application in section 7.

The computation of the pseudo-inverse of K requires the inverse of K^TK. It is wellknown that the multiplication with K^T worsen the condition of the problem and numerical difficulties may arise. The accuracy of the matrix K^+ is checked using the matrix K^+K. This matrix should equal the unit matrix. Thus K^+K is computed and printed.

7. APPLICATION TO A BOILER MODEL

In this section we will give an application of the reduction procedure to a practical problem. We will consider a drum-downcomer-riser loop of a drum boiler for a power station unit of approximately 150 MW. The drum pressure is 140 bar and the outlet steam temperature 530 $^{\circ}$C. The derivation of the basic nonlinear equations, computation of steady state values and linearization result in the matrices E,F,G,P,Q,R. The derivation of these results as well as a FORTRAN program for the computations are documented in ref {4}. The complete output of program RESTAF which includes the input matrices E,F,G,P,Q,R is given in table I.

The components of the vector v correspond to the following physical quantities:

$$v = \begin{bmatrix} p_d & \text{the drum pressure} \\ y & \text{the drum liquid level} \\ T_w & \text{the drum liquid temperature} \\ T_r & \text{the riser tube temperature} \\ x & \text{the steam quality} \\ w_o & \text{the riser outlet flow} \\ w_w & \text{the downcomer flow} \\ w_e & \text{the evaporation flow} \\ Q_r & \text{heat flow from the risers to the} \\ & \text{steam water mixture} \end{bmatrix}$$

The input variables are:

$$u = \begin{bmatrix} Q_g & \text{heat flow to risers} \\ w_{fw} & \text{feedwater flow} \\ w_s & \text{steam outlet flow} \end{bmatrix}$$

and the output variables are:

$$y = \begin{bmatrix} p_d & \text{the drum pressure} \\ y & \text{the drum liquid level} \end{bmatrix}$$

On the first two pages of table I the input matrices E,F,G,P,Q,R are listed. Inspecting the matrix E we find that the two last columns of E equal zero. All other columns have non zero elements. The column vectors six and seven of E have non zero elements only in the second row and consequently they are linear dependent. The rank of E maximally equals six and we have the interesting situation when the number of time derivatives of the variables v_i exceeds the rank of E. The time derivatives of the riser outlet flow and the downcomer flow which correspond to the non zero elements in columns six and seven arise from momentum equations for the riser and the downcomer.

Both conditions of theorem 1 are satisfied in this case and the reduction is successful. The computed rank of E equals six. In section 5 we found that in general the non uniqueness of the rank factorization could not be exploited for a favorable choice of the state variables. However, if the linear coupling between the columns of E was not too elaborate we could choose K as the linear independent columns of E and get a quite simple form of L. In the program this is done using an indicating vector (JBETA) which has non zero elements in positions corresponding to the linear independent columns of E. In this case K will equal the first six columns of E. Using the pseudo inverse of K the matrix L is computed to satisfy the equation E = KL. The state variables are given by the matrix L viz.

$$\begin{aligned} x_1 &= v_1 \\ x_2 &= v_2 \\ x_3 &= v_3 \\ x_4 &= v_4 \\ x_5 &= v_5 \\ x_6 &= v_6 + 2.656\ v_7 \end{aligned}$$

and we have a simple physical interpretation of the state variables.

Equation (3.7) gives all static relations between the variables v and u. A successful reduction requires $\rho((I-KK^+)F) = \ell-n$. This rank equals 3. The three static relations are given by the matrices $(I - KK^+)F$ and $(I - KK^+)G$ in table I. The original linearized equation contains two static relations between the variables v and u which are found in the fourth and ninth rows of F and G. These relations are refound in the fourth and ninth rows of $(I-KK^+)F$ and $(I-KK^+)G$. The third static relation is given by any of the rows 1,3,7,8 of $(I-KK^+)F$ and $(I-KK^+)G$. These four rows are in pairs linearly dependent. The existence of the third static relation is a consequence of the fact that the number of time derivatives of the variables v_i exceeds the rank of E and primarily of the assumptions made when basic physical laws were applied to the process.

Inspecting the system matrices A,B,C and D we find that D equals zero and that

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Several elements of A and B also approximately equal zero. However, some caution must be observed when approximating since the variables have true physical dimensions.

If the reduction is made manually it is indeed a very tedious work. It is therefore believed that this algorithm represents a very attractive solution to the reduction problem.

8. ILLUSTRATION OF THE IMPLICATIONS OF THE CONDITIONS OF THEOREM 1

Using examples suitable for hand computation we will demonstrate the physical implications when the reduction fails. As far as the author knows there do not exist any physical counterparts of the examples.

Example 1. We will demonstrate when the system fails to satisfy condition (i). Consider the system

$$\begin{bmatrix} 0 & 1 & 2 & 0 \\ 1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \dot{v} + \begin{bmatrix} 0 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \\ 3 & 3 & 3 & 3 \\ 0 & 2 & 1 & 1 \end{bmatrix} v + \begin{bmatrix} 0 & -1 \\ -1 & 0 \\ -1 & -1 \\ 1 & 0 \end{bmatrix} u = 0$$

We get

$$(I-KK^{+})F = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 1 \end{bmatrix}$$

thus

$$\rho((I-KK^{+})F) = 1$$

This rank should equal 2 since the rank of E equals 2 and ℓ equals 4. If we examine the original system it is clear that the second and third equations are equal except for the input variables. This implies that

$$2u_1 = u_2$$

The system has only one independent input variable and only one of the two considered equations contributes to the number of system equations. Hence a unique solution does not exist.

Example 2. The system in this example will satisfy the condition (i) but not condition (ii). We have

$$\begin{bmatrix} 0 & 1 & 2 & 0 \\ 1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \dot{v} + \begin{bmatrix} 0 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \\ 4 & 2 & 2 & 0 \\ 1 & 1 & 2 & 0 \end{bmatrix} v + \begin{bmatrix} 0 & -1 \\ -1 & 0 \\ -1 & -1 \\ 1 & 0 \end{bmatrix} u = 0 \qquad (8.1)$$

We get

$$(I-KK^+)F = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -0.3 & 0.3 & 0.3 & 0.9 \\ 0.1 & -0.1 & -0.1 & -0.3 \\ 1 & 1 & 2 & 0 \end{bmatrix}$$

and

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 \end{bmatrix}$$

Then

$$\rho((I-KK^+)F) = 2$$

$$\rho\begin{pmatrix} L \\ (I-KK^+)F \end{pmatrix} = 3$$

The existence of a unique solution requires that the rank of condition (ii) equals 4. If we solve the fourth equation of equation (8.1) for v_3 and substitute the result into the first equation the time derivative of v_2 will vanish. Eliminate v_3 using the remaining two equations. Then

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 3 & 0 & 0 \end{bmatrix} \dot{z} + \begin{bmatrix} 1/2 & -1/2 & 1 \\ 1/2 & 1/2 & 1 \\ 3 & 1 & 0 \end{bmatrix} z + \begin{bmatrix} 1/2 & 1 \\ -3/2 & 0 \\ -2 & 1 \end{bmatrix} u + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \dot{u} = 0$$

where $z = [v_1 \quad v_2 \quad v_4]^T$. This system can be reduced to a first order system on standard form. The number of state variables is thus less than the rank of E, which in this case causes the failure.

9. ACKNOWLEDGEMENT

The author wish to express his gratitude to Mr. K. Mårtensson who has written the subroutines MIART and DMATPROD which are used in the reduction program.

10. REFERENCES

1. Brown K.M.: A quadratically convergent method for solving simultaneous non-linear equations. Doctoral Thesis, Dept. Computer Sciences, Purdue U., Lafayette, Ind., Aug. 1966. The algorithm is found in CACM. Algo 316.

2. Broyden C.G.: A Class of Methods for Solving Nonlinear Simultaneous Equations. Math. Comput. 19 (1965), 577-593.

3. Deutsch R.: Estimation theory. Prentice Hall, Inc. N.Y. 1965.

4. Eklund K.: Linear Mathematical Models of the Drum-Downcomer-Riser Loop of a Drum Boiler, Report 6809 November 1968, Lund Institute of Technology,Division of Automatic Control

5. Koenig H.E., Tokad Y., Kesavan H.K.: Analysis of discrete Physical Systems. McGraw-Hill N.Y. 1967.

6. Matweer V.A.: Methods for the approximate solution of a system of nonlinear equations. USSR Comp. Maths. and Math. Physics., vol 4, p. 1, 1964.

7. Saaty T.L., Brown J.: Nonlinear Mathematics. McGraw-Hill N.Y. 1964.

8. Åström K.J.: Reglerteori. Almqvist & Wiksell/Gebers Förlag AB, Stockholm 1968.

APPENDIX A

Every matrix $A^{\{mxn\}}$ of rank $r > 0$ has a rank factorization of the form

$$A^{\{mxn\}} = B^{\{mxr\}} \cdot C^{\{rxn\}} \qquad (1)$$

where both B and C have rank r. The pair of matrices B and C are not unique. The requirements are that the columns of B form a basis for the column space generated by the columns of A. C is chosen to satisfy eq (1). The pseudoinverse is defined as

$$A^+ = C^T(CC^T)^{-1}(B^TB)^{-1}B^T \qquad (2)$$

Some of the properties are

1. For any matrix A there always exists a unique pseudo-inverse.
2. If A is quadratic and nonsingular $A^+ = A^{-1}$
3. $(A^+)^+ = A$
4. $AA^+A = A$
5. $(AA^+)^2 = AA^+$ and $(A^+A)^2 = A^+A$
6. AA^+ and A^+A are symmetric matrices.

We also have

$$B^+ = (B^TB)^{-1}B^T$$
$$C^+ = C^T(CC^T)^{-1}$$

where B and C are the same matrices as in eq (1).

Let K be a ℓ x r matrix where r is the rank of K and $\ell \geq r$. Then

a. The rank of the matrix KK^+ is r.

<u>Proof</u>: The rank of the matrices K and K^+ equals r. We have

$$(KK^+)K = K$$

then

$$\rho(K) = r \leq \min(\rho(KK^+), \rho(K))$$

and

$$\rho(KK^+) \leq \min(\rho(K), \rho(K^+))$$

thus

$$\rho(KK^+) = r$$

b. The rank of the matrix $(I-KK^{+})$ is $\ell - r$.

Proof: We have

$$\rho((I-KK^{+}) + KK^{+}) \leq \rho(I-KK^{+}) + \rho(KK^{+})$$

or

$$\rho(I-KK^{+}) \geq \rho(I) - \rho(KK^{+}) = \ell - r$$

and

$$(I-KK^{+})K = 0$$

The last equation implies that the dimension of the range space of $(I-KK^{+})$ is less or equal to $\ell - r$. Thus

$$\rho(I-KK^{+}) = \ell - r$$

APPENDIX B

```
      PROGRAM RESTAF
C
C     COMPUTES THE STANDARD FORM S(A,B,C,D) OF A LINEAR
C     EQUATION
C            E*VDOT+F*V+G*U=0
C               P*Y+Q*V+R*U=0
C     USING THE CONCEPT OF PSEUDOINVERSES.
C     THE STATE VARIABLES ARE
C                 X=L*V
C
C     NE-NE IS THE ORDER OF THE QUADRATIC MATRICES E AND F
C     NU-NU IS THE NUMBER OF INPUT VARIABLES U
C     NY-NY IS THE NUMBER OF OUTPUT VARIABLES Y
C     DINV-DINV IS A SMALL NUMBER USED WHEN INVERTING
C     DRANK
C     DRANK-DRANK IS A NUMBER USED WHEN FINDING RANK
C     THE NUMBER OF INPUT,OUTPUT AND STATE VARIABLES ARE
C     LIMITED TO 10
C     THE NUMBER OF EQUATIONS (NE) IS LIMITED TO 30
C
C     REFERENCE K EKLUND,NUMERICAL MODELBUILDING
C
C     SUBROUTINES REQUIRED
C     MIART,DMATPROD
C
      DIMENSION E(30,30),F(30,30),G(30,10),P(10,10),Q(10,30),R(10,10),AM
     1(30,30),BM(30,30),CM(30,30),DM(30,30),EM(30,30),AK(30,30),IBETA(30
     1),JBETA(30),ICOUNT(40)
  100 READ 1,NR
    1 FORMAT(I3)
      IF(NR-99) 2,2,121
    2 READ 3,NE,NU,NY,DINV,DRANK
    3 FORMAT(3I2,2E20.10)
      READ 4,((E(I,J),J=1,NE),I=1,NE)
      READ 4,((F(I,J),J=1,NE),I=1,NE)
      READ 4,((G(I,J),J=1,NU),I=1,NE)
      READ 4,((P(I,J),J=1,NY),I=1,NY)
      READ 4,((Q(I,J),J=1,NE),I=1,NY)
      READ 4,((R(I,J),J=1,NU),I=1,NY)
    4 FORMAT(4E20.10)
      PRINT 5
    5 FORMAT(1H1,30H PRINTOUTS FROM PROGRAM RESTAF,//)
      PRINT 6
    6 FORMAT(9H MATRIX E,/)
      DO 130 I=1,NE
  130 PRINT 10,(E(I,J),J=1,NE)
      PRINT 11
   11 FORMAT(//,9H MATRIX F,/)
      DO 131 I=1,NE
  131 PRINT 10,(F(I,J),J=1,NE)
      PRINT 12
   12 FORMAT(//,9H MATRIX G,/)
      DO 132 I=1,NE
  132 PRINT 10,(G(I,J),J=1,NU)
      PRINT 13
   13 FORMAT(//,9H MATRIX P,/)
```

```
      DO 133 I=1,NY
  133 PRINT 10,(P(I,J),J=1,NY)
      PRINT 14
   14 FORMAT(//,9H MATRIX Q,/)
      DO 134 I=1,NY
  134 PRINT 10,(Q(I,J),J=1,NE)
      PRINT 15
   15 FORMAT(//,9H MATRIX R,/)
      DO 135 I=1,NY
  135 PRINT 10,(R(I,J),J=1,NU)
   10 FORMAT(10E11.3)
C
      DO 16 I=1,NE
      DO 16 J=1,NE
   16 AM(I,J)=E(I,J)
C
      IA=30
      IB=30
      DELTA=DRANK
      INVRT=0
      IRANK=1
      IPS=0
C
      CALL MIART(AM,NE,NE,0,INVRT,IRANK,IPS,DELTA,IA,IB,IBETA,JBETA)
C
      PRINT 17,IRANK
   17 FORMAT(//,22H THE RANK OF MATRIX E=,I2)
      PRINT 18
   18 FORMAT(//,33H THE LINEAR INDEPENDENT ROWS OF E,/)
      DO 19 I=1,NE
      IF(IBETA(I)-1) 20,21,20
   21 ICOUNT(I)=I
      GO TO 19
   20 ICOUNT(I)=0
   19 CONTINUE
      PRINT 22,(ICOUNT(I),I=1,NE)
   22 FORMAT(40I3)
      PRINT 23
   23 FORMAT(//,37H THE LINEAR INDEPENDENT COLUMNS OF E,/)
      DO 24 I=1,NE
      IF(JBETA(I)) 25,26,25
   25 ICOUNT(I)=I
      GO TO 24
   26 ICOUNT(I)=0
   24 CONTINUE
      PRINT 22,(ICOUNT(I),I=1,NE)
C
C     COMPUTE MATRIX K=LINEAR INDEPENDENT COLUMNS OF E
C
      NI=0
      DO 28 I=1,NE
      IF(JBETA(I)) 29,28,29
   29 NI=NI+1
      DO 30 J=1,NE
   30 AK(J,NI)=E(J,I)
   28 CONTINUE
```

```
C
      PRINT 31
   31 FORMAT(//,43H THE RESULT OF THE RANK FACTORIZATION E=K*L,/)
      PRINT 32
   32 FORMAT(9H MATRIX K,/)
      DO 136 I=1,NE
  136 PRINT 10,(AK(I,J),J=1,NI)
C
C     COMPUTE THE PSEUDO INVERSE OF K
C
      DO 33 I=1,NE
      DO 33 J=1,NI
   33 AM(J,I)=AK(I,J)
C
      CALL DMATPROD(AM,AK,BM,NI,NE,NI,30,30,30,30,30,30)
C
C
      IA=30
      DELTA=DINV
      INVRT=1
      IPS=0
      IRANK=0
C
      CALL MIART(BM,NI,NI,0,INVRT,IRANK,IPS,DELTA,IA,IB,IBETA,JBETA)
C
      IF(INVRT+1) 36,37,36
   37 PRINT 38,INVRT
   38 FORMAT(//,48H THE INVERSION OF K(TRANSPOSED)*K FAILED  INVRT=,I2)
      GO TO 120
   36 CONTINUE
C
      CALL DMATPROD(BM,AM,CM,NI,NI,NE,30,30,30,30,30,30)
C
C
C     COMPUTE L=K(PSEUDO)*K
C
C
      CALL DMATPROD(CM,E,BM,NI,NE,NE,30,30,30,30,30,30)
C
C
      PRINT 43
   43 FORMAT(//,9H MATRIX L,/)
      DO 205 I=1,NI
  205 PRINT 10,(BM(I,J),J=1,NE)
      PRINT 44
   44 FORMAT(//,23H THE PSEUDOINVERSE OF K,/)
      DO 137 I=1,NI
  137 PRINT 10,(CM(I,J),J=1,NE)
C
C     COMPUTE K(PSEUDO)*K
C
C
      CALL DMATPROD(CM,AK,E,NI,NE,NI,30,30,30,30,30,30)
C
      PRINT 202
  202 FORMAT(//,48H MATRIX K(PSEUDO)*K SHOULD EQUAL THE UNIT MATRIX,/)
```

```
      DO 138 I=1,NI
  138 PRINT 10,(E(I,J),J=1,NI)
C
C     COMPUTE K*K(PSEUDO)
C
C
      CALL DMATPROD(AK,CM,E,NE,NI,NE,30,30,30,30,30,30)
C
C
C     COMPUTE (I-K*K(PSEUDO))
C
      DO 47 I=1,NE
   47 E(I,I)=E(I,I)-1.
      DO 48 I=1,NE
      DO 48 J=1,NE
   48 E(I,J)=-E(I,J)
C
C     COMPUTE (I-K*K(PSEUDO))*F
C
C
      CALL DMATPROD(E,F,AM,NE,NE,NE,30,30,30,30,30,30)
C
      PRINT 151
  151 FORMAT(//,25H MATRIX (I-K*K(PSEUDO))*F,/)
      DO 152 I=1,NE
  152 PRINT 10,(AM(I,J),J=1,NE)
C
C     COMPUTE (I-K*K(PSEUDO))*G
C
      CALL DMATPROD(E,G,DM,NE,NE,NU,30,30,30,10,30,30)
C
      PRINT 153
  153 FORMAT(//,25H MATRIX (I-K*K(PSEUDO))*G,/)
      DO 154 I=1,NE
  154 PRINT 10,(DM(I,J),J=1,NU)
      DO 51 I=1,NE
      DO 51 J=1,NE
   51 AK(I,J)=AM(I,J)
C
C     TEST THE RANK OF (I-K*K(PSEUDO))*F
C
      IA=30
      DELTA=DRANK
      INVRT=0
      IRANK=1
      IPS=0
C
      CALL MIART(AM,NE,NE,0,INVRT,IRANK,IPS,DELTA,IA,IB,IBETA,JBETA)
C
      NTEST=NE-NI
      IF(IRANK-NTEST) 52,53,53
   52 PRINT 54,IRANK,NTEST
   54 FORMAT(//,45H FAILURE THE RANK OF (I-K*K(PSEUDO))*F EQUALS,I3,13H
     1IT SHOULD BE,I3)
      DO 55 I=1,NE
      IF(IBETA(I)-1) 57,56,57
```

```
   56 ICOUNT(I)=I
      GO TO 55
   57 ICOUNT(I)=0
   55 CONTINUE
      PRINT 58
   58 FORMAT(//,49H THE LINEAR INDEPENDENT ROWS OF (I-K*K(PSEUDO))*F,/)
      PRINT 22,(ICOUNT(I),I=1,NE)
      GO TO 120
   53 CONTINUE
C
C     COMPUTE            L
C                  (I-K*K(PSEUDO))*F
C
      DO 59 I=1,NI
      DO 59 J=1,NE
   59 AM(I,J)=BM(I,J)
      DO 60 I=1,NE
      DO 60 J=1,NE
      NM=NI+I
   60 AM(NM,J)=AK(I,J)
C
C     TEST THE RANK OF          L
C                          (I-K*K(PSEUDO))*F
C
      DELTA=DRANK
      INVRT=0
      IRANK=1
      IPS=0
C
      CALL MIART(AM,NM,NE,0,INVRT,IRANK,IPS,DELTA,IA,IB,IBETA,JBETA)
C
      IF(IRANK-NE) 61,62,61
   61 PRINT 63,IRANK,NE
   63 FORMAT(//,47H FAILURE THE RANK OF L,(I-K*K(PSEUDO))*F EQUALS,I3,13
     1H IT SHOULD BE,I3)
      NT=NE+NI
      DO 64 I=1,NT
      IF(IBETA(I)-1)65,66,65
   66 ICOUNT(I)=I
      GO TO 64
   65 ICOUNT(I)=0
   64 CONTINUE
      PRINT 67
   67 FORMAT(//,51H THE LINEAR INDEPENDENT ROWS OF L,(I-K*K(PSEUDO))*F,/
     1)
      PRINT 22,(ICOUNT(I),I=1,NT)
      GO TO 120
   62 CONTINUE
C
C     COMPUTE L(TRANSPOSED)*L
C
      DO 68 I=1,NI
      DO 68 J=1,NE
   68 DM(J,I)=BM(I,J)
C
      CALL DMATPROD(DM,BM,AM,NE,NI,NE,30,30,30,30,30,30)
```

```
C
C
C     COMPUTE L(TRANSPOSED)*L+F(TRANSPOSED)*(I-K*K(PSEUDO))*F
C
      DO 71 I=1,NE
      DO 71 J=1,NE
   71 BM(J,I)=F(I,J)
C
      CALL DMATPROD(BM,AK,EM,NE,NE,NE,30,30,30,30,30,30)
C
      DO 74 I=1,NE
      DO 74 J=1,NE
   74 EM(I,J)=AM(I,J)+EM(I,J)
C
      PRINT 75
   75 FORMAT(//,33H MATRIX LT*L+FT*(I-K*K(PSEUDO))*F,/)
      DO 139 I=1,NE
  139 PRINT 10,(EM(I,J),J=1,NE)
C
C     COMPUTE (L(TRANSPOSED)*L+F(TRANSPOSED)*(I-K*K(PSEUDO))*F)(INVERSE)
C
      IA=30
      IB=30
      DELTA=DINV
      INVRT=1
      IRANK=0
      IPS=0
C
      CALL MIART(EM,NE,NE,0,INVRT,IRANK,IPS,DELTA,IA,IB,IBETA,JBETA)
C
      IF(INVRT+1) 76,77,76
   77 PRINT 78 ,INVRT
   78 FORMAT(//86H FAILURE (L(TRANSPOSED)*L+F(TRANSPOSED)*(I-K*K(PSEUDO)
     1)*F)(INVERSE) IS SINGULAR INVRT=,I2)
      GO TO 120
   76 CONTINUE
C
C     COMPUTE F(TRANSPOSED)*(I-K*K(PSEUDO))
C
C
      CALL DMATPROD(BM,E,AM,NE,NE,NE,30,30,30,30,30,30)
C
C
C     COMPUTE A
C
C
      CALL DMATPROD(EM,DM,AK,NE,NE,NI,30,30,30,30,30,30)
C
      CALL DMATPROD(F,AK,DM,NE,NE,NI,30,30,30,30,30,30)
C
      CALL DMATPROD(CM,DM,E,NI,NE,NI,30,30,30,30,30,30)
      DO 87 I=1,NI
      DO 87 J=1,NI
   87 E(I,J)=-E(I,J)
C
C     COMPUTE P(INVERSE)
```

```
C
      IA=10
      IB=10
      DELTA=DINV
      INVRT=1
      IRANK=0
      IPS=0
C
      CALL MIART(P,NY,NY,0,INVRT,IRANK,IPS,DELTA,IA,IB,IBETA,JBETA)
C
      IF(INVRT+1) 88,89,88
   89 PRINT 90,INVRT
   90 FORMAT(//,29H FAILURE P IS SINGULAR INVRT=,I2)
      GO TO 120
   88 CONTINUE
C
      PRINT 91
   91 FORMAT(//,18H MATRIX P(INVERSE),/)
      DO 140 I=1,NY
  140 PRINT 10,(P(I,J),J=1,NY)
C
C     COMPUTE P(INVERSE)*Q
C
      CALL DMATPROD(P,Q,DM,NY,NY,NE,10,10,10,30,30,30)
C
C     COMPUTE C
C
      CALL DMATPROD(DM,AK,BM,NY,NE,NI,30,30,30,30,30,30)
C
      DO 96 I=1,NY
      DO 96 J=1,NI
   96 BM(I,J)=-BM(I,J)
C
C     COMPUTE B
C
      CALL DMATPROD(EM,AM,AK,NE,NE,NE,30,30,30,30,30,30)
C
      CALL DMATPROD(F,AK,EM,NE,NE,NE,30,30,30,30,30,30)
C
      DO 102 I=1,NE
  102 EM(I,I)=EM(I,I)-1.
C
      CALL DMATPROD(CM,EM,AM,NI,NE,NE,30,30,30,30,30,30)
C
      CALL DMATPROD(AM,G,CM,NI,NE,NU,30,30,30,10,30,30)
C
C     COMPUTE D
C
      CALL DMATPROD(AK,G,AM,NE,NE,NU,30,30,30,10,30,30)
C
      CALL DMATPROD(DM,AM,AK,NY,NE,NU,30,30,30,30,30,30)
C
      CALL DMATPROD(P,R,AM,NY,NY,NU,10,10,10,10,30,30)
C
      DO 113 I=1,NY
      DO 113 J=1,NU
```

```
  113 DM(I,J)=AK(I,J)-AM(I,J)
C
C     PRINT THE RESULT
C
      PRINT 114
  114 FORMAT(//,16H SYSTEM MARTICES,//)
      PRINT 115
  115 FORMAT(9H MATRIX A,/)
      DO 141 I=1,NI
  141 PRINT 10,(E(I,J),J=1,NI)
      PRINT 116
  116 FORMAT(//,9H MATRIX B,/)
      DO 142 I=1,NI
  142 PRINT 10,(CM(I,J),J=1,NU)
      PRINT 117
  117 FORMAT(//,9H MATRIX C,/)
      DO 143 I=1,NY
  143 PRINT 10,(BM(I,J),J=1,NI)
      PRINT 118
  118 FORMAT(//,9H MATRIX D,/)
      DO 144 I=1,NY
  144 PRINT 10,(DM(I,J),J=1,NU)
  120 CONTINUE
      GO TO 100
  121 CALL EXIT
      END
```

NUMERICAL MODELBUILDING [†]

K. Eklund

ABSTRACT

This paper describes a systematic modelling technique which makes it possible to avoid many of the tedious elements that always are involved in the modelling of industrial processes. The technique is based on the nonlinear equations which are usually obtained from basic physical laws. It gives a systematic procedure to compute steady-state solutions and linearized equations from the nonlinear physical equations. The procedure is based on wellknown methods for solving nonlinear equations and numerical differentiation. The final result is a system description on state space form S(A,B,C,D). The essential difficulty is to find the smallest number of state variables and to assign these to the linearized model. A method for solving this problem is the main result of the paper. The paper also contains an application of this method to the modelling of a boiler.

[†] This work has been supported by the Swedish Board for Technical Development under Contract 68-336-f.

TABLE OF CONTENTS

1. INTRODUCTION

It is a very tedious work to establish a linearized mathematical model of an industrial process of some complexity. Even if basic physical laws are applicable it can be extremely laborious to compute the steady state values and to linearize the equations. In particular if we take into account that is is often highly desirable to develop linearized models for different operating conditions and to investigate the sensitivity of the model to physical parameters. It is also very difficult to check the linearized equations even if the basic equations themselves often are quite easy to check.

It thus seems rather attractive to develop an algorithm which enables a digital computer to perform all the tedious work. Such an algorithm is proposed in this paper. We start with the basic nonlinear equations and proceed to compute steady state values and linearize. The final result is a system description on state space form. The state space form is convenient because a large amount of control theory uses this representation of the system as a starting point and since important physical variables can be retained as state variables in the final model. The procedure is based on wellknown methods for solving nonlinear equations and numerical differentiation. The only difficulty is to assign the smallest number of state variables to the linearized model. A method for solving this problem is the main result of the paper.

The procedure is outlined in section 2. The proof of the theorem given in section 4 requires some knowledge of the concept of pseudo inverses. In section 3 we give some relevant properties of the pseudo inverse. A method to find the smallest number of state variables and to assign these to the linearized model is presented in section 4. The main result Theorem 1 is contained in this section. In section 5 we give an alternative formulation of some conditions of Theorem 1. The assignment of state variables is not unique. In section 6 it is discussed how this non-uniqueness can be exploited to choose state variables that are physically significant. In section 7 the reduction technique is used for the modelling of a drum boiler.

2. A SYSTEMATIC MODEL REDUCTION TECHNIQUE

To establish a mathematical model of an industrial process it is often convenient to divide the process into a number of components. These components are treated separately. A set of equations which describe the dynamic and static relations between the inputs and outputs for each component are derived. The components are linked up with a number of internal variables which might be of secondary interest. This technique simplifies the derivation of the basic equations but introduces a number of auxiliary variables, see e.g. Analysis of Discrete Physical Systems by Koenig et al {5}. The resulting mathematical model is usually a set of nonlinear equations which include both ordinary and partial differential equations. Partial differential equations are approximated by finite differences in the space co-ordinates. We also assume that the system has constant coefficients. The system behaviour for small disturbances about an equilibrium state is often of great interest. This behaviour might be described by the linearized system equations. Thus if we require the resulting model to be a set of linear ordinary differential equations the following systematic approach is proposed.

- The process is described by basic physical laws such as the laws of conservation of mass, energy and momentum. The resulting set of equation is

  $f(\dot{v},v,u) = 0$ (2.1a)

  $g(y,v,u) = 0$ (2.1b)

  where f is an ℓ-vector whose components are nonlinear functions of the variables v_i, their time derivatives and the process input variables u_i. g is a k-vector whose components are nonlinear functions of the variables v_i, u_i and the process output variables y_i. The input vector u and the output vector y are identified and treated separately already at this stage. All other variables are included in v. The set of equations is consistent if the number of variables v_i equals ℓ and if y is a k-vector. We can always assume that the vector f does not depend on du/dt because we can then introduce a new input variable $u^{*} = du/dt$.

- The steady state values are obtained if we put time derivatives equal to zero in equation (2.1) viz.

$$f(v,u) = 0 \qquad (2.2a)$$

$$g(y,v,u) = 0 \qquad (2.2b)$$

Given the steady state values u^o of u equations (2.2a), (2.2b) determine the steady state values v^o and y^o of v and y respectively. The zero solutions of the nonlinear equation (2.2) are obtained by standard techniques e.g. a Newton-Raphson method. Other methods are found in {1},{2},{6},{7}.

- Linearize equation (2.1). We get

$$E\dot{v} + Fv + Gu = 0 \qquad (2.3a)$$

$$Py + Qv + Ru = 0 \qquad (2.3b)$$

where

$E = f_{\dot{v}}(0,v^o,u^o)$ $\{\ell \times \ell\}$

$F = f_v(0,v^o,u^o)$ $\{\ell \times \ell\}$

$G = f_u(0,v^o,u^o)$ $\{\ell \times \ell\}$

$P = g_y(y^o,v^o,u^o)$ $\{k \times k\}$

$Q = g_v(y^o,v^o,u^o)$ $\{k \times \ell\}$

$R = g_u(y^o,v^o,u^o)$ $\{k \times m\}$

The perturbed variables that is the differences $v-v^o$, $y-y^o$ and $u-u^o$ are denoted as the variables themselves. The matrices of first partial derivatives of the vectors f and g with respect to $\dot{v}$, v and u and y, v and u respectively are obtained by numerical differences.

- The linearized set of equations (2.3) is reduced to state space form S(A,B,C,D).

$$\begin{cases} \dfrac{dx}{dt} = Ax + Bu \\ y = Cx + Du \end{cases} \qquad (2.4)$$

where x is the state n-vector, u is the input m-vector and y is the output k-vector. A,B,C and D are matrices of proper order.

  The problem of reducing equation (2.3) to state space form is to find the smallest possible number of state variables and to assign these to the model. A method which solves this problem is the main result of the paper and is presented in the following section.

- If the transfer function representations are needed these are easily obtained from equation (2.4) using standard methods, see e.g. {8}.

3. PSEUDO INVERSE

The proof of the reduction theorem given in the following section requires some knowledge of the concept of pseudo inverses. Therefore we give some relevant properties in this section. A detailed presentation is found in e.g. Estimation Theory by Deutch {3}.

Every matrix $A^{\{m \times r\}}$ of rank $n > 0$ has a rank factorization of the form

$$A^{\{m \times r\}} = B^{\{m \times n\}} C^{\{n \times r\}} \tag{3.1}$$

where both B and C have rank n. The pair of matrices B and C are not unique. The requirements are that the columns of B form a basis for the column space generated by the columns of A. C is then chosen to satisfy (3.1). The pseudo inverse of A is now defined as

$$A^+ = C^T(CC^T)^{-1} (B^TB)^{-1} B^T \tag{3.2}$$

Some of the properties are:

1. For any matrix A there always exists a unique pseudo inverse.
2. The rank of A^+ equals the rank of A.
3. If A is quadratic and nonsingular $A^+ = A^{-1}$
4. $(A^+)^+ = A$
5. $AA^+A = A$ and $A^+AA^+ = A^+$
6. $(AA^+)^2 = AA^+$ and $(A^+A)^2 = A^+A$. That is AA^+ and A^+A are projections.
7. AA^+ and A^+A are symmetric matrices.

Further we have

$$B^+ = (B^TB)^{-1} B^T$$
$$C^+ = C^T(CC^T)^{-1}$$

where B and C are the same matrices as in (3.1).

We will also use the following rank properties of the pseudo inverse. Let T be a $m \times n$ matrix where n is the rank of T and $m \geqslant n$.

Lemma 1. The rank of the matrix TT^+ is n.

Proof: The rank of T and T^+ equals n. We have according to 5 above

$(TT^+)T = T$

Then

$\rho(T) = n \leq \min(\rho(TT^+), \rho(T))$

where $\rho(T)$ denotes the rank of T.

We also have

$\rho(TT^+) \leq \min(\rho(T), \rho(T^+))$

Thus

$\rho(TT^+) = n$

Lemma 2. The rank of the matrix $(I-TT^+)$ is m-n

Proof: We have

$\rho((I-TT^+) + TT^+) \leq \rho(I-TT^+) + \rho(TT^+)$

or

$\rho(I-TT^+) \geq \rho(I) - \rho(TT^+) = m-n$

and

$(I-TT^+)T = 0$

The last equation implies that the dimensions of the range space of $(I-TT^+)$ is less than or equal to m-n. Thus

$\rho(I-TT^+) = m-n$

Introduce the notations $\mathcal{R}(T)$ and $\mathcal{N}(T)$ which stand for the range space and the null space of the matrix T respectively. Then we also have

Lemma 3. $\mathcal{R}(T) = \mathcal{N}(I-TT^+)$

Proof: We have

$(I-TT^+)T = T - TT^+T = 0$

Hence

$$t_i \in \mathcal{N}(I-TT^+) \qquad i = 1,\ldots,n \qquad (3.3)$$

where t_i is the i:th column vector of T. According to Lemma 2 we have

$$\rho(I-TT^+) = m-n$$

The dimension of the null space of $(I-TT^+)$ is then n. The vectors t_i, $i = 1,\ldots,n$ are linearly independent and span the range space of T. Thus

$$\mathcal{R}(T) = \mathcal{N}(I-TT^+)$$

4. REDUCTION OF LINEARIZED SYSTEM EQUATIONS TO STATE SPACE FORM

Before the theorem is stated it is convenient to have a look at the structure of equation (2.3). It is immediately clear that the number of state variables at most equals the rank of E. If the rank of E equals ℓ the number of state variables is ℓ and equation (2.3a) can be solved directly using the inverse of E. The reduction to standard form is in this case trivial. However, in general the rank of E is less than ℓ and greater than zero. This is a consequence of two facts. Static relations between the variables v_i and u_i create rows of zeros in E. The number of time derivatives introduced might exceed the rank of E. That is we have additional static relations between the variables v_i and u_i which are not quite apparent. In the boiler application, section 7, the matrices of equation (2.3) are given in Table 1. Inspecting matrix E we find that E has dimension 9 x 9 and that the rank maximally equals six.

If the inverse of P does not exist one or several of the outputs are a linear combination of the others. The linear dependent outputs may be excluded by computation of the linear independent rows of P. However, physical insight usually permits us to avoid this problem and there is no loss of generality if we assume that the inverse of P exists.

<u>Theorem 1</u>

Given a linear dynamic system with constant coefficients described by

$$E\dot{v}(t) + Fv(t) + Gu(t) = 0 \tag{4.1a}$$

$$Py(t) + Qv(t) + Ru(t) = 0 \tag{4.1b}$$

where $v(t)$ is an ℓ-vector, $u(t)$ is an m-vector, $y(t)$ is a k-vector and E,F,G,P,Q,R are matrices of proper order. Assume that the rank of E is n, $0 < n \leq \ell$. The rank factorization of E then is

$$E^{\{\ell \times \ell\}} = K^{\{\ell \times n\}} L^{\{n \times \ell\}} \tag{4.2}$$

where the matrices K and L both have rank n. If

(i) $\rho((I-KK^{+})F) = \ell - n$

(ii) $\mathcal{R}((I-KK^{+})F) \cap \mathcal{R}(L) = 0$

then the state space form of the linear system (4.1) is

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t) \qquad (4.3)$$

where the state n-vector is

$$x(t) = Lv(t)$$

and

$$A = -K^{+}F\{L^{T}L + F^{T}(I-KK^{+})F\}^{-1}L^{T}$$
$$B = K^{+}\{F\{L^{T}L + F^{T}(I-KK^{+})F\}^{-1}F^{T}(I-KK^{+}) - I\}G$$
$$C = -P^{-1}Q\{L^{T}L + F^{T}(I-KK^{+})F\}^{-1}L^{T}$$
$$D = P^{-1}\{Q\{L^{T}L + F^{T}(I-KK^{+})F\}^{-1}F^{T}(I-KK^{+})G-R\}$$

Proof:

The problem we have stated can be formulated as: Prove that there exists a unique solution to the differential equation (4.1) and that this solution is given by equation (4.3).

Before the arguing is started we will write down, quite formally, some equations. Combining (4.1a) and (4.2) we get

$$KL\dot{v} + Fv + Gu = 0 \qquad (4.4)$$

Introduce the n-vector x and set

$$x = Lv \qquad (4.5)$$

The equations (4.4) and (4.5) then give

$$K\dot{x} = -(Fv + Gu) \qquad (4.6)$$

To prove the existence of a solution to (4.6) we must show that the vector (Fv + Gu) lies in the column space of K or equivalently that

$$(Fv + Gu) \in \mathcal{R}(K) \qquad (4.7)$$

Using Lemma 3 equation (4.7) can be replaced with

$$(Fv + Gu) \in \mathcal{N}(I-KK^{+}) \qquad (4.8)$$

where

$$K^{+} = (K^{T}K)^{-1} K^{T}$$

By definition equation (4.8) is true if

$$(I-KK^+)(Fv + Gu) = 0 \tag{4.9}$$

or

$$(I-KK^+)Fv = -(I-KK^+)Gu \tag{4.10}$$

Thus if (4.10) holds the solution of (4.6) is

$$\dot{x} = -K^+Fv - K^+Gu \tag{4.11}$$

Rewriting (4.5) and (4.10) as one equation we get

$$\begin{bmatrix} L \\ (I-KK^+)F \end{bmatrix} v = \begin{bmatrix} x \\ -(I-KK^+)Gu \end{bmatrix} \tag{4.12}$$

or

$$Tv = z \tag{4.13}$$

To prove the existence of a unique solution to (4.12) we must show that the vector z lies in the column space of T and that the rank of T equals ℓ. The vector x by definition lies in the column space of L. If condition (i) holds then

$$\mathcal{R}(I-KK^+) = \mathcal{R}((I-KK^+)F) \tag{4.14}$$

Hence

$$\mathcal{R}((I-KK^+)G) \subset \mathcal{R}((I-KK^+)F) \tag{4.15}$$

which implies that the vector $(I-KK^+)Gu$ belongs to the range space of $(I-KK^+)F$. Then the vector z lies in the column space of T and a solution of (4.12) exists. If condition (ii) holds then the rank of T equals ℓ and the solution is unique.

The proof of the theorem can now be carried out in the following manner. Take an arbitrary vector $x(t_1)$ and input vector $u(t_1)$. Then using (4.15) we can always find a vector $v(t_1)$ such that (4.10) is satisfied. This vector $v(t_1)$ is uniquely determined by (4.12) if also condition (ii) of the theorem holds. The solution of (4.12) is obtained using the pseudo inverse of T. Also using the fact that $(I-KK^+)$ is a symmetric projection we get

$$v(t_1) = \tilde{A}x(t_1) + \tilde{B}u(t_1) \tag{4.16}$$

where

$$\tilde{A} = \{L^TL + F^T(I-KK^+)F\}^{-1}L^T$$
$$\tilde{B} = \{L^TL + F^T(I-KK^+)F\}^{-1}F^T(I-KK^+)G$$

Now equation (4.11) uniquely determines the derivative of $x(t_1)$. Using (4.16) we get

$$\dot{x}(t_1) = -K^+F \tilde{A}x(t_1) - (K^+F \tilde{B} + K^+G)\ u(t_1) \tag{4.17}$$

Certain linear combinations of the derivatives of the variables $v_i(t_1)$ are uniquely determined by (4.5). Differentiating (4.5) we get

$$L\dot{v}(t_1) = \dot{x}(t_1) \tag{4.18}$$

Since there are some static relations, given by (4.10), between the variables $v_i(t_1)$ and $u_i(t_1)$ the derivative of $v(t_1)$ is uniquely determined only if we also have the derivative of the input vector $u(t_1)$. This also follows if (4.16) is differentiated.

Thus given arbitrary vectors $x(t_1)$ and $u(t_1)$ we have shown the existence of a unique vector $v(t_1)$ and n unique linear combinations of $\dot{v}_i(t_1)$ which satisfy the original differential equation (4.1a). The last statement follows if (4.18) is substituted into (4.6).

The solution of (4.1a) in the interval (t_1,t_2) for an arbitrary input vector $u(t)$ is obtained by integrating (4.17) with the initial value $x(t_1)$. Using (4.16) and (4.17) and the arguing above this solution will satisfy the original equation (4.1a) in the whole interval (t_1,t_2).

When (4.1b), (4.16) and (4.17) are combined the state space form of the original set of equations is obtained and the theorem is proved.

Notice that the theorem does only supply a solution to the reduction problem when the number of state variables equals the rank of E. Notice also that all static relations between the variables v_i and u_i are given by equation (4.10).

5. CHECK OF CONDITIONS

An algorithm which performs the rank factorization of the matrix E is a necessary subroutine of the reduction program. It is then convenient to formulate the conditions as rank conditions. Condition (i) is already on a suitable form. If condition (i) holds condition (ii) is equivalent to that the rank of T equals ℓ. Hence

(i) $\rho((I-KK^+)F) = \ell - n$

(ii) $\rho(T) = \ell$

where the matrix T is defined by equation (4.13). The conditions should be checked in the listed order.

6. CHOICE OF STATE VARIABLES

The state variables are given by

$$x = Lv$$

The matrix L is not uniquely determined but L is chosen to satisfy

$$E = KL$$

where the column vectors of K form a basis for the vector space generated by the column vectors of E. In this section we will investigate if this non-uniqueness of L could be exploited to get a simple physical interpretation of the state variables.

The variables v_i do often have a simple physical interpretation. It seems then attractive to retain these variables as state variables. Let E be arranged so that the n first columns of E are linearly independent. Assume that s is the largest number of variables v_i which can be retained as state variables viz.

$$\begin{aligned} x_1 &= v_1 \\ x_2 &= v_2 \\ &\vdots \\ x_s &= v_s \qquad s \leq n \end{aligned}$$

The matrix L can then be partitioned as

$$L = \left[\begin{array}{c|c} L_{11} & L_{12} \\ \hline L_{21} & L_{22} \end{array}\right]$$

where

$$\begin{aligned} L_{11} &= I \qquad \{s \times s\} \\ L_{12} &= 0 \qquad \{s \times (\ell-s)\} \\ L_{21} & \qquad \{(n-s) \times s\} \\ L_{22} & \qquad \{(n-s) \times (\ell-s)\} \end{aligned}$$

However, in general the number s will equal zero. To show this let

$$E = [e_1 \ \ e_2 \ \ldots \ e_n \ \ e_{n+1} \ \ldots \ e_\ell]$$

where e_i is the i:th column vector of E. The vectors $e_1, \dots, e_n$ are linearly independent and $e_{n+1}, \dots, e_\ell$ are linearly dependent. Assume

$$e_{n+i} = \sum_{j=1}^{n} \lambda_{ji} e_j \qquad i = 1,2,\dots,\ell-n \tag{6.1}$$

where

$$\lambda_{ji} \neq 0 \qquad \forall\ i,j$$

and that no directions of vectors $e_1, \dots, e_\ell$ coincide. Let $k_1, \dots, k_n$ be an arbitrary basis for the column space generated by $e_1, \dots, e_n$. Then the rank factorization of E may be written as

$$[e_1 \dots e_n \; e_{n+1} \dots e_\ell] = [k_1 \dots k_n] \left[\begin{array}{c|c} I & 0 \\ \hline L_{21} & L_{22} \end{array}\right] = \left[\begin{array}{c|c} K_{11} & K_{12} \\ \hline K_{21} & K_{22} \end{array}\right] \left[\begin{array}{c|c} I & 0 \\ \hline L_{21} & L_{22} \end{array}\right] \tag{6.2}$$

where the submatrices of the partitioned matrix K have the dimensions

K_{11} $\{s \times s\}$
K_{12} $\{s \times (n-s)\}$
K_{21} $\{(\ell-s) \times s\}$
K_{22} $\{(\ell-s) \times (n-s)\}$

Evaluating equation (6.2) we get

$$[e_1 \dots e_s \;\; e_{s+1} \dots e_n \; e_{n+1} \dots e_\ell] = \left[\begin{array}{c|c} K_{11} + K_{12}L_{21} & K_{12}L_{22} \\ K_{21} + K_{22}L_{21} & K_{22}L_{22} \end{array}\right]$$

The identity above implies

$$[e_{s+1} \dots e_n \; e_{n+1} \dots e_\ell] = [k_{s+1} \dots k_n] L_{22} \tag{6.3}$$

There are still n linearly independent column vectors in the left hand matrix in (6.3) since we assumed that $\lambda_{ji} \neq 0$, $\forall$ i,j and that $e_1, \dots, e_\ell$ were in pairs linearly independent. The right hand matrix has maximally n-s linearly independent column vectors. Equation (6.3) thus gives s = 0. This means that there is in general no especially favorable choice of the state variables available.

A natural way to constitute a basis for the column space of E is to choose the linearly independent columns of E as a basis. The rank factorization of E then is

$$E = K[L_1 \mid L_2] \tag{6.4}$$

where

$K = [e_1 \ \dots \ e_n]$
$L_1 = I \qquad \{n \times n\}$
$L_2 \qquad \{n \times (\ell-n)\}$

The submatrix L_2 gives the coefficients of the expressions which constitute the ℓ-n linearly dependent column vectors $e_{n+1}, \dots, e_\ell$ of E. Using the notions of (6.1) we have

$$(\ell_2)_i = [\lambda_{1i} \ \dots \ \lambda_{ni}]^T \qquad i = 1,2,\dots,\ell-n$$

where $(\ell_2)_i$ denotes the i:th column vector of L_2. If the linear dependence of the vectors $e_{n+1}, \dots, e_\ell$ is not as elaborate as indicated in (6.1) but e.g. only one of the vectors $e_1, \dots, e_n$, say e_1, is needed to establish $e_{n+1}, \dots, e_\ell$ then the choice (6.4) creates rows of zeros in the submatrix L_2 except in the row corresponding to e_1. Considering that we have no especially favorable choice of state variables available it seems attractive to use the choice of (6.4).

7. APPLICATION TO A BOILER MODEL

In this section we will give an application of the reduction procedure to a practical problem. We will consider a drum-downcomer-riser loop of a drum boiler for a power station unit of approximately 150 MW. The drum pressure is 140 bar and the outlet steam temperature 530 °C. The derivation of the basic nonlinear equations, computation of steady state values and linearization result in the matrices E,F,G,P,Q,R. The derivation of these results as well as a FORTRAN program for the computations are documented in Linear Mathematical Models of the Drum-downcomer-riser Loop of a Drum Boiler by the author {4}. The complete output of the FORTRAN program which computes the matrices of the state space form given the matrices E,F,G,P,Q,R is presented in Table 1. The table also includes some intermediate results. The matrix K^+K which should equal the unit matrix is used to check the accuracy of K^+.

The components of the vector v correspond to the following physical quantities:

$$v = \begin{bmatrix} p_d & \text{the drum pressure} \\ y & \text{the drum liquid level} \\ T_w & \text{the drum liquid temperature} \\ T_r & \text{the riser tube temperature} \\ x & \text{the steam quality} \\ w_o & \text{the riser outlet flow} \\ w_w & \text{the downcomer flow} \\ w_e & \text{the evaporation flow} \\ Q_r & \text{heat flow from the risers to the steam water mixture} \end{bmatrix}$$

The input variables are:

$$u = \begin{bmatrix} Q_g & \text{heat flow to risers} \\ w_{fw} & \text{feedwater flow} \\ w_s & \text{steam outlet flow} \end{bmatrix}$$

and the output variables are:

$$y = \begin{bmatrix} p_d & \text{the drum pressure} \\ y & \text{the drum liquid level} \end{bmatrix}$$

Inspecting the matrix E we find that the two last columns of E equal zero. All other columns have non-zero elements. The column vectors six and seven of E have non-zero elements only in the second row and consequently they are linearly dependent. The rank of E maximally equals six and we have the interesting situation when the number of time derivatives of the variables v_i exceeds the rank of E. The time derivatives of the riser outlet flow and the downcomer flow which correspond to the non-zero elements in columns six and seven arise from momentum equations for the riser and the downcomer.

Both conditions of Theorem 1 are satisfied in this case and the reduction is successful. The computed rank of E equals six. In section 6 we found that in general the non-uniqueness of the rank factorization could not be exploited for a favorable choice of the state variables. However, if the linear coupling between the columns of E was not too elaborate we could choose K as the linearly independent columns of E and get a quite simple form of L. This choice is used in the reduction program and in the boiler application K will equal the first six columns of E. Using the pseudo inverse of K the matrix L is computed to satisfy the equation E = KL. The state variables are given by the matrix L. We get

$$x_1 = v_1$$
$$x_2 = v_2$$
$$x_3 = v_3$$
$$x_4 = v_4$$
$$x_5 = v_5$$
$$x_6 = v_6 + 2.656\ v_7$$

and we have a simple physical interpretation of the state variables.

Equation (4.10) gives all static relations between the variables v_i and u_i. A successful reduction requires $\rho((I-KK^+)F) = \ell-n$. This rank equals three and consequently the number of static relations between the variables u_i and v_i is three. These relations are given by the matrices $(I-KK^+)F$ and $(I-KK^+)G$ in Table 1. The original linearized equation contains two apparent static relations between the variables v_i and u_i which are found in the fourth and ninth rows of F and G. These relations are refound in the fourth and ninth rows of $(I-KK^+)F$ and $(I-KK^+)G$. The third static relation is given by any of the rows 1,3,7,8 of $(I-KK^+)F$ and $(I-KK^+)G$. These four rows are in pairs linearly dependent. The existence of the third static relation is a consequence of the fact that the number of time derivatives of the variables v_i exceeds the rank of E and primarily of the assumptions made when basic physical laws were applied to the process.

Inspecting the system matrices A,B,C and D we find that D equals zero and that the two outputs equal the first two state variables as expected.

Several elements of A and B also approximately equal zero. However, some caution must be observed when approximating since the variables have true physical dimensions.

If the reduction is made manually it is indeed a very tedious work. It is therefore believed that this algorithm represents a very attractive solution to the reduction problem.

8. ACKNOWLEDGEMENT

The author would like to express his gratitude to Professor K-J Åström who proposed this problem and provided valuable guidance throughout its solution. I am also indebted to Civ.ing. K. Mårtensson who wrote some necessary subroutines used in the reduction program.

9. REFERENCES

1. Brown K.M.: A quadratically convergent method for solving simultaneous nonlinear equations. Doctoral Thesis, Dept. Computer Sciences, Purdue U., Lafayette, Ind., Aug. 1966. The algorithm is found in CACM. Algo 316.
2. Broyden C.G.: A Class of Methods for Solving Nonlinear Simultaneous Equations. Math. Comput. 19 (1965), 577-593.
3. Deutsch R.: Estimation theory. Prentice Hall, Inc. N.Y. 1965.
4. Eklund K.: Linear Mathematical Models of the Drum-Downcomer-Riser Loop of a Drum Boiler, Report 6809 November 1968, Lund Institute of Technology, Division of Automatic Control
5. Koenig H.E., Tokad Y., Kesavan H.K.: Analysis of discrete Physical Systems. McGraw-Hill N.Y. 1967.
6. Matweer V.A.: Methods for the approximate solution of a system of nonlinear equations. USSR Comp. Maths. and Math. Physics., vol 4, p. 1, 1964.
7. Saaty T.L., Brown J.: Nonlinear Mathematics. McGraw-Hill N.Y. 1964.
8. Åström K.J.: Reglerteori. Almqvist & Wiksell/Gebers Förlag AB, Stockholm 1968.

Table I

PRINTOUTS FROM PROGRAM RESTAF

MATRIX E

1.494+000	-0.000+000	-0.000+000	-0.000+000	-1.539+003	-0.000+000	-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	2.039+000	5.414+000	-0.000+000	-0.000+000
3.224-003	-0.000+000	-0.000+000	-0.000+000	1.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	1.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000
-0.000+000	1.242+004	3.484+001	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000
-0.000+000	1.242+004	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000
1.000+000	-1.312+002	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000

MATRIX F

-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	1.724-002	-1.724-002	-0.000+000	-0.000+000
2.899+001	-0.000+000	-0.000+000	-0.000+000	-2.058+004	7.665+000	7.140+000	-0.000+000	-0.000+000
7.320-005	-0.000+000	-1.632-004	-0.000+000	5.299-002	2.962-006	5.528-006	-0.000+000	-3.658-005
1.653+003	-0.000+000	-0.000+000	-3.305+003	-0.000+000	-0.000+000	-0.000+000	-0.000+000	1.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	2.468-005
-1.495+000	-0.000+000	2.974+000	-0.000+000	1.421+003	-8.864-001	1.000+000	1.758+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	1.358+003	-8.471-001	1.000+000	1.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	-1.024+002	-1.153-002	-0.000+000	-7.540-002	-0.000+000
3.418-001	-0.000+000	-6.837-001	-0.000+000	-0.000+000	-0.000+000	-0.000+000	1.000+000	-0.000+000

MATRIX G

-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000
-2.488-005	-0.000+000	-0.000+000
-0.000+000	-6.667-001	-0.000+000
-0.000+000	-1.000+000	-0.000+000
-0.000+000	-0.000+000	7.540-002
-0.000+000	-0.000+000	-0.000+000

Table I (continued)

```
MATRIX P

 1.000+000 -0.000+000
-0.000+000  1.000+000

MATRIX Q

-1.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000
-0.000+000 -1.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000 -0.000+000

MATRIX R

-0.000+000 -0.000+000 -0.000+000
-0.000+000 -0.000+000 -0.000+000


THE RANK OF MATRIX E= 6

THE LINEAR INDEPENDENT ROWS OF E

 1  2  0  0  5  6  7  8  0

THE LINEAR INDEPENDENT COLUMNS OF E,

 1  2  3  4  5  6  0  0  0
```

THE RESULT OF THE RANK FACTORIZATION E=K*L

MATRIX K

1.494+000	-0.000+000	-0.000+000	-0.000+000	-1.539+003	-0.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	2.039+000
3.224-003	-0.000+000	-0.000+000	-0.000+000	1.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	1.000+000	-0.000+000	-0.000+000
-0.000+000	1.242+004	3.484+001	-0.000+000	-0.000+000	-0.000+000
-0.000+000	1.242+004	-0.000+000	-0.000+000	-0.000+000	-0.000+000
1.000+000	-1.312+002	-0.000+000	-0.000+000	-0.000+000	-0.000+000
-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000	-0.000+000

MATRIX L

1.000+000	-4.729-009	-9.939-012	0.000+000	-1.244-008	0.000+000	0.000+000	0.000+000	0.000+000
-7.178-018	1.000+000	-6.327-014	0.000+000	7.198-014	0.000+000	0.000+000	0.000+000	0.000+000
-7.834-015	6.281-009	1.000+000	0.000+000	-1.856-011	0.000+000	0.000+000	0.000+000	0.000+000
0.000+000	0.000+000	0.000+000	1.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000
1.139-014	-2.645-012	-3.436-015	0.000+000	1.000+000	0.000+000	0.000+000	0.000+000	0.000+000
0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	1.000+000	2.656+000	0.000+000	0.000+000

THE PSEUDOINVERSE OF K

2.726-006	0.000+000	4.195-003	0.000+000	0.000+000	-2.853-013	1.056-002	1.000+000	0.000+000
2.318-012	0.000+000	3.567-009	0.000+000	0.000+000	-1.816-015	8.052-005	-1.496-011	0.000+000
-8.263-010	0.000+000	-1.272-006	0.000+000	0.000+000	2.870-002	-2.670-002	5.333-009	0.000+000
0.000+000	0.000+000	0.000+000	0.000+000	1.000+000	0.000+000	0.000+000	0.000+000	0.000+000
-6.498-004	0.000+000	4.494-006	0.000+000	0.000+000	-9.862-017	1.025-005	9.706-004	0.000+000
0.000+000	4.905-001	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000

MATRIX K(PSEUDO)*K SHOULD EQUAL THE UNIT MATRIX

1.000+000	-4.729-009	-9.939-012	0.000+000	-1.244-008	0.000+000
-7.178-018	1.000+000	-6.327-014	0.000+000	7.198-014	0.000+000
-7.834-015	6.281-009	1.000+000	0.000+000	-1.856-011	0.000+000
0.000+000	0.000+000	0.000+000	1.000+000	0.000+000	0.000+000
1.139-014	-2.645-012	-3.436-015	0.000+000	1.000+000	0.000+000
0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	1.000+000

MATRIX (I-K*K(PSEUDO))*F

4.756-008	0.000+000	-1.060-007	0.000+000	2.744-004	6.502-008	-3.248-008	1.767-007	-2.377-011
0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000
7.320-005	0.000+000	-1.632-004	0.000+000	4.223-001	1.001-004	-4.998-005	2.719-004	-3.658-008
1.653+003	0.000+000	0.000+000	-3.305+003	0.000+000	0.000+000	0.000+000	0.000+000	1.000+000
0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000	0.000+000
9.956-020	0.000+000	-2.220-019	0.000+000	2.857-008	-1.783-011	2.104-011	2.103-011	-4.975-023
-3.277-009	0.000+000	7.297-009	0.000+000	-1.870-005	-4.442-009	2.224-009	-1.202-008	1.621-012
-3.070-007	0.000+000	6.844-007	0.000+000	-1.771-003	-4.197-007	2.096-007	-1.141-006	1.534-010
3.418-001	0.000+000	-6.837-001	0.000+000	0.000+000	0.000+000	0.000+000	1.000+000	0.000+000

MATRIX (I-K*K(PSEUDO))*G

0.000+000	2.879-008	-2.055-007
0.000+000	0.000+000	0.000+000
0.000+000	4.431-005	-3.162-004
0.000+000	0.000+000	0.000+000
0.000+000	0.000+000	0.000+000
0.000+000	-2.104-011	9.215-015
0.000+000	-1.965-009	1.401-008
0.000+000	-1.858-007	1.326-006
0.000+000	0.000+000	0.000+000

MATRIX LT*L+FT*(I-K*K(PSEUDO))*F

2.731+006	-4.729-009	-2.337-001	-5.463+006	3.086-005	7.351-009	-3.690-009	3.418-001	1.653+003
-4.729-009	1.000+000	6.281-009	0.000+000	-2.573-012	0.000+000	0.000+000	0.000+000	0.000+000
-2.337-001	6.281-009	1.467+000	0.000+000	-6.883-005	-1.638-008	8.219-009	-6.837-001	5.969-012
-5.463+006	0.000+000	0.000+000	1.093+007	0.000+000	0.000+000	0.000+000	0.000+000	-3.305+003
3.086-005	-2.573-012	-6.883-005	0.000+000	1.178+000	4.222-005	-2.107-005	1.149-004	-1.545-008
7.353-009	0.000+000	-1.639-008	0.000+000	4.222-005	1.000+000	2.656+000	2.717-008	-3.660-012
-3.692-009	0.000+000	8.223-009	0.000+000	-2.107-005	2.656+000	7.054+000	-1.354-008	1.828-012
3.418-001	0.000+000	-6.837-001	0.000+000	1.149-004	2.717-008	-1.355-008	1.000+000	-9.947-012
1.653+003	0.000+000	5.969-012	-3.305+003	-1.545-008	-3.660-012	1.828-012	-9.947-012	1.000+000

MATRIX P(INVERSE)

1.000+000	0.000+000
0.000+000	1.000+000

Table I (continued)

SYSTEM MATRICES

MATRIX A

-3.061-002	-3.913-010	3.985-002	2.481-002	1.489+000	-1.078-004
-6.622-006	-7.854-013	-7.282-005	9.987-005	-4.578-001	-1.467-005
4.996-002	8.518-010	-1.004-001	1.144-003	-5.797+000	1.793-004
4.113-002	1.945-010	-9.056-012	-8.225-002	-5.180-009	-8.541-013
-3.506-005	-5.319-013	3.588-005	3.972-005	-5.311-002	-1.876-006
-1.337+001	-1.643-008	4.363-001	-2.473+000	1.872+004	-1.708+000

MATRIX B

0.000+000	1.422-003	-1.051-002
0.000+000	4.365-005	2.613-004

0.000+000	-9.989-003	2.991-003
2.488-005	6.663-013	2.336-012
0.000+000	-4.389-006	3.071-005
0.000+000	9.139-001	-6.469+000

MATRIX C

1.000+000	4.729-009	3.599-011	3.689-006	2.243-007	7.620-013
1.958-013	1.000+000	6.327-014	1.744-014	-6.960-014	8.611-020

MATRIX D

0.000+000	-1.736-011	-1.240-010
0.000+000	-2.065-019	-1.448-018