Geometric Theory for Multivariable Linear Systems with MATLAB Examples

Bernhardsson, Bo

1990

LUND UNIVERSITY

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

CODEN: LUTFD2/(TFRT-7447)/1-030/(1990)

Geometric Theory for Multivariable Linear Systems with MATLAB Examples

Bo Bernhardsson

Department of Automatic Control
Lund Institute of Technology
March 1990

Department of Automatic Control
Lund Institute of Technology
P.O. Box 118
S-221 00 Lund Sweden

Document name
Report

Date of issue
March 1990

Document Number
CODEN: LUTFD2/(TFRT-7447)/1-030/(1990)

Author(s)
Bo Bernhardsson

Supervisor
Per Hagander

Sponsoring organisation

Title and subtitle
Geometric Theory for Multivariable Linear Systems with MATLAB examples

Abstract

The geometric approach to analysis of multivariable dynamical systems is usually found hard to understand due to the lack of numerical examples. This report is intended to help the student read [Bengtsson, 1974] or [Wonham] by reviewing the basic concepts along with such examples.

The report also describes a set of MATLAB functions for geometric computations. These functions are based on the singular value decomposition and has been used in the course Linear Systems during the last two years.

Key words
Multivariable systems, Matlab, Geometric theory

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title

ISBN

Language
English

Number of pages
30

Recipient's notes

Security classification

Contents

1. Matlab algorithms for geometric computations

This section gives a brief overview of some algorithms that have been implemented in Matlab, see page 25. [Bengtsson] has also written algorithms like these but used QR-factorization for the basic calculations, the algorithms in this paper will use the SVD-factorization instead. Note that some of the algorithms have also been implemented for symbolic calculations in Macsyma, see [Holmberg]. Symbolic calculation is however only possible for low order examples.

Representations of linear subspaces

There are two possible representations of a linear subspace $\mathcal{A}$:

$$\begin{aligned} \textbf{Im}-\textbf{form}: &\quad \mathcal{A} = \text{Im}\,(A_1) \\ \textbf{Ker}-\textbf{form}: &\quad \mathcal{A} = \text{Ker}\,(A_2) \end{aligned}$$

$\mathcal{A}$ is spanned by the columns in A_1 and is orthogonal to the rows in A_2. In the following the notation $A := A_1$ and $A^{\perp} := A_2$ will be used. It is important to separate between $\mathcal{A}^{\perp}$ which is the subspace orthogonal to $\mathcal{A}$ and $A^{\perp}$ which is a matrix representing $\mathcal{A}$ in Ker-form.

Example The one-dimensional subspace to R^3 spanned by the first unit vector has the two representations

$$\text{Im}\,(\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}) = \text{Ker}\,(\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix})$$

Using the matlab-routines on page 24 a transformation from Im to Ker-form would be

$$A2 = \text{imtoker}([1,0,0]',3)$$

Exercise 1 Assume that $\mathcal{A} = \text{Im}\,(A_1) = \text{Ker}\,(A_2)$. Prove that the orthogonal complement of $\mathcal{A}$, denoted $\mathcal{A}^{\perp}$, has the representations $\mathcal{A}^{\perp} = \text{Im}\,(A_2^T) = \text{Ker}\,(A_1^T)$.

The SVD-decomposition

The SVD is a numerically stable way to transform a matrix A to diagonal form using two orthogonal transformations U and V

$$A = U\Sigma V^T = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix}$$

See figure 1 for a clarifying picture. In practice a user specified tolerance is used to distinguish which elements of Σ are zero.

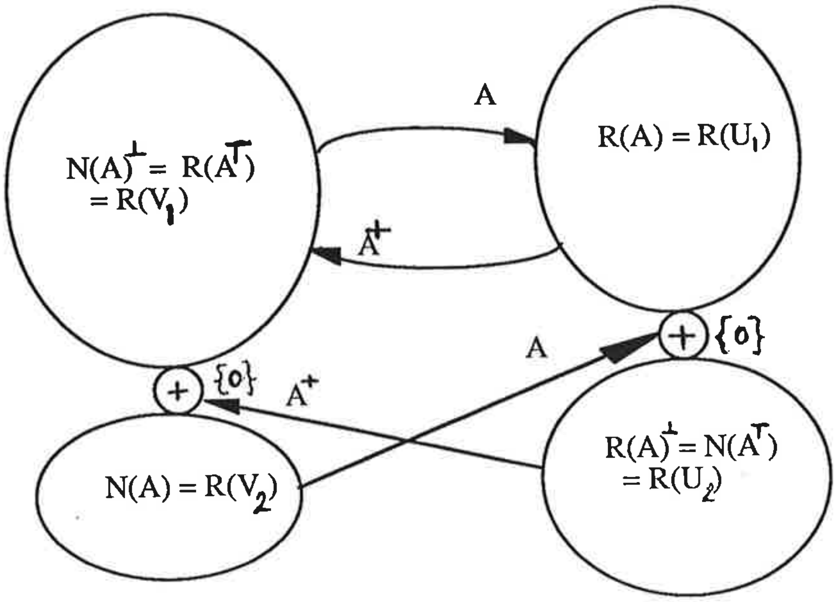

Figure 1. Very clarifying picture

Basic operations on linear subspaces

If the vectors describing the subspace are not linear independent, then the subspace can be described with a fewer number of vectors. These can be computed through row compression or column compression. Using the SVD also gives a numerically stable way to transform between the two representations. Most operations on linear subspaces can be directly implemented using the SVD.

Row compression

$$U^T A = \Sigma V^T = \begin{pmatrix} S V_1^T \\ 0 \end{pmatrix} \quad \leftarrow \text{ full row rank}$$

Column compression

$$AV = U\Sigma = \begin{pmatrix} U_1 S & 0 \end{pmatrix} \quad \text{(full column rank)}$$

Im-form to Ker-form

$$\text{Im}\,(A) = \text{Ker}\,(U_2^T)$$

Ker-form to Im-form Suppose a SVD for the ker-form $A^\perp$ is $A^\perp = \tilde{U}\Sigma\tilde{V}^T$, then

$$\text{Ker}\,(A^\perp) = \text{Im}\,(\tilde{V}_2^\perp)$$

Calculation of $\mathcal{A}+\mathcal{B}$, $\mathcal{A}\cap\mathcal{B}$, $A\mathcal{B}$, $A^{-1}\mathcal{B}$, $\mathcal{A}\subset\mathcal{B}$

$$\begin{aligned}
\mathcal{A}+\mathcal{B} &= \text{Im}\,(\begin{pmatrix} A & B \end{pmatrix}) \\
\mathcal{A}\cap\mathcal{B} &= \text{Ker}\,(\begin{pmatrix} A^{\perp} \\ B^{\perp} \end{pmatrix}) \\
A\mathcal{B} &= \text{Im}\,(AB) \\
A^{-1}\mathcal{B} &= \text{Ker}\,(B^{\perp}A) \\
\mathcal{A}\subset\mathcal{B} &\;:\; \text{rank}\,(\begin{pmatrix} A & B \end{pmatrix}) = \text{rank}\,(B)
\end{aligned}$$

Exercise 2 Show that $A^{-1}\mathcal{B} = \{x \mid Ax \in \mathcal{B}\} = \text{Ker}\,(B^{\perp}A)$.

2. A-invariant Subspaces

Definition Let A be a square $n \times n$ matrix and $\mathcal{V}$ a subspace of R^n, then $\mathcal{V}$ is A-invariant iff $A\mathcal{V} \subset \mathcal{V}$.

In control theory terms: $\mathcal{V}$ is A-invariant iff

$$x_{t+1} = Ax_t \quad , x_0 \in \mathcal{V} \quad \Longrightarrow \quad x_t \in \mathcal{V} \quad \forall t$$

Note that completing $\mathcal{V}$ to a basis for R^n and rewriting A in this basis, that is

$$T = \begin{pmatrix} \mathcal{V} & T_2 \end{pmatrix}, \quad A = T^{-1}A_{\text{old}}T$$

will give A the following structure:

$$A = \begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix}$$

Examples of A-invariant subspaces are the controllable subspace and the non-observable subspace. Compare with Kalman's decomposition theorem.

Definition If $\mathcal{V} = \text{Im}\,(V_1)$ is A-invariant, where V_1 is column compressed, then the **restriction** A_1 of A to $\mathcal{V}$ is given, uniquely, by

$$AV_1 = V_1A_1$$

The eigenvalues of A restricted to $\mathcal{V}$ can be calculated as

$$\text{eig}(A_1) = \text{eig}(V_1^{+}AV_1)$$

where V_1^{+} is the pseudoinverse of V_1.

Maximal A-invariant subspace in $\mathcal{S}$, $\mathcal{V}^*(\mathcal{S})$ (e.g. $\mathcal{S} =$ Ker (C))
"Unobservable subspace when inputs are known? (old stuff)".

Introduce the class

$$\overline{\mathcal{V}}(\mathcal{S}) = \{\mathcal{V} \mid \mathcal{V} \subset \mathcal{S} \ \& \ A\mathcal{V} \subset \mathcal{V}\} = \{\mathcal{V} \mid \mathcal{V} \subset (\mathcal{S} \cap A^{-1}\mathcal{V})\}$$

It is easy to show that $\overline{\mathcal{V}}(\mathcal{S})$ is closed under summation, that is, if two vector spaces are in $\overline{\mathcal{V}}(\mathcal{S})$ then so is their sum. We conclude that the subspace

$$\mathcal{V}^*(\mathcal{S}) = \text{ sum of all elements in } \overline{\mathcal{V}}(\mathcal{S})$$

is in $\overline{\mathcal{V}}(\mathcal{S})$ and by construction it is maximal. The maximal subspace is also unique.

Here is a constructive algorithm for $V^*(\mathcal{S})$:

$$\begin{cases} \mathcal{V}_0 = \mathcal{S} \\ \mathcal{V}_{i+1} = \mathcal{S} \cap A^{-1}\mathcal{V}_i \end{cases} \tag{1}$$
$$\text{Let } \mathcal{V}^* \text{ be the first } \mathcal{V}_\sigma \text{ such that } \mathcal{V}_{\sigma+1} = \mathcal{V}_\sigma$$

Assume that $\mathcal{S} =$ Ker (C). By exercise 2 in section 1, we can express the algorithm by

$$\begin{cases} \mathcal{V}_0 = \text{ Ker } (C) \\ \mathcal{V}_{i+1} = \text{ Ker } (C) \cap A^{-1}\mathcal{V}_i = \text{ Ker } \left(\begin{pmatrix} C \\ V_i^{\perp} A \end{pmatrix} \right) = [\text{induction}] \\ \qquad = \text{ Ker } \left(\begin{pmatrix} C \\ \begin{pmatrix} C \\ \vdots \\ CA^i \end{pmatrix} A \end{pmatrix} \right) = \text{ Ker } \left(\begin{pmatrix} C \\ CA \\ \vdots \\ CA^{i+1} \end{pmatrix} \right) \end{cases} \tag{2}$$

We therefore see that it is the usual algorithm for calculating maximal unobservable subspace. Also note that

$$\mathcal{V}_i = \{x \mid \ x, Ax, \ldots, A^i x \in \mathcal{S}\} \tag{3}$$

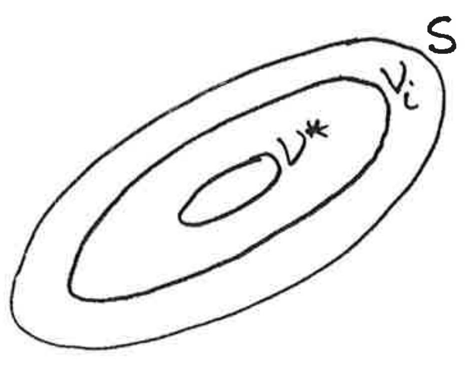

Figure 2. Illustrates algorithm (1)

THEOREM 1

Algorithm (1) converges in at most dim($\mathcal{S}$) steps to a $\mathcal{V}^*$ that is the maximal A-invariant subspace in $\mathcal{S}$.

Proof: From (3) we see that $\mathcal{V}_{i+1} \subset \mathcal{V}_i$ and that $\mathcal{V}^* \subset \mathcal{V}_i$. Since the dimension of $\mathcal{V}_i$ can not decrease for ever there will have to be a $\sigma \leq \dim(\mathcal{S})$ such that $\mathcal{V}_{\sigma+1} = \mathcal{V}_\sigma$, so the algorithm has converged. It is now easy to show that this $\mathcal{V}_\sigma \in \overline{\mathcal{V}}(\mathcal{S})$:

$$\mathcal{V}_\sigma = \mathcal{V}_{\sigma+1} = \mathcal{S} \cap A^{-1}\mathcal{V}_\sigma$$

Exercise 3 Show that if $\mathcal{S} =$ Ker (C) then σ=observability index of (A, C) as defined in [Kailath]. How is dim($\mathcal{V}_i$) related to all the observability indices and the "Crate 2"-diagram in [Kailath] ?

MATLAB : The function maxainv(A,S) described in the appendix returns the maximal A-invariant subspace in S (S given in Im-form).

3. (A,B)-invariant Subspaces

Definition: $\mathcal{V}$ is (A, B)-invariant iff $A\mathcal{V} \subset \mathcal{V} + \mathcal{B}$.

Consider the system

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ x_0 &\in \mathcal{V} \end{aligned} \tag{4}$$

$\mathcal{V}$ is (A, B)-invariant iff there is a control signal u_t such that $x_t \in \mathcal{V}, \quad \forall t$. Of course it is easier to be (A, B)-invariant than A-invariant.

The next theorem says that this control signal can be implemented as constant feedback from the states (which are supposed to be measurable).

THEOREM 2

$$A\mathcal{V} \subset \mathcal{V} + \mathcal{B} \quad \Longleftrightarrow \quad \exists F \quad (A + BF)\mathcal{V} \subset \mathcal{V}$$

Proof: $\Leftarrow$ is trivial since $BF\mathcal{V} \subset \mathcal{B}$
$\Rightarrow$ Choose a basis $\{v_1, \ldots, v_q\}$ in $\mathcal{V}$. From $A\mathcal{V} \subset \mathcal{V} + \mathcal{B}$ we know that there $\exists\ w_i \in \mathcal{V}, z_i \in \mathcal{U}$ such that

$$Av_i = w_i + Bz_i$$

Let F be a feedback matrix such that

$$Fv_i = -z_i$$

This is always possible since $\{v_i\}$ is a basis. We then get

$$(A + BF)v_i = w_i \in \mathcal{V}$$

which concludes the proof.

MATLAB: The function feedb(A,B,V) (B and V in Im-form) returns a feedback matrix such that $(A + BF)V \subset V$.

Which feedback matrices work ? The answer is given by the next theorem :

THEOREM 3

Suppose F_0 is such that $(A + BF_0)\mathcal{V} \subset \mathcal{V}$ then

$$(A + BF)\mathcal{V} \subset \mathcal{V} \Longleftrightarrow B(F - F_0)\mathcal{V} \subset \mathcal{V}$$

Proof: Exercise, or see [Bengtsson] theorem 3.2.

$\mathcal{V}_i$ are said to be **compatible** if there exists a common F that works, that is

$$(A + BF)\mathcal{V}_i \subset \mathcal{V}_i \qquad \forall i$$

It is an unsolved problem to find necessary and sufficient conditions for arbitrary subspaces to be compatible. Some results are shown in [Bengtsson].

Maximal (A,B)-invariant subspace in $\mathcal{S}$, $\mathcal{V}^*(\mathcal{S})$

"Unobservable subspace with unknown inputs ?"

The maximal (A, B)-invariant subspace in S is larger than the maximal A-invariant subspace in S.

Introduce the class

$$\overline{\mathcal{W}}(\mathcal{S}) = \{\mathcal{V} \mid \mathcal{V} \subset \mathcal{S} \ \& \ A\mathcal{V} \subset \mathcal{V} + \mathcal{B}\} = \{\mathcal{V} \mid \mathcal{V} \subset (\mathcal{S} \cap A^{-1}(\mathcal{V} + \mathcal{B}))\}$$

One can show that $\overline{\mathcal{W}}$ is closed under summation and we conclude as before that the subspace

$$\mathcal{V}^*(\mathcal{S}) = \text{ sum of all elements in } \overline{\mathcal{W}}$$

is in $\overline{\mathcal{W}}$ and by construction maximal. The maximal subspace is unique. For historical reasons we use the same notation $\mathcal{V}^*$ for maximal A-invariant and (A, B)-invariant subspaces. The context will determine which is meant.

Here is a constructive algorithm for $\mathcal{V}^*(\mathcal{S})$:

$$\begin{cases} \mathcal{V}_0 = \mathcal{S} \\ \mathcal{V}_{i+1} = \mathcal{S} \cap A^{-1}(\mathcal{V}_i + \mathcal{B}) \end{cases} \tag{5}$$

Let $\mathcal{V}^*$ be the first $\mathcal{V}_\sigma$ such that $\mathcal{V}_{\sigma+1} = \mathcal{V}_\sigma$

Exercise 4 Show that

$$\mathcal{V}_i = \{x_0 \mid \exists u_0, \ldots, u_{i-1} \text{ such that } x_t \in S, \ \forall t = 0, \ldots, i\} \tag{6}$$

Conclude that if $S = \ker(C)$ then $x \in V_i \Longleftrightarrow \mathcal{O}_i x \subset \operatorname{Im}(T_i)$ where

$$\mathcal{O}_i = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^i \end{pmatrix} \qquad T_i = \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 \\ CB & 0 & \cdots & 0 & 0 \\ CAB & CB & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots & 0 \\ CA^{i-1}B & \cdots & CAB & CB & 0 \end{pmatrix} \tag{7}$$

(see p. 80 [Kailath] or [AK]).

THEOREM 4

$\mathcal{V}_i$ in (5) converges in at most dim(S) steps to a $\mathcal{V}^*$ that is the maximal (A, B)-invariant subspace in $\mathcal{S}$.

Proof: (6) shows that $\mathcal{V}_{i+1} \subset \mathcal{V}_i$ and that $\mathcal{V}^* \subset \mathcal{V}_i$ (since for $x \in \mathcal{V}^*$ (6) should be valid for all i). Since the dimension can not decrease for ever there has to be a $\sigma \leq \dim(\mathcal{S})$ such that $\mathcal{V}_{\sigma+1} = \mathcal{V}_\sigma$, and the algorithm has converged. It is now easy to show that this $\mathcal{V}_\sigma \subset \overline{\mathcal{W}}(\mathcal{S})$

$$\mathcal{V}_\sigma = \mathcal{V}_{\sigma+1} = \mathcal{S} \cap A^{-1}(\mathcal{V}_\sigma + \mathcal{B})$$

MATLAB : The function maxabinv(A,B,S) returns the maximal (A, B)-invariant subspace in S (B and S given in Im-form).

Solution to the general static feedforward problem

We are now able to solve a non trivial problem using rather elegantly using the theory we have developed so far. Suppose we have the system

$$\begin{cases} x_{t+1} = Ax_t + Bu_t + Gv_t & x_0 = 0 \\ y_t = Cx_t \end{cases} \tag{8}$$

with the static feedforward-feedback law :

$$u_t = Fx_t + Hv_t$$

Is it possible to choose F, H so that the disturbance v_t in

$$\begin{cases} x_{t+1} = (A+BF)x_t + (G+BH)v_t & x_0 = 0 \\ y_t = Cx_t \end{cases}$$

does not effect the output y_t? Since $x_0 = 0$ this is equivalent to

$$\begin{cases} x_{t+1} = (A+BF)x_t \\ y_t = Cx_t = 0 \quad \forall t \end{cases} \qquad \forall x_1 \in \text{ Im }(G+BH)$$

The maximal subspace M for which $\exists F$ such that $x_1 \in M \Rightarrow x_t \in$ Ker (C), $\forall t$ is $M = \mathcal{V}^*$=maximal (A,B)-invariant subspace in Ker(C). So :

THEOREM 5

The feedforward-feedback problem is solvable iff there is a map H such that

$$\text{Im }(G+BH) \subset \mathcal{V}^*$$

this condition is equivalent to

$$\text{Im }(G) \subset \mathcal{V}^* + \mathcal{B}$$

Proof: All that remains to prove is the equivalence. $\Downarrow$ is trivial since $BH \subset \mathcal{B}$
$\Uparrow$ Take vectors $\{v_i\}$ such that $\{Gv_i\}$ spans Im (G). Then there is z_i such that $Gv_i = v^* + Bz_i$, where $v^* \in \mathcal{V}^*$ and we can choose H such that $Hv_i = z_i$ which gives $(G+BH)v_i = v^*$.

Warning There is no guarantee that the system $(A+BF)$ is stable.

Exercise 5 Construct an exercise that illustrates this theorem.

Disturbance decoupling problem (DDP) (v_t not measurable)

If v_t is not measurable we must set $H = 0$ above and the problem is to find F such that with $u_t = Fx_t$ the output is unaffected by the disturbance v_t. We obtain the following result

THEOREM 6

(DDP) is solvable $\Longleftrightarrow$ Im $(G) \subset \mathcal{V}^*$.

Proof: Just put $H = 0$ in theorem 5.

4. Controllability Subspaces

"Which x can be reached from 0 in such a way that $x_t \in$ Ker (C), $\forall t$?"

In this section we denote the controllable subspace with $\langle A \mid B \rangle$. So

$$\langle A \mid B \rangle = \text{Im}\,(B,\ AB,\ \ldots,\ A^{n-1}B)$$

The following results should be kept in mind

Exercise 6 $\langle A + BF \mid \mathcal{B} \rangle = \langle A \mid \mathcal{B} \rangle$

Exercise 7 (hard) If $\{A, B\}$ is controllable and $b \in$ Im $(B), b \neq 0$ then there is a F such that $\{A + BF, b\}$ is controllable. (It is theoretically only necessary with one control signal + feedback.)

Definition

$\mathcal{R}$ is a controllability subspace $\iff \exists F, G \ : \ \mathcal{R} = \langle A + BF \mid \text{Im}\,(BG) \rangle$

It is possible to obtain a definition with only F as unknown :

THEOREM 7

$$\exists F, G \ : \ \mathcal{R} = \langle A + BF \mid \text{Im}\,(BG) \rangle \iff \exists F \ : \ \mathcal{R} = \langle A + BF \mid \mathcal{B} \cap \mathcal{R} \rangle$$

Proof: $\Rightarrow$ Put $\widehat{\mathcal{B}} =$ Im (BG). Since $\widehat{\mathcal{B}} \subset \mathcal{R}$ and $\widehat{\mathcal{B}} \subset \mathcal{B}$, it follows that $\widehat{\mathcal{B}} \subset \mathcal{B} \cap \mathcal{R}$. So

$$\mathcal{R} = \langle A + BF \mid \widehat{\mathcal{B}} \rangle \subset \langle A + BF \mid \mathcal{B} \cap \mathcal{R} \rangle$$

Since $(A + BF)^i(\mathcal{B} \cap \mathcal{R}) \subset (A + BF)^i\mathcal{R} \subset \mathcal{R}$ we conclude that

$$\langle A + BF \mid \mathcal{B} \cap R \rangle \subset \mathcal{R}$$

Putting this together we conclude that $\mathcal{R} = \langle A + BF \mid \mathcal{B} \cap \mathcal{R} \rangle$
$\Leftarrow$ Let b_i be the i:th column in B and let $r_1, \ldots, r_q$ be a basis for $\mathcal{B} \cap \mathcal{R}$. Then

$$r_j = \sum_{i=1}^{m} g_{ij} b_i$$

for suitable g_{ij}. Let $G = \{g_{ij}\}$. Then $B \cap \mathcal{R} =$ Im (BG).

Exercise 8 Show that every controllability subspace is a subspace of the controllable subspace $\langle A \mid \mathcal{B} \rangle$.

Here is an algorithm to check if a subspace $\mathcal{R}$ is a controllability subspace. Note that no construction of F is needed :

$$\mathcal{R} \text{ is } (A,B)\text{-invariant, that is } A\mathcal{R} \subset \mathcal{R} + \mathcal{B} \text{ iff } \mathcal{R} = R_n \text{ where}$$
$$\begin{cases} R_0 = 0 \\ R_i = \mathcal{R} \cap (AR_{i-1} + \mathcal{B}) \quad i = 1, \dots, n \end{cases} \tag{9}$$

Proof: This follows from algorithm (11) below by putting $\mathcal{S} = \mathcal{R}$.

Maximal controllability subspace in S, $\mathcal{R}^*(\mathcal{S})$ $(\mathcal{R}^* \subset \mathcal{V}^*)$

Using algorithm (9) one can show that the class of controllability subspaces is closed under subspace addition. It can also be shown that the space $\{\mathcal{R} \mid \mathcal{R}$ is contr. subsp. & $\mathcal{R} \subset S\}$ is closed under addition. Hence it will, as before, have a maximal element, $\mathcal{R}^*(\mathcal{S})$. A proof of the following result can be found in [Bengtsson] theorem 4.3.

THEOREM 8

$$\begin{aligned} &\mathcal{R}^* = \langle A + BF \mid \mathcal{B} \cap \mathcal{V}^* \rangle \\ &\quad \text{where } \mathcal{V}^* = \text{ maximal } (A,B) - \text{invariant in } S \\ &\quad \text{and } F \text{ is chosen such that} \\ &(A+BF)\mathcal{V}^* \subset \mathcal{V}^* \quad (\Leftrightarrow \langle A + BF \mid \mathcal{V}^* \rangle = \mathcal{V}^*) \end{aligned} \tag{10}$$

Proof: See [Bengtsson] theorem 4.3.

Remark. Compare this result with the decomposition in theorem 7.6.2 in Kailath (also discussed in the next section). As we will see later, $\mathcal{R}^*$ is the space giving the last elements in the "suitable choice of basis" such that

$$A_{22} = \begin{pmatrix} \bar{A}_{22} & 0 \\ \bar{A}_{32} & \bar{A}_{33} \end{pmatrix} \quad B_2 = \begin{pmatrix} 0 \\ \bar{B}_3 \end{pmatrix}$$

and (10) say that $\{\bar{A}_{33}, \bar{B}_3\}$ is controllable. If $\mathcal{R}^* = 0$ there will be no $\bar{A}_{33}$ and $\bar{B}_3$. Also note that $\operatorname{Im}(\bar{B}_3) = \mathcal{B} \cap \mathcal{V}^*$.

Here is a constructive algorithm for the calculation of maximal controllability subspace in S :

$$\begin{aligned} &\begin{cases} \mathcal{R}_0 = 0 \\ \mathcal{R}_i = \mathcal{V}^* \cap (A\mathcal{R}_{i-1} + \mathcal{B}) \end{cases} \\ &\text{Let } \mathcal{R}^* \text{ be the first } \mathcal{R}_\sigma \text{ such that } \mathcal{R}_\sigma = \mathcal{R}_{\sigma+1} \end{aligned} \tag{11}$$

Proof: This is proved partly in [Bengtsson] and in full in [Wonham] theorem 5.6.

MATLAB : The function maxcs(A,B,S) returns the maximal controllability subspace in S (B and S in Im-form).

Spectral assignability

THEOREM 9

If $\mathcal{R}$ is a c.s. then for every symmetric set $\overline{\Lambda}$ of dim($\mathcal{R}$) complex numbers there $\exists F$ such that $(A+BF)\mathcal{R} \subset \mathcal{R}$ with $\sigma[(A+BF)\mid_{\mathcal{R}}] = \overline{\Lambda}$

Proof: According to theorem 7 we can choose F_0 and G such that $A_0 = (A+BF_0)\mid_{\mathcal{R}}$ and $B_0 = BG$ (where $\text{Im}\,(BG) = \mathcal{B} \cap \mathcal{R}$). We then have

$$\langle A_0 \mid B_0 \rangle = \mathcal{R}$$

and application of exercise 7 yields the existence of F_1, b such that

$$\mathcal{R} = \langle A_0 + B_0F_1 \mid b \rangle$$

and the scalar theorem on spectral assignability gives an f such that $\sigma(A_0 + B_0F_1 + bf) = \overline{\Lambda}$. Putting all feedbacks together concludes the proof.

A characterization of $\mathcal{R}^*(\mathcal{S})$

THEOREM 10

$\mathcal{R}^*(\mathcal{S})$ is exactly the x that can be reached from 0 in such a way that $x_t \in \mathcal{S} \quad \forall t$

Proof: This follows by induction from algorithm (11). Note that $\mathcal{V}^*$ is exactly the x that can be kept in S, $\forall t$. The problem with continuous time is treated in [Bengtsson] theorem 4.5.

Duality

The following intriguing result can now be obtained:

THEOREM 11

$$\mathcal{R}^* = \mathcal{V}^* \cap \mathcal{W}_*$$

where $\mathcal{W}_*^{\perp} = \max(A^T, C^T)$-invariant subspace in $\text{Ker}\ B^T$.

Proof: Rewriting (5) we get a recursive algorithm for $\mathcal{W}_*$:

$$\begin{cases} \mathcal{W}_0 = \mathcal{B} \\ \mathcal{W}_{i+1} = \mathcal{B} + A(\mathcal{W}_i \cap \text{Ker}\ C) \end{cases} \qquad (12)$$

Let $\mathcal{W}_*$ be the first $\mathcal{W}_\sigma$ such that $\mathcal{W}_{\sigma+1} = \mathcal{W}_\sigma$

It is enough to show that

$$\mathcal{R}_{i+1} = \mathcal{V}^* \cap \mathcal{W}_i, \qquad \forall i \qquad (*)$$

where $\mathcal{R}_i$ are the subspaces given in (11). Since $\mathcal{R}_1 = \mathcal{V}^* \cap \mathcal{B}$ and $W_0 = \mathcal{B}$ it is true for $i = 0$.

The algorithm for $\mathcal{R}$ is

$$\mathcal{R}_{i+1} = \mathcal{V}^* \cap (A\mathcal{R}_i + B)$$

Assume now that (*) above is true for $i-1$, then

$$A\mathcal{R}_i = A(\mathcal{V}^* \cap \mathcal{W}_{i-1})$$

But in the limit of the $\mathcal{V}^*$-algorithm we have

$$\mathcal{V}^* = \ker C \cap (A^{-1}(\mathcal{V}^* + \mathcal{B}))$$

so that

$$\begin{aligned} A\mathcal{R}_i + \mathcal{B} &= (A \ker C \cap (\mathcal{V}^* + \mathcal{B}) \cap A\mathcal{W}_{i-1}) + \mathcal{B} \\ &= ((\mathcal{V}^* + \mathcal{B}) \cap A(\mathcal{W}_{i-1} \cap \ker C)) + \mathcal{B} \\ &= ((\mathcal{V}^* + \mathcal{B}) \cap \{A(\mathcal{W}_{i-1} \cap \mathrm{Ker} C) + \mathcal{B}\}) + \mathcal{B} \\ &= ((\mathcal{V}^* + \mathcal{B}) \cap \mathcal{W}_i) + \mathcal{B} = \mathcal{V}^* \cap \mathcal{W}_i + \mathcal{B} \end{aligned}$$

Since $\mathcal{B} \in \mathcal{W}_i$ we now get

$$\mathcal{R}_{i+1} = \mathcal{V}^* \cap \mathcal{W}_i + \mathcal{V}^* \cap \mathcal{B} = \mathcal{V}^* \cap \mathcal{W}_i$$

□

5. State-space form

"Kailath chap 7.6 etc"

The notion of maximal (A, B) invariant subspace and maximal controllability subspace in Ker (C) enables us to extend Kalmans standard form, see p 133 [Kailath] or p 105 [AK]. We will investigate the controllable-observable sub-block further for extra structure. So let us assume that (A, B, C) is minimal.

We know that by using just feedback $A \rightarrow A + BF$ we can not loose controllability. Further the zero locations are unaffected by any feedback that does not affect the minimality (exercise 6.5.4). By putting as many poles as possible "under" the zeros we will thus obtain maximal unobservability. This will exploit the zero structure of (A, B, C)more extensively. Note that it is not enough to put a pole "under" a zero to increase unobservability, they have to be "at the same place" in the matrix. A minimal realization can have poles and zeros at the same place in the complex plane.

A question connected with maximal unobservability is: When can we from knowledge of just $y(t)$ ($u(t)$ unknown) calculate the initial state x_0 ? This is called "**perfect observability**".

Kailath theorem 7.6.2

Suppose (A,B,C) is minimal. Let $\mathcal{V}^*$ be the maximal (A,B)-invariant subspace in Ker (C). Choose feedback F such that $(A+BF)\mathcal{V}^* \subset \mathcal{V}^*$. Choose a basis in state-space with the last basis elements in $\mathcal{V}^*$. Then we will have

$$\begin{pmatrix} x_{i+1}^1 \\ x_{i+1}^2 \end{pmatrix} = \begin{pmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} x_i^1 \\ x_i^2 \end{pmatrix} + \begin{pmatrix} B_1 \\ B_2 \end{pmatrix} u_i$$

$$y_i = \begin{pmatrix} C & 0 \end{pmatrix} \begin{pmatrix} x_i^1 \\ x_i^2 \end{pmatrix}$$

Note that (if $\mathcal{V}^* \neq R^n$ so there is a $\overline{A}_{11}$-block) $B_1 \neq 0$ (otherwise (A,B) would not be controllable). So by a column compression (= we dont use extra input signals) on B_1 we can write

$$\begin{pmatrix} B_1 \\ B_2 \end{pmatrix} u_i = \begin{pmatrix} B_1 \\ B_2 \end{pmatrix} G\overline{u}_i = \begin{pmatrix} \overline{B}_1 & 0 \\ \overline{B}_{21} & \overline{B}_2 \end{pmatrix} \overline{u}_i$$

The column compression can be seen as transformation of the input: $u = G\overline{u}$. Since $\overline{B}_1$ now has full column rank we can perform a state transformation

$$T = \begin{pmatrix} I & 0 \\ X & I \end{pmatrix} \qquad A \to TAT^{-1} \quad B \to TB \quad C \to CT^{-1}$$

so that by proper choice of X (e.g. $X = -\overline{B}_{21}\overline{B}_1^+$ will do) we get a zero in the B_{21} position. Note that the structure of A and C is unaffected. Any extra feedback will have to be on the form

$$F = \begin{pmatrix} F_{11} & 0 \\ F_{21} & F_{22} \end{pmatrix}$$

not to destroy the maximal unobservability structure.

So we have Kailaths theorem 7.6.2. :

$$\begin{pmatrix} x_{i+1}^1 \\ x_{i+1}^2 \\ y_i \end{pmatrix} = \begin{pmatrix} A_{11}+B_1F_{11} & 0 & B_1 & 0 \\ A_{21}+B_2F_{21} & \overline{A}_{22}+B_2F_{22} & 0 & B_2 \\ C_1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_i^1 \\ x_i^2 \\ u_i \end{pmatrix} \tag{13}$$

Here $\{A_{11}+\overline{B}_1F_{11}, \overline{B}_1, C_1\}$ is perfectly observable, that is x_0^1 can be determined from knowledge of $\{y_i, i \leq 0\}$ irrespective of what the inputs $\{u_i\}$ are (if there were a $\{A_{11}, \overline{B}_1\}$ invariant subspace in Ker (C) we could obtain a larger $\mathcal{V}^*$, contradiction). Note that F_{11}, F_{21}, F_{22} can be chosen freely.

Let us dissect $\{\overline{A}_{22}, \overline{B}_2\}$ further. We know from the spectrum assignability theorem that the controllable subspace of $\{A_{22}, \overline{B}_2\}$ is exactly $\mathcal{R}^*$ so we can, by a suitable choice of basis arrange that

$$\overline{A}_{22} = \begin{pmatrix} \overline{A}_{22} & 0 \\ \overline{A}_{32} & \overline{A}_{33} \end{pmatrix}, \qquad \overline{B}_2 = \begin{pmatrix} 0 \\ \overline{B}_3 \end{pmatrix} \tag{14}$$

$\mathcal{R}^*$

where $\{\overline{A}_{33}, \overline{B}_3\}$ is controllable. So the unobservable modes can be divided into those of $\{\overline{A}_{33}, \overline{B}_3\}$ which can be put anywhere in the complex plane, and those of $\overline{A}_{22}$ which are fixed. We will in the next section see that these fixed zeros equals the transmission zeros (given by $\epsilon_i(s) = 0$)

6. The relation to the Smith form and zeros

Definition The **transmission zeros** are the zeros of $\epsilon_i(s)$ in the Smith-McMillan form of $H(s)$, see 446 [Kailath]. If $r < \min(n_o, n_i)$, where r is the normal rank of H, we say that $H(s)$ has zeros everywhere in the complex plane.

To study the zeros further we put

$$\mathcal{E}(s) = \begin{bmatrix} \epsilon_1(s) & & & \\ & \ddots & & 0 \\ & & \epsilon_r(s) & \\ & 0 & & 0 \end{bmatrix} \qquad (n_o \times n_i)$$

and introduce the pencil

$$P(s) = \begin{pmatrix} sI - A & B \\ -C & 0 \end{pmatrix}$$

It is shown in [Kailath] p. 448 (22) that for minimal systems (A, B, C) we have

$$P(s) \sim \begin{pmatrix} I_n & 0 \\ 0 & \mathcal{E}(s) \end{pmatrix} \tag{15}$$

Definition The **invariant zeros** are those s for which $P(s)$ looses normal rank. If normal rank $P(s) < \min(n_o, n_i)$ one say that $P(s)$ has invariant zeros everywhere in the complex plane.

From (14) we see that:

THEOREM 12

For minimal systems

$$\text{transmission zeros} = \text{invariant zeros}$$

A SISO-system cannot have zeros everywhere in the complex plane. To see this we use the appendix in [Kailath] to write

$$\det \begin{pmatrix} sI - A & b \\ -c & 0 \end{pmatrix} = \det(sI - A)c(sI - A)^{-1}b = a(s)\frac{b(s)}{a(s)} = b(s)$$

so $P(s)$ will have full normal rank (this means full rank except for a finite number of s). For MIMO-systems this is not necessarily true.

Remember (p. 449) that if $P(s)$ looses column rank for $s = s_0$ this means that there $\exists \begin{pmatrix} x_0 \\ u_0 \end{pmatrix}$ such that

$$\begin{pmatrix} s_0 I - A & B \\ -C & 0 \end{pmatrix} \begin{pmatrix} x_0 \\ u_0 \end{pmatrix} = 0 \Leftrightarrow$$

$$u(t) = u_0 e^{s_0 t} \Rightarrow \exists x_0 \text{ such that } y(t) = 0 \quad t \geq 0$$

We will now show that for minimal systems $P(s)$ has less than full column rank everywhere in the complex plane exactly when $\overline{A}_{33} \neq 0$, that is exactly when $\mathcal{R}^* \neq 0$:

We used only feedback and change of basis in input- and state-space to obtain (12 & 13). Since these transformations do not change the Smith- (or Kroenecker-) form we can use (12 & 13) to calculate the Smith form of $P(s)$. To show that the Smith form do not change with feedback we write

$$\begin{pmatrix} sI-(A+BF) & B \\ -C & 0 \end{pmatrix} = \begin{pmatrix} sI-A & B \\ -C & 0 \end{pmatrix} \begin{pmatrix} I & 0 \\ -F & I \end{pmatrix}$$

Note by the way that this proves that the invariant zeros are not affected by state-feedback. In the same way we can prove that input or state-space transformation do not change the Smith form of $P(s)$ either. So we have proved (p. 545):

THEOREM 13

$$P(s) \sim \begin{pmatrix} sI-A_{11} & & & B_1 & 0 \\ -\overline{A}_{21} & sI-\overline{A}_{22} & & 0 & 0 \\ -\overline{A}_{31} & -\overline{A}_{32} & sI-\overline{A}_{33} & 0 & \overline{B}_3 \\ C_1 & 0 & 0 & 0 & 0 \end{pmatrix} \sim$$

$$\sim \begin{pmatrix} sI-A_{11} & B_1 & & & \\ C_1 & 0 & & & \\ -\overline{A}_{21} & 0 & sI-\overline{A}_{22} & & \\ -\overline{A}_{31} & 0 & -\overline{A}_{32} & sI-\overline{A}_{33} & B_3 \end{pmatrix}$$

By using unimodular transformations we can go further (the structure at infinity will be destroyed however). Since $\{\overline{A}_{33}, \overline{B}_3\}$ is controllable

$$\begin{pmatrix} sI-\overline{A}_{33} & \overline{B}_3 \end{pmatrix} \sim \begin{pmatrix} I & 0 \end{pmatrix}$$

This is the PBH test in Smith form see e.g (40) p 366. Moreover we have (exercise or see [Kailath] 544)

$$\begin{pmatrix} sI-A_{11} & B_1 \\ C_1 & 0 \end{pmatrix} \sim \begin{pmatrix} I \\ 0 \end{pmatrix}$$

so we obtain the Smith form (see p 545) :

THEOREM 14

$$P(s) \sim \begin{pmatrix} I & & & \\ 0 & & & \\ & sI-\overline{A}_{22} & & \\ & & I & 0 \end{pmatrix}$$

From this we immediately see as promised that $P(s)$ has less than full column rank for all s iff $\{\overline{A}_{33}, \overline{B}_3\} \neq 0$ and that the eigenvalues of $\overline{A}_{22}$ are the s for which $P(s)$ looses normal column rank, that is the invariant zeros which are the same as the transmission zeros since we assumed a minimal system. Note that these can be calculated as the eigenvalues of $(A+BF)\,|_{\mathcal{V}^*/\mathcal{R}^*}$.

7. MATLAB and MACSYMA examples

Boiler example

The following example shows some of the geometrical calculations and illustrates the use of theorem 14 for calculation of transmission theory using geometric theory. The transmission zeros are the eigenvalues of $A+BF$ restricted to $\mathcal{V}^*/\mathcal{R}^*$.

```
>> a
a =
   -0.1290         0    0.0396    0.0250    0.0191
    0.0033         0   -0.0001    0.0001   -0.6210
    0.0718         0   -0.1000    0.0009   -3.8500
    0.0411         0         0   -0.0822         0
    0.0004         0    0.0000    0.0000   -0.0743

>> b
b =
         0    0.0014
         0    0.0000
         0   -0.0099
    0.0000         0
         0    0.0000

>> c
c =
     1     0     0     0     0
     0     1     0     0     0

>> s=kertoim(c)
s =
     0     0     0
     0     0     0
     1     0     0
     0     1     0
     0     0     1

>> vstar=maxabinv(a,b,s)
vstar =
    0.0000         0
    0.0000    0.0000
   -0.9930   -0.1180
    0.1180   -0.9930
    0.0017    0.0010

>> rstar=maxcs(a,b,s)
rstar =
     []

>> f=feedback(a,b,vstar)
f =
   1.0e+04 *

    0.0000    0.0000   -3.1081   -2.1641    0.0073
    0.0000    0.0000   -0.0028   -0.0018    0.0000

>> abf=a+b*f
abf =
   -0.1290    0.0000    0.0000    0.0000    0.0192
    0.0033    0.0000   -0.0011   -0.0005   -0.6210
    0.0718    0.0000    0.1815    0.1786   -3.8506
    0.0411    0.0000   -0.7739   -0.6211    0.0018
    0.0004    0.0000    0.0002    0.0001   -0.0743

>> subset(abf*vstar,vstar)
ans =
     1
```

```
>> eig(pinv(vr)*abf*vr)
ans =
    -3.680512037067634e-01
    -6.467751190094627e-02

>> tzero(a,b,c,[0 0 ;0 0])
ans =
      3.910567938947270e+30 + 1.062672189768686e+09i
      5.485100474083862e+07 - 3.430418706706532e-03i
     -5.485102340952299e+07 - 3.430433891841253e-03i
     -3.680512036036726e-01
     -6.467751189940564e-02
```

Note that MATLAB:s function tzero also give the zeros at infinity. The results corresponds very well.

Transmission zeros 2

This shows a typical example of calculation of transmission zeroes with a nontrivial $\mathcal{R}^*$. We start with a matrix on Kailath standard form, see p. 542.

```
>> a
a =
     1     0     0     0
     1     2     0     0
     1     0     3     0
     0     0     0     4

>> b
b =
     1     0
     0     0
     0     0
     0     1

>> c
c =
     1     0     0     0
     0     1     0     0

>> d
d =
     0     0
     0     0

>> s=kertoim(c)
s =
     0     0
     0     0
     1     0
     0     1

>> vstar=maxabinv(a,b,s)
vstar =
     0     0
     0     0
     1     0
     0     1

>> rstar=maxcs(a,b,s)
rstar =
     0
     0
     0
     1

>> f=feedback(a,b,vstar)
```

```
f =
         0         0         0         0
         0         0         0   -2.0000

>> abf= a+b*f
abf =
    1.0000         0         0         0
    1.0000    2.0000         0         0
    1.0000         0    3.0000         0
         0         0         0    2.0000

>> subset(abf*vstar,vstar)
ans =
     1

>> vr=over(vstar,rstar)
vr =
     0
     0
    -1
     0

>> eig(pinv(vr)*abf*vr)
ans =
     3

>> tzero(a,b,c,d)
ans =
    0.0000
    3.0000
```

Transmission zeros 3

This shows another example of calculation of transmission zeros with a non-trivial $\mathcal{R}^*$. It also shows that tzero in MATLAB can give extra zeros at an arbitrary position.

```
>> a
a =
     2     3     2     5
     1     2     0     0
     1     0     3     0
     4     2     6     9

>> b
b =
     1     0
     0     0
     0     0
     0     1

>> c
c =
     1     0     0     0
     0     1     0     0

>> d
d =
     0     0
     0     0

>> s=kertoim(c)
s =
     0     0
     0     0
     1     0
     0     1
```

```
>> vstar=maxabinv(a,b,s)
vstar =
     0     0
     0     0
     1     0
     0     1

>> rstar=maxcs(a,b,s)
rstar =
     0
     0
     0
     1

>> f=feedback(a,b,vstar)
f =
         0         0   -2.0000   -5.0000
         0         0   -3.0000   -4.5000

>> abf=a+b*f
abf =
    2.0000    3.0000         0         0
    1.0000    2.0000         0         0
    1.0000         0    3.0000         0
    4.0000    2.0000    3.0000    4.5000

>> subset(abf*vstar,vstar)
ans =
     1

>> vr=over(vstar,rstar)
vr =
     0
     0
    -1
     0

>> eig(pinv(vr*abf*vr)
     3

>> tzero(a,b,c,d)
ans =
     3.451039687098235e+17
     5.495643991996925e+00
     3.000000000000000e+00

>> tzero(a+b*rand(2,4),b,c,d)
ans =
    -2.882511903229958e-02
    -1.367140837864369e-16
     3.000000000000000e+00

>> tzero(a+b*rand(2,4),b,c,d)
ans =
     5.051333569801571e+00
     2.813478555864238e+00
     2.999999999999999e+00
```

The result of the last three calculations should be the same. As seen MATLAB gives different answers. This is because of the non trivial $\mathcal{R}^*$.

Smith form for the previous example

The following shows the calculation of the Smith form for the previous example. Note that (compare (22) p. 448 in Kailath)

$$P(s) \sim \begin{pmatrix} I_4 & 0 \\ 0 & \mathcal{E}(s) \end{pmatrix}$$

so this will again explain the zero structure.

```
(c1) load("ulf.mac");
```

(d1) $\quad ulf.mac$

```
(c2) a:matrix([2,3,2,5],[1,2,0,0],[1,0,3,0],[4,2,6,9]);
```

(d2)
$$\begin{bmatrix} 2 & 3 & 2 & 5 \\ 1 & 2 & 0 & 0 \\ 1 & 0 & 3 & 0 \\ 4 & 2 & 6 & 9 \end{bmatrix}$$

```
(c3) b:matrix([1,0],[0,0],[0,0],[0,1]);
```

(d3)
$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

```
(c4) c:matrix([1,0,0,0],[0,1,0,0]);
```

(d4)
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

```
(c5) g:c . (s*ident(4)-a)^^(-1) . b;
```

(d5)
$$\begin{bmatrix} \frac{s^3-14s^2+51s-54}{s^4-16s^3+54s^2-38s-39} & \frac{5s^2-25s+30}{s^4-16s^3+54s^2-38s-39} \\ \frac{s^2-12s+27}{s^4-16s^3+54s^2-38s-39} & \frac{5s-15}{s^4-16s^3+54s^2-38s-39} \end{bmatrix}$$

```
(c6) smith(g);
```

(d6)
$$\begin{bmatrix} \frac{s-3}{s^4-16s^3+54s^2-38s-39} & 0 \\ 0 & 0 \end{bmatrix}$$

Note that $P(s)$ has normal rank 5 although the system is both controllable and observable. This is a direct consequence from having a $\mathcal{R}^* \neq 0$, compare with (16) p. 545.

Tzero in Matlab can give 'strange' answer

This example shows that when P(s) do not have full rank, then MATLAB can give extra transmission zeros anywhere.

An example of transformations to Kailaths 7.6.2-form

```
First step: determine V*:

>> a

a =

    1.0000    3.0000    1.0000
    7.5000    0.0484    1.1129
    2.5000    1.5968    2.7258

>> b

b =

    2.0000    3.0000
         0   -1.7500
    1.0000   -2.0000

>> c

c =

     1     0     0

>> s=kertoim(c)

s =

     0     0
     1     0
     0     1

>> vstar=maxabinv(a,b,s)

vstar =

     0     0
     1     0
     0     1

Compute feedback matrix that makes V* (A+BF)-invariant:

>> f=feedb(a,b,vstar)

f =

         0   -1.3524   -1.2263
         0   -0.0984    0.4842

>> a0=a+b*f

a0 =

    1.0000    0.0000    0.0000
    7.5000    0.2206    0.2656
    2.5000    0.4412    0.5311

First step in Kailaths procedure finished.

Let us now transform B:

>> b
```

```
b =

    2.0000    3.0000
         0   -1.7500
    1.0000   -2.0000

>> b0=b*[1 -1.5 ; 0 1]  (input-trans. to zero b(1,2))

b0 =

    2.0000         0
         0   -1.7500
    1.0000   -3.5000

>> T        (state-trans. to zero out b0(3,1:2))

T =

    1.0000         0         0
         0    1.0000         0
   -0.5000         0    1.0000

>> b1=T*b0

b1 =

    2.0000         0
         0   -1.7500
         0   -3.5000

>> c1=c*inv(T)

c1 =

     1     0     0

>> a1=T*a0*inv(T)

a1 =

    1.0000    0.0000    0.0000
    7.6328    0.2206    0.2656
    2.2656    0.4412    0.5311


>> rstar=maxcs(a,b,s)   (determine contr.subs.)

rstar =

         0
    0.4472
    0.8944

Aha there is a controllability subspace:

Now transform to see uncontrollable block

>> T1=[1 0 0 ; 0 -2 1; 0 0 1];  (exercise)

>> b2=T1*b1

b2 =

    2.0000         0
         0         0
         0   -3.5000

>> a2=T1*a1*inv(T1)

a2 =
```

```
    1.0000    0.0000    0.0000
  -13.0000    0.0000    0.0000
    2.2656   -0.2206    0.7517
```

Kailaths form 7.6.2 !

By the way lets check the zeros with matlab:

```
>> tzero(a,b,c,d)

ans =

   1.3114e-15
```

correct !

8. Program documentation

Summary of functions

Im-form should be used in all function calls, except kertoim. Answers will be in Im-form, except imtoker. The following functions have been implemented:

Representations

`colcomp(A)`	Computes a column compressed A (matrix with full column rank)
`rowcomp(A)`	Computes a row compressed A (matrix with full row rank)
`imtoker(A,dim)`	Im-form to Ker-form (gives eye(dim) as answer when A is empty)
`kertoim(A,dim)`	Ker-form to Im-form (gives eye(dim) as answer when A is empty)

Basic operations

`cup(A,B)`	Gives the sum of A and B, $(A + B)$
`cut(A,B)`	Gives the intersection of A and B $(A \cap B)$
`invim(A,B)`	Computes the inverse image of B under A $(A^{-1}B)$.
`subset(A,B)`	0 if A is a subset of B, 1 otherwise $(A \subset B)$
`over(A,B)`	Computes representation for A over B (A/B)
`orth(A,dim)`	Calculates the orthogonal complement $(A^{\perp})$ (answer will be eye(dim) when A is empty)

Geometric computations

`maxainv(A,S)`	Maximal A-invariant subspace in S
`maxabinv(A,B,S)`	Maximal (A,B)-invariant subspace in S
`feedb(A,B,S)`	Calculates F such that $(A + BF)S \subset S$
`maxcs(A,B,S)`	Maximal controllability subspace in S

Matlab functions, documentation

```
function C=colcomp(A)
% function C=colcomp(A)
% computes a column compressed version of A
% AV = US   where A=USV' is the svd-decomposition of A
bigeps=1e6*eps;
[U,S,V]=svd(A);
r=rank(S,bigeps);
if r>0
   C(:,1:r)=U*S(:,1:r);
else
   C=[];
end;
```

```
function C=cup(A,B)
% C=cup(A,B)
% calculates the union of two vector spaces given on
% Im-form
% C = colcomp([A B]);
C=colcomp([A B]);


function C=cut(A,B)
% C=cut(A,B)
% calculates the intersection of two vector spaces
% given on Im-form
% C=kertoim([imtoker(A) ; imtoker(B)]);
% Some care has to be taken for empty matrices
if size(A)==0 ! size(B)==0
   C=[];
else
   [ar,ac]=size(A);
   C =kertoim([imtoker(A,-1) ; imtoker(B,-1)],ar);
end;


function F=feedb(A,B,V)
% function F=feedb(A,B,V)
% calculates a feedback vector F such that (A+BF)V C V
% B and V should be given in Im-form
% Note: One must have AV C V+B
if subset(A*V,[V B])
   [vr vc]=size(V);
   [br bc]=size(B);
   XZ=pinv([V B])*(A*V);
   Z=XZ(vc+1:vc+bc,:);
   F=-Z*(pinv(V')');
else
   disp('Warning: V is not (A,B)-invariant');
end;


function B=imtoker(A,dim)
% B=imtoker(A,dim)
% transforms an Im-form given by A to a Ker-form
% given by B
% ker(U2') = Im(A)
% Warning: imtoker([],dim)=eye(dim)
if size(A)==0
   B=eye(dim);
else
   bigeps=1e+6*eps;
   [U,S,V]=svd(A);
   r=rank(S,bigeps);
```

```
    if r<size(U)
       B(1:size(U)-r,:)=U(:,r+1:size(U))';
    else
       B=[];
    end;
end;


function C=invim(A,B)
% function C=invim(A,B)
% calculates the inverse image A^(-1)B , that is all
% y such that Ay is in B
% B should be given on Im-form, C will be in Im-form
% C=kertoim(imtoker(B)*A)
[ar,ac]=size(A);
C=kertoim(imtoker(B,ar)*A,ac);


function B=kertoim(A,dim)
% B=kertoim(A,dim)
% transforms an Ker-form given by A to a Im-form
% given by B
% Im(V2')=ker(A)
% Warning: kertoim([],dim)=eye(dim)
if size(A)==0
    B=eye(dim);
else
    bigeps=1e+6*eps;
    [U,S,V]=svd(A);
    r=rank(S,bigeps);
    if r<size(V)
       B(:,1:size(V)-r)=V(:,r+1:size(V));
    else
       B=[];
    end;
end;



function V=maxabinv(A,B,S)
% function maxabinv(A,B,S)
% calculates maximal (A,B)-invariant subspace in S.
% B and S should be given i Im-form, V will be
% in Im-form
% Alg. (12) p. 18 in Gunnar Bengtssons report
% (1974) is used
V0=S;
V1=cut(S,invim(A,cup(V0,B)));
while ~ subset(V0,V1)
    V0=V1;
    V1=cut(S,invim(A,cup(V0,B)));
```