BOLEY, DANIEL, LUCIUS

COMPUTING THE CONTROLLABILITY-OBSERVABILITY DECOMPOSITION OF A LINEAR TIME-INVARIANT DYNAMIC SYSTEM, A NUMERICAL APPROACH

Stanford University PH.D. 1981

COMPUTING THE CONTROLLABILITY-OBSERVABILITY DECOMPOSITION OF A LINEAR TIME-INVARIANT DYNAMIC SYSTEM, A NUMERICAL APPROACH.

A DISSERTATION

SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE

AND THE COMMITTEE ON GRADUATE STUDIES

OF STANFORD UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

by

Daniel Lucius Boley

June 1981

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Gene H. Golub

(Principal Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

James H. Wilkinson

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Joseph Oliger

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

John G. Herriot

Approved for the University Committee on Graduate Studies:

Harold J. Lieberman

Dean of Graduate Studies and Research

COMPUTING THE CONTROLLABILITY-OBSERVABILITY DECOMPOSITION OF A LINEAR TIME-INVARIANT DYNAMIC SYSTEM, A NUMERICAL APPROACH.

Daniel Boley

ABSTRACT

We examine various numerical properties involved in computing the complete Controllability-Observability (Kalman) Decomposition for a linear time-invariant dynamic system, of the form

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$$
$$\mathbf{y} = C\mathbf{x}.$$

where A, B, C are matrices (A square), and $\mathbf{u}$, $\mathbf{x}$, $\mathbf{y}$ are vector functions of time. In particular, we discuss the numerical stability, the cost, and the particular advantages of several algorithms. We also examine several ways to measure ill-conditioning in the data.

Here's to the man who invented stairs
And taught our feet to soar.
He was the first who ever burst
Into a second floor.
The world would be downstairs today
Had he not found the key,
So let his name go down in fame,
Whatever it may be.

– Oliver Herford

Preface

The fields of Numerical Analysis and Control Theory are both very young. Modern Numerical Analysis dates from the first attempts to solve mathematical problems on the earliest computers in the late 1940's and early 1950's. Modern Control Theory is even younger, going back to the early 1960's. In the 20 to 30 years since that time both fields have grown into mature disciplines. For Numerical Analysis in particular, problems in many areas of Applied Mathematics have been studied from a computational, numerical point of view, resulting in a large body of theory. However, not until the last 3 years did anyone think of applying some of the experience fom Numerical Analysis to problems in Control Theory. Among the poeple who took the lead in this direction were Alan Laub, Chris Paige and Paul Van Dooren (I have probably insulted loads of people by leaving them out). This thesis represents another attempt to analyse some specific problems in Control Theory using the theory and methods from Numerical Analysis. Hopefully it should further define the directions that might be pursued with respect to these problems.

This thesis would not exist today but for the help of many people. The first and foremost of these is without question my advisor Gene Golub. Without his unfailing support, technical, personal and financial, this thesis would never have been written. In fact, my original decision to become a Numerical Analyst was in large part due to his spirit and skill as a teacher. I am also indebted to James Wilkinson, with whom I have never had a conversation on any topic that was not inspiring, in spite of the fact that our total overlap of time spent together at Stanford did not amount to more than six months. In addition, this thesis could not have been written without Paul Van Dooren, Gene Franklin, Abbas Emami-Naeini. From a crash course in Control Theory to guidance during the writing of the thesis, they were always ready with essential and material assistance.

On a more general level, there are so many people that helped me in essential ways during my stay at Stanford that to list them all would require a whole book. To list a few: Mark Brown, Frank Luk, my family, everyone now or formerly part of the Numerical Analysis group at Stanford. Between the faculty, the students and the hordes of visitors passing through from all over the world, the Numerical Analysis group here was, and still is, a particularly lively, invigorating and pleasant place to work.

On the material side I am indebted to Gene Golub and various U.S. Government agencies for their financial support during most of my stay here, to the National Science Foundation for their Graduate Fellowship which allowed me to obtain Masters Degree with a broad based program in Computer Science, to the Stanford Linear Accelerator Center, funded by the U.S. Department of Energy, for extensive use of their computer time, and to Donald Knuth and the Stanford Computer Science Department for the opportunity to use the text formatting program TEX, which was used to type this thesis including all the mathematics.

One cannot possibly omit the fact that this research would have been impossible without the Singular Value Decomposition.

Daniel Boley
Stanford, CA, May 1981

Table of Contents

Chapter I. Introduction

Modern Control Theory attempts to build and analyse mathematical models of physical, social economic and electronic systems. Since the early papers dealing in Control Theory appeared in the early 1960's, the theory has grown into a large and fruitful discipline. Typically a system is modelled by a system of ordinary differential equations with extra right hand sides which stand for the inputs to the system, and an extra set of equations to represent the system outputs. Linear ordinary differential equations are frequently used because they can reasonably model whatever system is under study, and because they are very amenable to analysis.

A fundamental problem in linear Control Theory is to compute how far the inputs can be used to control the system, and how much information about the system can be discerned from the outputs. In this report, we ignore such items as constraints on the inputs or feedback from the output to the input, and we assume the parameters or coefficients do not change with time. The problem that remains is one of computing linear invariant subspaces, specifically that part which is controllable from the inputs and/or observable through the outputs. The problem of computing these spaces and manipulating them to obtain the combination spaces: controllable and observable, not controllable and observable, etc, is then what is addressed in this thesis.

The specific problem discussed in this paper is to compute the four-way decomposition of the system

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u} \\ \mathbf{y} &= C\mathbf{x} \end{aligned} \qquad (start)$$

into the controllable and observable (CO), controllable and not observable ($C\bar{O}$), not controllable and observable ($\bar{C}O$), and not controllable and not observable ($\bar{C}\bar{O}$) parts. If one has such a split, one can examine each part in isolation to determine certain properties, like how stable the system is, or how close to uncontrollable (or unobservable) certain variables are. To date no complete computer algorithm has been published; the best procedure so far was given by [Kalman], in which is outlined a method to compute the complete decomposition into the four parts. However, his – and others' – results have a number of deficiencies, not the least of which is the lack of regard for numerical stability. In addition, in a previous note [Boley], the author constructed an example for which the algorithm of [Kalman] failed.

In contrast to previous results, our goal here is to construct algorithms that

1) use numerically stable transformations,

2) have been programmed and tried on actual test cases on a computer.

In this paper we describe just such procedures. We use orthogonal transformations wherever possible, and where this is not possible, we use transformations whose conditioning depend on the separation of the eigenvalues of A, a property inherent to the problem. Examples are given which were actually run on an IBM 370/168 computer in FORTRAN double precision (about 16 decimal digits), though only 3 to 4 digits are shown in this report.

The circuit in Figure 1 should serve to illustrate this problem.

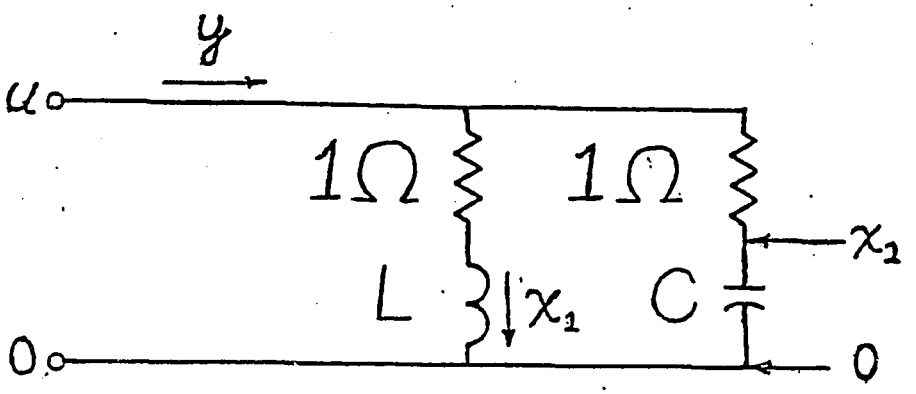

Figure 1.

We let u, the voltage across the terminals, be the input, and y, the current through the circuit at the terminals, be the output. We model the system with 2 internal states: x_1, the current through the inductor, and x_2, the voltage drop across the capacitor. Using Kirchoff's Laws, we can describe the system behavior with the equations:

$$\begin{aligned}\dot{x}_1 &= -\frac{1}{L}x_1 + \frac{1}{L}u \\ \dot{x}_2 &= -\frac{1}{C}x_2 + \frac{1}{C}u \\ y &= x_1 - x_2 + u.\end{aligned}$$

We write this in the usual matrix form common in Control Theory:

$$\begin{aligned}\begin{pmatrix}\dot{x}_1 \\ \dot{x}_2\end{pmatrix} &= \begin{pmatrix}-\frac{1}{L} & 0 \\ 0 & -\frac{1}{C}\end{pmatrix}\begin{pmatrix}x_1 \\ x_2\end{pmatrix} + \begin{pmatrix}\frac{1}{L} \\ \frac{1}{C}\end{pmatrix}u \\ y &= (1 \quad -1)\begin{pmatrix}x_1 \\ x_2\end{pmatrix} + (1)u\end{aligned}$$

If $L = C = -\frac{1}{k}$, we can transform this system by the orthogonal rotation

$$\begin{aligned}z_1 &= (x_1 + x_2)/\sqrt{2} \\ z_2 &= (x_1 - x_2)/\sqrt{2}\end{aligned}$$

to uncouple the equations, arriving at

$$\begin{aligned}\dot{z}_1 &= kz_1 - (k\sqrt{2})u \\ \dot{z}_2 &= kz_2 \\ y &= \sqrt{2}z_2 + u.\end{aligned}$$

Notice how we have been able to decompose the $\mathbf{x}$-space into 2 states, one which can be controlled by the input u, but has no effect on the output, and the other which can be observed

through the output y, but is independent of the input. If $k < 0$, z_2 will eventually decay to zero (if not there to start with), and it is evident that this circuit is equivalent to a simple resistor.

This is a much simplified example. In the general case, the problems we shall examine in this report will have the form

$$\begin{aligned} \dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t), \end{aligned} \qquad (start)$$

where A is a square n-by-n matrix, B and C are rectangular matrices, $\mathbf{x}$ is an n-vector of internal states in the model, $\mathbf{u}$ is the vector of inputs, and $\mathbf{y}$ is the vector of outputs. The term in u in the second equation for the example for figure 1 plays no role in the definition of controllability or observability and is omitted. Frequently, one adds feedback from output to the input in order, for example, to stabilize it, but this is beyond the scope of this paper.

The formal definition of controllable is as follows:

Definition. *A state $\mathbf{x}_0$ is controllable if it can be forced to zero in finite time by some input function. More precisely, a state $\mathbf{x}_0$ is controllable if there exists an input function $\mathbf{u}(t)$ defined on some interval $[t_0, t_1]$, such that if $\mathbf{x}(t_0) = \mathbf{x}_0$ and the function $\mathbf{u}$ is applied to the system, then at the finite time t_1, $\mathbf{x}(t_1) = 0$.*

If every state $\mathbf{x}$ is controllable, then the system is said to be completely controllable.

The set S_c of all controllable states is called the *controllable space*, and it is easy to show that it is indeed a linear subspace. (See [Kalman, Ho & Narendra].) In fact, we have the following purely algebraic definition for S_c:

Proposition.

$$S_c = \mathbf{range}(B \quad AB \quad A^2B \quad A^3B \quad \ldots)$$

Proof: *omitted,* see [Kalman, Ho & Narendra]. $$$

Observability is the dual concept to controllability: a system is said to be *completely observable* if the knowledge of the output function $y(t)$ over some finite interval $[t_0, t_1]$ is sufficient to determine the value of the state variables $\mathbf{x}$ at time t_0. From the purely algebraic point of view, we define the *unobservable space* to be

$$S_{\bar{o}} = \mathbf{nullsp}\begin{pmatrix} C \\ CA \\ CA^2 \\ CA^3 \\ \vdots \end{pmatrix}$$

The problems addressed in this paper are two-fold: first to compute S_c and $S_{\bar{o}}$, and second to combine the spaces to obtain the four combination spaces: the controllable-unobservable part $S_{c\bar{o}}$, the controllable-observable part S_{co}, the uncontrollable-unobservable part $S_{\bar{c}\bar{o}}$, and the uncontrollable-observable part $S_{\bar{c}o}$.

We will discuss the numerical properties of the various methods presented in some detail. In general, it is important to examine the numerical properties of the methods to see whether the

results obtained using them are reasonable and correct. In many cases, the "correct" answer can be extremely sensitive to perturbations in the data from the underlying problem. A simple case of this occurs in the example described in Figure .1. If the two resistors in that circuit do not have the exact same resistance value the whole analysis as described breaks down and the system becomes completely controllable and observable. If the resistance values are close to each other, a slight perturbation in their values will allow us to decompose the system into a controllable-unobservable and an uncontrollable- observable part. The question of how close is close depends on the tolerances in the original problem. In this thesis, we will attempt to measure how much perturbations in the original problem would affect the final computed results from the various methods.

In addition, one must take care that the methods themselves do not introduce numerical instabilities into the computed results. It may happen that, although a slight perturbation in the original problem would not change the final true result, the method used is so sensitive to such perturbations that the computed result it produces may well be affected. As an example, we consider an 11 by 11 system with randomly selected distinct eigenvalues

$$(5.121 \quad -1.127 \quad -0.899 \quad -0.779 \quad -0.373 \quad 0.041 \quad 0.727 \quad 0.905 \quad 0.506 \quad 0.472 \quad 0.954).$$

This is the example (*eleven*) described in the next chapter, where it is seen using the Staircase Algorithm that this system has a controllable part of dimension 7. If one attempts to compute this using the algebraic definition from the proposition above, one is led to computing the rank of the controllability matrix

$$(B \quad AB \quad A^2B \quad A^3B \quad \ldots).$$

The singular values of this matrix in this example are

$$10^7 \times \begin{pmatrix} 1.6 \\ 1.3 \times 10^{-6} \\ 7.0 \times 10^{-7} \\ 5.4 \times 10^{-7} \\ 1.3 \times 10^{-7} \\ 5.6 \times 10^{-8} \\ 3.1 \times 10^{-8} \\ 4.9 \times 10^{-17} \\ 8.7 \times 10^{-18} \\ 2.1 \times 10^{-18} \\ 7.0 \times 10^{-19} \end{pmatrix}.$$

If we consider any value less than 10^{-5} (relative to the norm of the controllability matrix) to be negligible then we would then be led to believe that the controllable part has dimension 1, but if we lower the zero tolerance to 10^{-8} we would then arrive at the correct answer. One can thus see

how the method itself can introduce numerical instabilities not present in the original problem, here shown in the poor scaling of the singular values. It also demonstrates how the interpretation of the results may demand some discretion on the part of the user.

History

There is actually very little in the literature on algorithms to compute the four combination spaces. The first mention was in [Kalman]. The method described in that paper was based on the idea of annihilating the appropriate elements in the coefficient matrices A, B, C, transforming the system (*start*) into canonical form. There was no thought given to numerical stability, and, in fact, it was not described in sufficient detail to be directly implemented. In addition, the method had a logical error described in [Boley]. The Matrix Algorithm presented in Chapter 3 of this report is a corrected version of this algorithm, using numerically stable transformations wherever possible, which works under the assumption that the eigenvalues are distinct.

Another method to compute the combination spaces is described in [Desoer] and [Wonham]. This method is based on the idea of computing the relevant subspaces of the x-space directly, using the geometry of the system. A concise description of the idea, using notation very similar to this report, appears as the proof to the *Canonical Decomposition Existence Theorem* in [Desoer], but without the algorithmic details. The Geometric Method in Chapter 3 of this report is an implementation of this method, paying due regard to numerical stability, and is described also in [Emami-Naeini & Franklin] and [Boley, Emami-Naeini & Franklin]. In the algorithm a subspace is represented by a set of column vectors which form an orthonormal basis for the space.

There are many more methods in the literature for computing the simple controllability decomposition. In this thesis, two such methods are discussed. The most popular by far is the so-called Staircase Algorithm. There are so many papers on this algorithm that I mention here those with particular relevance to numerical computation. One of the earliest references appears in the book [Rosenbrock], in which the author develops the concepts in terms of transfer functions. He defines the problem, sketches the method, and even carries it out on a few small examples. His point of view is very different from that given here, and the method as he has it takes no account of numerical stability.

Following [Rosenbrock], there appeared several papers discussing variations and improvements to his algorithm. They generally try to compute the Luenberger or Echelon Canonical Form, in which all the pivot elements are forced to unity, a numerically unstable process. Some of these papers are [Mayne], [Tse, Medanic & Perkins], [Daly], as well as the book [Wonham]. Numerical stability began to receive some attention in [Van Dooren, Emami-Naeini & Silverman], in which they suggest the use of the Singular Value Decomposition, a reliable way to compute ranks (see below). Subsequently a paper by [Konstantinov et al] appeared with a very similar idea. Neither paper examines the consequences of using the S.V.D. in terms of stability of the algorithm as a whole, for example, by looking at numerical examples. An interesting general discussion on the numerical problems encountered in trying to compute the controllable part occurs in [Paige].

The Modal Method for computing the controllable space is based on a very simple idea: given a unique decomposition of the entire x-space into a direct sum of disjoint subspaces, invariant under the mapping A, compute which of the subspaces are needed to cover the vectors B. So far, it does not appear that there are any references to such an approach, although similar ideas are used in methods to adjust the modes (eigenvalues) in order to make the system stable.

Two recent books on Control Theory might be of interest to the reader. One, [Luenberger], is an introductory level book, but full of interesting examples from a wide variety of disciplines. This book presents the more basic concepts of Control Theory for the uninitiated. The other book, [Kailath], also has interesting examples, and also provides a good solid foundation to the Theory of Control. It is at a more advanced level, discussing state-of-the-art methods.

There are many results from Numerical Linear Algebra that play important roles in this report. The foremost is the Singular Value Decomposition (see e.g. [Golub & Reinsch]), in which a matrix A is decomposed into $A = U\Sigma V^T$, with U, V unitary and Σ non-negative diagonal. A much cheaper, but somewhat less robust, method to compute ranks is the QR-Decomposition With Pivoting [LINPACK], in which A is decomposed into $A = QRE$, with Q unitary, R "graded" upper triangle, and E a permutation matrix recording the column interchanges. (Here "Graded" means that in every right principal submatrix (consisting of all but the first k rows and columns, $k = 0, \ldots, n-1$), the 1, 1 element of that submatrix is larger than the 2-norm of any other column in the submatrix. With this property, the bottom row of R is normally very small in cases where R is rank-deficient.) This is a direct method, as opposed to the slower iterative S.V.D.

An important paper regarding bounds on perturbed subspaces is the S.I.A.M. Review paper [Stewart 1973b]. His bounds depend on the separation between the eigenvalues, and are based on a parameter that is very difficult to compute. He summarizes much previous work [Varah], [Kahan & Davis], etc. [Wedin BIT]. The bounds that result from this theory are very pessimistic because they do not take into account the effect of trying to cover or embed one subspace in another where both may be perturbed. The theory for two-space perturbations needs to be developed.

Chapter II. Computing the Controllable Space

Section II A. Staircase Algorithm

In this chapter, we describe two methods for computing the controllable space of a linear system. The first of these is the Staircase Algorithm. In this section we describe how the Staircase Algorithm is used in computing the controllable space of

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u}; \\ \mathbf{y} &= C\mathbf{x}. \end{aligned} \qquad (start)$$

The observable space can be computed in an analogous manner by applying this procedure to the transpose of the system.

This algorithm has appeared in many forms, but the first allusion to it appeared in [Rosenbrock].

The essential building block used in this method is the decomposition

$$M = Q\begin{pmatrix} R \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \qquad (rank)$$

where Q is orthogonal, and R has full row rank. A very stable method to compute the rank of M is the Singular Value Decomposition (S.V.D.) [Golub], [Golub & Reinsch]

$$M = U\begin{pmatrix} \Sigma_{11} & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} V_1^{\mathrm{T}} \\ V_2^{\mathrm{T}} \end{pmatrix}.$$

We can then obtain $(rank)$ by setting $Q = U$ and $R = \Sigma_{11}V_1^{\mathrm{T}}$.

One can compute this decomposition with fewer operations, and almost as reliably, using the QR-Decomposition with Pivoting [LINPACK], which gives rise to the QRE Decomposition in which a matrix M is decomposed as

$$M = QRE$$

where Q is an orthogonal matrix, E is a permutation matrix, and R is upper triangular with the property that

$$|r_{ii}| \geq |r_{jk}| \quad \text{for all } i, j, k,\ i \leq j, i \leq k$$

[LINPACK]. An R with this property is said to be *graded.* We then obtain $(rank)$ by using the same Q from this decomposition and setting the R in $(rank)$ to be the RE from this decomposition. In the current implementation, the QRE Decomposition is used.

A note on notation: In this section I will use the notation FRR to mean Full Row Rank, FCR to mean Full Column Rank, and I will put iteration numbers as superscripts.

The Staircase Algorithm consists of a series of similarity transformations to the system $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ as follows:

Stage 0.

Let

$$T^{(0)-1}\begin{pmatrix} B_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

be the QR-decomposition of B, where B_1 is of full row rank, and $T^{(0)}$ is orthogonal.

We then apply the transformation as follows:

$$\begin{aligned} A^{(1)} &= T^{(0)} A T^{(0)\mathrm{T}} \\ B^{(1)} &= T^{(0)} B = \begin{pmatrix} B_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \\ C^{(1)} &= C T^{(0)\mathrm{T}}, \end{aligned} \tag{D.1}$$

The part $B^{(1)}$ will be unchanged in subsequent stages.

Stage i, i=1,...,k

At each subsequent stage (i) we compute an orthogonal $T^{(i)}$ and apply it to $A^{(i)}$, $B^{(i)}$, $C^{(i)}$, as in (D.1), to obtain $A^{(i+1)}$, $B^{(i+1)}$, $C^{(i+1)}$ so that

$$\begin{aligned} A^{(i+1)} &= T^{(i)} A^{(i)} T^{(i)\mathrm{T}} \\ B^{(i+1)} &= T^{(i)} B^{(i)} = B^{(1)} \\ C^{(i+1)} &= C^{(i)} T^{(i)\mathrm{T}}. \end{aligned} \tag{D.2}$$

To describe the i-th stage we must define some additional notation. (I recommend you read what follows first with $i = 1$ in your mind, noting that in this case the entire upper left part in (D.3a) will be empty!) At the i-th stage we have (The superscripts give the number of the last stage in which each block was changed.)

$$A^{(i)} = \left(\begin{array}{ccccccc|c|c} A_{1,1}^{(1)} & A_{1,2}^{(2)} & A_{1,3}^{(3)} & \cdots & A_{1,i-2}^{(i-1)} & A_{1,i-1}^{(i-1)} & A_{1,i}^{(i)} & & \\ A_{2,1}^{(2)} & A_{2,2}^{(2)} & A_{2,3}^{(3)} & \cdots & A_{2,i-2}^{(i-1)} & A_{2,i-1}^{(i-1)} & A_{2,i}^{(i)} & & \\ & A_{3,2}^{(3)} & A_{3,3}^{(3)} & \cdots & A_{3,i-2}^{(i-1)} & A_{3,i-1}^{(i-1)} & A_{3,i}^{(i)} & & \\ & 0 & & & \vdots & \vdots & \vdots & & X \\ & & & & A_{i-2,i-2}^{(i-1)} & A_{i-2,i-1}^{(i-1)} & A_{i-2,i}^{(i)} & & \\ & & & & A_{i-1,i-2}^{(i-1)} & A_{i-1,i-1}^{(i-1)} & A_{i-1,i}^{(i)} & & \\ \hline & 0 & & \cdots & 0 & A_{i,i-1}^{(i-1)} & A_{i,i}^{(i)} & & X \\ \hline & & & & & 0 & A_{i+1,i}^{(i)} & & X \\ & 0 & & \cdots & 0 & & X & & X \end{array}\right) \tag{D.3a}$$

We write the above partition, grouped by the lines, as

$$A^{(i)} = \begin{pmatrix} F_{1,1}^{(i)} & F_{1,2}^{(i)} & F_{1,3}^{(i)} \\ F_{2,1}^{(i)} & F_{2,2}^{(i)} & F_{2,3}^{(i)} \\ F_{3,1}^{(i)} & F_{3,2}^{(i)} & F_{3,3}^{(i)} \end{pmatrix} \qquad (D.3b)$$

where

$F_{1,1}^{(i)}$ is square of size $r_1 + \ldots + r_{i-1}$,

$F_{2,2}^{(i)} = A_{ii}^{(i)}$ is of size r_i by r_i,

$A^{(i)}$ (the entire matrix) is n by n,

$F_{2,1}^{(i)}$ and $F_{3,2}^{(i)}$ have rank r_i and r_{i+1} respectively,

and the other blocks have dimensions to match.

At stage $i = 1, F_{1,1}^{(1)}$ is empty, so we have

$$A^{(1)} = \begin{pmatrix} F_{2,2}^{(1)} & F_{2,3}^{(1)} \\ F_{3,2}^{(1)} & F_{3,3}^{(1)} \end{pmatrix} = \begin{pmatrix} A_{1,1}^{(1)} & X \\ X & X \end{pmatrix}$$

Then the i-th stage proceeds as follows:

a) Decompose

$$Q^{\mathrm{T}} \begin{pmatrix} R \\ 0 \\ \vdots \\ 0 \end{pmatrix} = F_{3,2}^{(i)} \qquad (stage.a)$$

with Q orthogonal, R with full row rank r_{i+1} using some scheme based on, for example, the QRE Decomposition.

b) Let

$$T^{(i)} = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & Q \end{pmatrix} \qquad (stage.b)$$

with blocks matching those in $(D.3b)$

c) Then

$$\begin{aligned} A^{(i+1)} &= T^{(i)} A^{(i)} (T^{(i)})^{\mathrm{T}} \\ &= \begin{pmatrix} F_{1,1}^{(i)} & F_{1,2}^{(i)} & F_{1,3}^{(i)} Q^{\mathrm{T}} \\ F_{2,1}^{(i)} & F_{2,2}^{(i)} & F_{2,3}^{(i)} Q^{\mathrm{T}} \\ Q F_{3,1}^{(i)} & Q F_{3,2}^{(i)} & Q F_{3,3}^{(i)} Q^{\mathrm{T}} \end{pmatrix} \\ &= \begin{pmatrix} \bar{F}_{1,1}^{(i)} & \bar{F}_{1,2}^{(i)} & \bar{F}_{1,3}^{(i)} \\ \bar{F}_{2,1}^{(i)} & \bar{F}_{2,2}^{(i)} & \bar{F}_{2,3}^{(i)} \\ 0 & \begin{pmatrix} R \\ 0 \\ \vdots \\ 0 \end{pmatrix} & \bar{F}_{3,3}^{(i)} \end{pmatrix} \end{aligned} \qquad (stage.c)$$

where $F^{(i)}_{1,1} = \bar{F}^{(i)}_{1,1}$.

d) We then repartition the matrix into sub-matrices in preparation for the next stage $(i+1)$, so that the result of the i-th stage (*stage.c*) is rewritten

$$A^{(i+1)} = \begin{pmatrix} F^{(i+1)}_{1,1} & F^{(i+1)}_{1,2} & F^{(i+1)}_{1,3} \\ F^{(i+1)}_{2,1} & F^{(i+1)}_{2,2} & F^{(i+1)}_{2,3} \\ F^{(i+1)}_{3,1} & F^{(i+1)}_{3,2} & F^{(i+1)}_{3,3} \end{pmatrix} \qquad (stage.d)$$

where

$$F^{(i+1)}_{1,1} = \begin{pmatrix} \bar{F}^{(i)}_{1,1} & \bar{F}^{(i)}_{1,2} \\ \bar{F}^{(i)}_{2,1} & \bar{F}^{(i)}_{2,2} \end{pmatrix} \text{ square of size } r_1 + \ldots + r_i$$

$$F^{(i+1)}_{2,1} = (0 \ \cdots \ 0 \ R)$$

$$F^{(i+1)}_{2,2} = A^{(i+1)}_{i+1,i+1} \quad \text{from } (D.3a) \quad (r_{i+1} \text{ by } r_{i+1})$$
$$\text{(part of subblock } \bar{F}^{(i)}_{33})$$

$$\begin{pmatrix} F^{(i+1)}_{1,2} & F^{(i+1)}_{1,3} \end{pmatrix} = \begin{pmatrix} \bar{F}^{(i)}_{1,3} \\ \bar{F}^{(i)}_{2,3} \end{pmatrix}$$

$$\begin{pmatrix} F^{(i+1)}_{2,2} & F^{(i+1)}_{2,3} \\ F^{(i+1)}_{3,2} & F^{(i+1)}_{3,3} \end{pmatrix} = \bar{F}^{(i)}_{3,3}$$

We are now ready to do the $(i+1)$-th stage. Observe that the block denoted at the i-th stage by $F^{(i)}_{1,1}$ is not changed at the i-th stage, or at subsequent stages.

We repeat stage i for $i = 1, 2, \ldots$ until, say at stage k, the block $F^{(k)}_{3,2}$ has rank zero, either because it has all zero entries, or because it is an empty block (e.g. has zero rows, in which case, the whole system is entirely controllable.) Thus at the final stage (k), the matrix can be written as

$$A^{(k)} = \begin{pmatrix} F^{(k)}_{1,1} & F^{(k)}_{1,2} & F^{(k)}_{1,3} \\ F^{(k)}_{2,1} & F^{(k)}_{2,2} & F^{(k)}_{2,3} \\ 0 & 0 & F^{(k)}_{3,3} \end{pmatrix}. \qquad (D.4a)$$

We can group together the 1,3 and 2,3 blocks to get

$$A^{(k)} = \begin{pmatrix} \begin{pmatrix} \bar{F}^{(k)}_{1,1} & \bar{F}^{(k)}_{1,2} \\ \bar{F}^{(k)}_{2,1} & \bar{F}^{(k)}_{2,2} \end{pmatrix} & \bar{A}^{(k)}_{1,2} \\ 0 & \bar{A}^{(k)}_{2,2} \end{pmatrix}. \qquad (D.4b)$$

We may expand $(D.4b)$ to obtain the matrix $(D.5a)$ where the bars mark the partition in $(D.4b)$:

$$A^{(k)} = \left(\begin{array}{ccccccc|c} A_{1,1}^{(1)} & A_{1,2}^{(2)} & A_{1,3}^{(3)} & \cdots & A_{1,k-2}^{(k-1)} & A_{1,k-1}^{(k-1)} & A_{1,k}^{(k)} & \\ A_{2,1}^{(2)} & A_{2,2}^{(2)} & A_{2,3}^{(3)} & \cdots & A_{2,k-2}^{(k-1)} & A_{2,k-1}^{(k-1)} & A_{2,k}^{(k)} & \\ & A_{3,2}^{(3)} & A_{3,3}^{(3)} & \cdots & A_{3,k-2}^{(k-1)} & A_{3,k-1}^{(k-1)} & A_{3,k}^{(k)} & \\ & 0 & & & \vdots & \vdots & \vdots & \bar{A}_{1,2}^{(k)} \\ & & & & A_{k-2,k-2}^{(k-1)} & A_{k-2,k-1}^{(k-1)} & A_{k-2,k}^{(k)} & \\ & & & & A_{k-1,k-2}^{(k-1)} & A_{k-1,k-1}^{(k-1)} & A_{k-1,k}^{(k)} & \\ \hline & & 0 & = & \bar{A}_{2,1}^{(k)} & & & \bar{A}_{2,2}^{(k)} \end{array}\right) \qquad (D.5a)$$

where

$$\begin{aligned} &A_{2,1}^{(2)}, \ldots, A_{k-1,k-2}^{(k-1)} \text{ all have full row rank } r_2, \ldots, r_{k-1} \text{ respectively,} \\ &A_{1,1}^{(1)}, \ldots, A_{k-1,k-1}^{(k-1)} \text{ are of size } r_1 \text{ by } r_1, \ldots, r_{k-1} \text{ by } r_{k-1} \text{ respectively,} \\ &\bar{A}_{2,1}^{(k)} = 0, \bar{A}_{2,2}^{(k)}, \bar{A}_{1,2}^{(k)} \text{ may all be empty, if entirely controllable,} \\ &r_1 \geq r_2 \geq \ldots \geq r_{k-1}. \end{aligned} \qquad (D.5b)$$

(The superscripts give the number of the last stage in which each block was changed.)
We also have

$$B^{(k)} = \begin{pmatrix} B_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = B^{(1)} \qquad (D.5c)$$

We can give a very easy estimate on the cost in the number of operations required by considering that the number of multiplications or additions/subtractions is approximately equal to the number of matrix elements annihilated. But that is no more than for the QR decomposition, so we can bound the cost simply by the the cost of doing a QR decomposition on A. That cost is approximately $\frac{4}{3}n^3 + O(n)$. Since the decomposition stops when the controllable part has been computed, after n_c steps, the cost of a staircase sweep becomes approximately

$$\text{cost} \quad \approx \quad \frac{4}{3}n^3 - \frac{4}{3}n_{\bar{c}}^3 + O(n),$$

where $n_{\bar{c}} = n - n_c$ is the size of the uncontrollable part.

Theoretical Proof

Theorem 1. *(Stair)*

Given $A^{(k)}$ and $B^{(k)}$ as in (D.5), then we have a controllability decomposition of $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$.

Proof.

Observe that

(a) $\mathbf{colsp}(B^{(k)}) = \mathbf{span}(e_1 \ \cdots \ e_{r_1})$,

(b)

$$A^{(k)}B^{(k)} = \begin{pmatrix} X \\ A_{2,1}^{(2)}B_1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Since $A_{2,1}^{(2)}$ has full row rank r_2 and B_1 has full row rank $r_1 \geq r_2$, it follows $A_{2,1}^{(2)}B_1$ has full row rank r_2 and

$$\mathbf{rank}\begin{pmatrix} B_1 & X \\ 0 & A_{2,1}^{(2)}B_1 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{pmatrix} = \mathbf{rank}(B^{(k)} \quad A^{(k)}B^{(k)}) = r_1 + r_2,$$

(c)

$$(A^{(k)})^2B^{(k)} = \begin{pmatrix} X \\ X \\ A_{3,2}(A_{2,1}B_1) \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

(the $^{(k)}$ superscripts are omitted), where, as in (b), $A_{3,2}(A_{2,1}B_1)$ has full row rank r_3 . Therefore,

$$\mathbf{rank}\begin{pmatrix} B_1 & X & X \\ 0 & A_{2,1}B_1 & X \\ 0 & 0 & A_{3,2}(A_{2,1}B_1) \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \end{pmatrix} = \mathbf{rank}(B \ \ AB \ \ A^2B) = r_1 + r_2 + r_3.$$

(d) Continuing in this way, we see that

$$\begin{aligned} &\mathbf{rank}(B^{(k)} \ A^{(k)}B^{(k)} \ \cdots \ (A^{(k)})^{k-1}B^{(k)}) \\ &\quad = \mathbf{rank}(B \ AB \ \cdots \ A^{k-1}B) \\ &\quad = r_1 + r_2 + \ldots + r_k \end{aligned}$$

(e) Since $\bar{A}_{2,1}^{(k)}$ in $(D.5a)$ is zero, and $\bar{A}_{1,1}^{(k)}$ (the upper left block) has $r_1 + r_2 + \ldots + r_k$ rows, and $B^{(k)}$ has the form in $(D.5c)$, only the first $r_1 + \ldots + r_k$ rows of $(A^{(k)})^k B^{(k)}$ will be nonzero, and indeed this will be true also for all higher powers of $A^{(k)}$. Hence

$$\begin{aligned}
&\mathbf{rank}(B \;\; AB \;\; \cdots \;\; A^{n-1}B) \\
&\quad = \mathbf{rank}\big(B^{(k)} \;\; A^{(k)}B^{(k)} \;\; \cdots \;\; (A^{(k)})^{n-1}B^{(k)}\big) \\
&\quad = r_1 + r_2 + \ldots + r_k
\end{aligned}$$

where n is the size of the original matrix A. By the Cayley Hamilton Theorem, we need not go any further than $n - 1$. $$$

We illustrate the algorithm step by step with the following 4 by 4 example:

$$\dot{\mathbf{x}} = \begin{pmatrix} -\frac{1}{2} & 0 & \frac{5}{2} & 0 \\ -\sqrt{2} & -1 & \frac{8}{\sqrt{2}} & 0 \\ -\frac{3}{2} & 0 & \frac{7}{2} & 0 \\ \frac{1}{\sqrt{2}} & -1 & \frac{3}{\sqrt{2}} & -2 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \mathbf{u}. \qquad (sample)$$

After stage 0., the B matrix has been collapsed to a part of full row rank plus zeroes, giving the result

$$\dot{\mathbf{x}}^{(0)} = \begin{pmatrix} -2.000 & -0.354 & -5.500 & -0.354 \\ -0.707 & 0 & 0 & 1.000 \\ 0 & 1.061 & 3.500 & 1.061 \\ 0.707 & -0.500 & -3.536 & -1.500 \end{pmatrix} \mathbf{x}^{(0)} + \begin{pmatrix} -1.414 \\ 0 \\ 0 \\ 0 \end{pmatrix} \mathbf{u}. \qquad (sample.s0)$$

After stage 1 of the reduction ($C/\bar{C}$ split) of the matrix A we have

$$\dot{\mathbf{x}}^{(1)} = \begin{pmatrix} -2.000 & 0 & -5.500 & -0.500 \\ 1.000 & -1.000 & -2.500 & -1.500 \\ 0 & {}^{*}0 & 3.500 & 1.500 \\ 0 & {}^{*}0 & -2.500 & -0.500 \end{pmatrix} \mathbf{x}^{(1)} + \begin{pmatrix} -1.414 \\ 0 \\ 0 \\ 0 \end{pmatrix} \mathbf{u}. \qquad (sample.s1)$$

The *-ed entries are all zero, so that this is all there is to this reduction. In practice, in noting the zeroes, the method goes through an extra iteration, resulting in a change to this last result:

$$\dot{\mathbf{x}}^{(2)} = \left(\begin{array}{cc|cc} -2.000 & 0 & 4.075 & -3.727 \\ 1.000 & -1.000 & 1.082 & -2.707 \\ \hline 0 & 0 & 2.516 & -3.793 \\ 0 & 0 & .206 & .484 \end{array}\right) \mathbf{x}^{(2)} + \left(\begin{array}{c} -1.414 \\ 0 \\ \hline 0 \\ 0 \end{array}\right) \mathbf{u}. \qquad (sample.s2)$$

It is this last result which is used in subsequent computations.

To further illustrate this method, we give an 11 by 11 example, with randomly chosen eigenvalues. For the original matrix A in the system (*start*), we use

$$\begin{pmatrix}
-0.292 & 0.518 & 0.263 & 0.906 & -0.143 & 0.189 & 0.467 & 1.598 & 0.905 & 0.530 & 0.076 \\
-0.148 & 0.308 & 0.859 & 0.341 & 0.970 & 0.552 & -0.016 & 1.216 & 0.957 & 0.770 & -0.093 \\
0.807 & 0.705 & 0.730 & 0.640 & 0.269 & 0.615 & 0.830 & -0.366 & 0.624 & 0.313 & 1.009 \\
0.141 & 1.305 & -0.153 & 1.208 & -0.778 & -0.499 & 1.392 & 0.109 & 0.176 & 0.649 & 0.510 \\
-0.048 & 0.896 & 0.035 & 0.474 & -0.309 & 0.178 & 1.210 & 0.199 & 0.770 & 0.448 & 0.395 \\
1.183 & 0.422 & 0.232 & 0.328 & 0.933 & 0.093 & 0.496 & -0.389 & -0.403 & 1.410 & 0.591 \\
1.118 & 1.188 & 0.083 & 1.053 & -0.059 & 0.390 & 1.562 & -0.272 & 0.474 & 0.477 & 0.940 \\
0.598 & 0.025 & 0.121 & 0.249 & 0.376 & 0.113 & -0.064 & 0.646 & 0.149 & 0.220 & 0.257 \\
-0.161 & 0.247 & 0.546 & 0.475 & 0.487 & 0.154 & 0.127 & 1.416 & 0.781 & 0.585 & 0.086 \\
1.351 & -0.237 & 0.993 & -0.138 & 0.718 & 1.557 & -0.257 & 0.577 & 0.731 & 0.183 & 0.122 \\
0.850 & -0.182 & 1.164 & 0.127 & 1.543 & 0.460 & -0.152 & 1.133 & 0.410 & 0.791 & 0.638
\end{pmatrix}. \qquad (eleven.a)$$

The starting values for the input vectors are

$$\begin{pmatrix}
0.558 & 0.995 & -1.433 & -0.075 \\
-0.964 & 0.501 & -2.963 & 0.739 \\
1.022 & -0.367 & 2.727 & -0.069 \\
2.551 & 0.063 & 1.243 & -0.422 \\
0.831 & 0.491 & 1.115 & 0.041 \\
-0.270 & -0.957 & 0.206 & -0.011 \\
2.722 & -0.637 & 3.297 & -0.593 \\
-0.140 & -0.461 & 0.836 & -0.047 \\
-0.328 & 0.584 & -2.275 & 0.254 \\
-1.208 & -0.135 & 0.011 & 0.152 \\
-0.893 & -0.185 & -2.903 & 0.089
\end{pmatrix}. \qquad (eleven.b)$$

After the algorithm has been applied, the final system has the form

$$\left(\begin{array}{cccc|ccc|cccc}
-0.762 & 1.799 & -0.798 & 0.261 & 0.457 & -1.362 & -0.509 & 1.038 & 1.394 & 0.978 & 0.617 \\
-0.013 & 1.596 & -1.097 & -1.184 & -0.792 & -2.069 & 0.339 & -0.350 & 0.074 & -1.305 & -0.589 \\
0.022 & -0.487 & -0.761 & -0.129 & 1.887 & -0.230 & -0.280 & 0.071 & -0.612 & -0.429 & -1.160 \\
-0.006 & -0.663 & 0.290 & 1.109 & 1.579 & 0.063 & -0.206 & -0.135 & 0.054 & 0.142 & 0.385 \\
\hline
-0.008 & -1.325 & 0.645 & 1.224 & 3.612 & -0.988 & 0.384 & 0.545 & 0.120 & 0.791 & 0.628 \\
-0.024 & 0 & 0.089 & 0.067 & 0.029 & -1.201 & 0.527 & 0.166 & 0.349 & -0.223 & 0.314 \\
-0.039 & 0 & 0 & 0.057 & 0.126 & -0.304 & -0.240 & 0.321 & 0.360 & 0.179 & 0.087 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.517 & 0.229 & 0.026 & 0.025 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.084 & 0.636 & 0.007 & -0.136 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.269 & 0 & 0 & 0.276 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.266 & -0.124 & 0.165 & 0.162
\end{array}\right)$$

The input vectors B, in their final reduced form, are

$$\left(\begin{array}{cccc}
3.097 & -0.994 & 6.802 & -0.821 \\
3.123 & 0.663 & 0 & -0.409 \\
0 & 1.456 & 0 & 0.200 \\
0 & 0 & 0 & -0.553 \\
\hline
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{array}\right)$$

Sensitivity Analysis.

The Staircase Algorithm can give a rough estimate of the sensitivity of the results to small perturbations in the original data. Since the algorithm uses only orthogonal transformations, the resulting system (*D*.5) will be similar to a system that differs from the original system (*start*) by only a small multiple of the machine precision [Wilkinson], as long as the ranks of the subdiagonal blocks are computed correctly. If the ranks are not computed correctly, then the partitions will fall in the wrong place, and the computed values for the dimension of the controllable space will be off. Once a rank error is made in a block, it is propagated throughout the remaining stages.

Given any system, a small random perturbation will most likely cause the system to become completely controllable. After we have reduced the system by the Staircase Algorithm, the controllability matrix looks like

$$(B \;\; AB \;\; A^2B) = \begin{pmatrix} B_1 & X & X \\ 0 & A_{2,1}B_1 & X \\ 0 & 0 & A_{3,2}(A_{2,1}B_1) \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \end{pmatrix}$$

where only the first r rows are nonzero, r being the dimension of the controllable space S_c. A small perturbation to either A or B will introduce nonzeroes below row r, signifying a controllable space of larger dimension. Our problem is to decide what remains controllable through all such small perturbations, or equivalently find the smallest S_c we can obtain over all such perturbations. Specifically, if we allow perturbations up to size TOL, we would like to determine what is the smallest S_c we can find within such a tolerance TOL. These comments apply not only to the Staircase Algorithm, but also to any other method used to compute the controllable space, including the one given in the next section.

Computing the rank of a sub-block involves computing the singular values of that block. If we decide that all the singular values below a certain value TOL $\equiv \sigma_{r+1}$ are 0, where r is the computed rank, then we have essentially perturbed the matrix system by the amount TOL. These perturbations will depend on the tolerance implied in the input data, and, in general, they will be much greater than the machine precision. The relevant measure of sensitivity to perturbations would thus appear to be based on the collection of the smallest singular values for each sub-block. However, these individual values are not good indications of bad conditioning, as is shown in the following example:

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & 0 \\ \sqrt{\epsilon} & 1 \end{pmatrix}\mathbf{x} + \begin{pmatrix} \sqrt{\epsilon} \\ b_2 \end{pmatrix}\mathbf{u},$$

where $b_2 = 0$. The only singular value for each subdiagonal block (we must include the block b_1 from stage 0.) is large (on the order of $\sqrt{\epsilon}$), so it appears the controllable space has dimension 2. But if we set $\tilde{b}_2 = -\epsilon$, a perturbation of size ϵ, then $\tilde{\mathbf{b}}$ becomes an eigenvector of A, and the controllable space is reduced to dimension 1. However, the *product* of these small singular values is a more reliable measure, meaning that if the product is large then there is no perturbation that

will give rise to a different answer. In the multiple input case this is just an empirical observation, but for the single input case, we can actually prove the following theorem:

Theorem 2. (*StairMeasure*)

Consider the system

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}. \qquad (system)$$

Suppose B consists of one column, and that the system is in the Staircase form, which in this case is equivalent to saying that $B = b_1\mathbf{e}_1$ and A upper Hessenberg, where $\mathbf{e}_1 = (1 \quad 0 \quad \dots \quad 0)^{\mathrm{T}}$. Suppose further that

$$\|A\|_2 + \|B\|_2 \leq \frac{1}{4}$$

and that the product of the subdiagonal elements satisfy

$$|b_1 a_{2,1} \dots a_{n,n-1}| \equiv \mu_s > \frac{\epsilon}{4}$$

with $\epsilon < \frac{1}{4}$.

Then *the system (system) is completely controllable within any ϵ perturbation – i.e. if the matrices A and B are subjected to any ϵ perturbation, the system remains controllable.*

Proof:

Let

$$F(\lambda) = (B \quad \lambda I - A).$$

If the system were almost uncontrollable, then $F(\lambda_i)$ would have to be almost row-rank deficient, for λ_i an eigenvalue of A. So we need only look at $\lambda = \lambda_i$. For any such λ_i, $|\lambda_i| \leq \|A\|_2 \leq \frac{1}{4}$. If A is perturbed by ϵ, then its eigenvalues will still satisfy

$$|\lambda| \leq \|A + \epsilon E\|_2 \leq \frac{1}{4} + \epsilon, \qquad (\lambda bound)$$

where $\|E\| \leq 1$. We will prove that $F(\lambda)$ has full row-rank for all λ satisfying this last condition.

Augment $F(\lambda)$ to

$$\bar{F}(\lambda) = \begin{pmatrix} F(\lambda) \\ \frac{1}{4}\mathbf{e}_{n+1} \end{pmatrix},$$

a square $n+1$ by $n+1$ upper triangular matrix. Using the Triangle Inequality, we have that

$$\left\|\bar{F}(\lambda)\right\|_2 \leq \frac{1}{4} + (\|A\|_2 + \|B\|_2) + |\lambda| \leq 1,$$

for any λ satisfying ($\lambda bound$).

Let us pick a particular value for λ satisfying ($\lambda bound$), and let $G = \bar{F}(\lambda)$. Observe that

$$\det(G) = \left|b_1 a_{2,1} \dots a_{n,n-1} \frac{1}{4}\right| > \epsilon.$$

Consider the S.V.D. of $G = U\Sigma V^{\mathrm{T}}$. Then

$$\epsilon < |\det(G)| = \left|\det(U)\det(\Sigma)\det(V^{\mathrm{T}})\right| = |\det(\Sigma)|,$$

since for unitary matrices, the determinant has absolute value 1. If we assume that the singular values are ordered, then we can write

$$1 \geq \|G\|_2 = \sigma_1 \geq \ldots \geq \sigma_{n+1} \geq 0.$$

Since $\sigma_1 \ldots \sigma_n \leq 1$, it follows that

$$\epsilon < |\det(\Sigma)| = (\sigma_1 \ldots \sigma_n)\sigma_{n+1} \leq \sigma_{n+1}$$

Thus, for any λ satisfying (λ*bound*), there is no ϵ-perturbation of $G = \bar{F}(\lambda)$ that will make it singular.

In order to show that any ϵ-perturbation of A, B is completely controllable, we must show that the resulting perturbed $\tilde{F}(\tilde{\lambda})$ corresponding to $\bar{F}(\lambda)$, using a perturbed eigenvalue $\tilde{\lambda}$, is still non-singular. But any such eigenvalue $\tilde{\lambda}$ still satisfies (λ*bound*), and any such perturbed $\tilde{F}(\tilde{\lambda})$ is not more than an ϵ-perturbation of $\bar{F}(\tilde{\lambda})$ coming from the original A, B. Hence $\tilde{F}(\tilde{\lambda})$ cannot be singular. $$$

Following the conjecture given before the theorem, we can define our sensitivity to be the product μ_s of the smallest singular values of the sub-diagonal blocks in ($D.5a$). Note that, although this bound is robust, it has severe limitations. For one, it is a function of the determinant of the augmented matrix $\bar{F}$ and hence can be very sensitive to perturbations in the data. The following example is a case in point

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} -\frac{1}{2} & 0 \\ -\sqrt{\epsilon} & -\frac{1}{2} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} \sqrt{\epsilon} \\ 0 \end{pmatrix} u.$$

It is shown below that though the product of the subdiagonal elements is small (equal to ϵ), there does not exist an ϵ-perturbation to this system with a controllable part of dimension 1. Furthermore, in the multiple input case, the bound is only a heuristic, but one which seems to work as well as in the single input case. It will be seen in Chapter 4 that this measure can be, and very often is, extremely pessimistic.

We now prove the assertion concerning the above example.

Proposition.

If $\epsilon < .04$ then no ϵ-perturbation of the system $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}$ with

$$A = \begin{pmatrix} -\frac{1}{2} & 0 \\ -\sqrt{\epsilon} & -\frac{1}{2} \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} \sqrt{\epsilon} \\ 0 \end{pmatrix}.$$

has a controllable part of dimension 1.

Proof:

We consider what happens if we apply a small perturbation E to A and F to B, where we denote

$$E = \begin{pmatrix} \varsigma_1 & \varsigma_2 \\ \varsigma_3 & \varsigma_4 \end{pmatrix} \quad \text{and} \quad F = \begin{pmatrix} \varsigma_5 \\ \varsigma_6 \end{pmatrix},$$

and the elements of E and F satisfy $|\varsigma_i| \leq \frac{\sqrt{\epsilon}}{3}$, $i = 1, \ldots, 6$. We will show that under any such perturbation, the controllable space S_c will have dimension 2.

The rank of $B + F$ is 1 for any such F so we may construct an orthogonal rotation so that

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix}(B + F) = \begin{pmatrix} z \\ 0 \end{pmatrix}, \qquad (rotate)$$

for some $z \neq 0$. This yields the relations

$$c(\sqrt{\epsilon} + \varsigma_5) + s\varsigma_6 = z$$
$$-s(\sqrt{\epsilon} + \varsigma_5) + c\varsigma_6 = 0,$$

from which it follows that $c \neq 0$ and

$$\frac{s}{c} = \frac{-\varsigma_6}{\sqrt{\epsilon} + \varsigma_5}.$$

Hence

$$\left|\frac{s}{c}\right| \leq \frac{1}{2}.$$

Since we have an orthogonal rotation, we may use $s^2 + c^2 = 1$ to obtain a bound on c:

$$c^2 \geq \frac{4}{5}.$$

We apply the rotation used in (*rotate*) to A to get

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix}(A + E)\begin{pmatrix} c & -s \\ s & c \end{pmatrix} = \begin{pmatrix} x & x \\ -y & x \end{pmatrix}.$$

In order to show $\dim S_c = 2$, it suffices to show that $|y| > \epsilon$. But from the last equation, we have the following expression for y:

$$\begin{aligned} y &= c^2(-\sqrt{\epsilon} + \varsigma_3) - s^2\varsigma_2 + cs(\varsigma_4 - \varsigma_1) \\ &= c^2\left(-\sqrt{\epsilon} + \varsigma_3 - \frac{s^2}{c}\varsigma_2 + \frac{s}{c}(\varsigma_4 - \varsigma_1)\right) \end{aligned}$$

Thus we obtain the following estimate for $|y|$:

$$\begin{aligned} |y| &\geq c^2\left(\frac{2}{3}\sqrt{\epsilon} - \frac{1}{12}\sqrt{\epsilon} - \frac{1}{3}\sqrt{\epsilon}\right) \\ &\geq \frac{1}{4}c^2\sqrt{\epsilon} \geq \frac{1}{5}\sqrt{\epsilon} > \epsilon. \end{aligned}$$

Hence we may conclude that we would have to perturb the system by at least order $\sqrt{\epsilon}$ in order to achieve a controllable space of dimension 1. $$$

For the 4 by 4 example (*sample*), the computed measure is

$$\mu_s = 7.44 \times 10^{-3},$$

and for the 11 by 11 example, it is

$$\mu_s = 5.05 \times 10^{-4}.$$

In these examples, it was necessary to scale the matrices to be of norm unity in order to satisfy the requirements for the Theorem.

Section II B. The Modal Method

In this section we describe another way to compute the controllable decomposition, as an alternative to the Staircase Algorithm. For the scheme to work, the eigenvalues of A must be distinct, and for numerical stability, we must also ask that they be well separated. This method has the advantage of showing explicitly whether a perturbation of order ϵ in the A and B matrix will change the dimension of the computed controllable space.

First we discuss perturbations in B. We start with the system

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u}, \qquad (start)$$

where, as noted above, A has distinct and well-separated eigenvalues. Let

$$A = Y\Lambda Y^{-1} \qquad (eigen)$$

be the eigen-decomposition of A, where Y is the matrix of eigenvectors, and $\Lambda = \mathbf{diag}(\lambda_1, \ldots, \lambda_n)$ is the matrix of eigenvalues. If we set $X = Y^{-1}B$, $\mathbf{z} = Y^{-1}\mathbf{x}$ then we can write (*start*) as

$$\dot{z} = \Lambda\mathbf{z} + X\mathbf{u}.$$

Since (*eigen*) is the unique decomposition of n-space into invariant spaces of A, the controllable space S_c is the span of exactly those columns of Y which correspond to non-zero rows of X. This should be clear if one recalls that the controllable space is just $\mathbf{range}(B \;\; AB \;\; A^2B \;\; A^3B \;\; \ldots)$.

We would like to compute what $O(\epsilon)$ changes to B will cause a whole row of X to become zero. If we let $U\Sigma V^{\mathrm{T}}$ be the S.V.D. of Y, then we can write

$$\begin{aligned} B &= YX \\ &= U\Sigma V^{\mathrm{T}}X \end{aligned}$$

or

$$U^{\mathrm{T}}B = \Sigma V^{\mathrm{T}}X. \qquad (stretch)$$

To zero the i-th row of X, we must subtract from it a matrix whose i-th row is the same as that of X. One such matrix is

$$\Delta_i = \frac{1}{v_{ij}}\mathbf{v}_j\mathbf{x}_i^T \qquad (delta)$$

where v_{ij} denotes the ij-th element of V, $\mathbf{v}_j$ the i-th column of V, $\mathbf{x}_i^T$ the i-th row of X, and j is any index such that $v_{ij} \neq 0$.

Since U is orthogonal, an ϵ change in B corresponds to an ϵ change in $U^{\mathrm{T}}B$. Hence it is enough to find out how $U^{\mathrm{T}}B$ is affected when the correction Δ_i is applied to X. The change in $U^{\mathrm{T}}B$ resulting from the zeroing of the row $\mathbf{x}_i^T$ is, from (*stretch*),

$$\begin{aligned} \text{change in } U^{\mathrm{T}}B &= \Sigma V^{\mathrm{T}}\Delta_i \\ &= \Sigma V^{\mathrm{T}}\mathbf{v}_j\mathbf{x}_i^T\frac{1}{v_{ij}} \\ &= \Sigma\mathbf{e}_j\mathbf{x}_i^T\frac{1}{v_{ij}} \end{aligned}$$

(where $\mathbf{e}_j$ is the j-th coordinate unit vector)

$$= \frac{\sigma_j}{v_{ij}}\mathbf{x}_i^T.$$

Therefore, a condition sufficient to guarantee that zeroing the i-th row of X results in only an ϵ change in B is

$$\epsilon \geq \|\Sigma V^{\mathrm{T}} \Delta_i\| = \frac{\sigma_j}{|v_{ij}|} \|\mathbf{x}_i^T\|,$$

which we can achieve exactly when

$$\|\mathbf{x}_i^T\| \leq \epsilon \frac{|v_{ij}|}{\sigma_j}. \qquad (bound)$$

Thus if we are given a problem for which this condition is satisfied for a certain value of i, then the corresponding row of X should be zeroed, and the corresponding eigenvector in Y should be considered *not* part of the controllable space. We can formalize this in the following theorem

Theorem 3.

The bound (bound) is sufficient to guarantee that there is an ϵ- perturbation of B that will result in a smaller controllable space. Conversely, if the bound

$$\|\mathbf{x}_i^T\| \geq \frac{\epsilon}{\sigma_j}.$$

is satisfied for all j, then this is sufficient to guarantee that there is no such small perturbation.

Proof:

Suppose $\mathbf{x}_i^T$ satisfies

$$\|\mathbf{x}_i^T\| = k\epsilon \frac{|v_{ij}|}{\sigma_j}$$

for some constant k. Then the change in B using (*delta*) would be

$$\begin{aligned} \Delta B &= U\Sigma V^{\mathrm{T}} \Delta_i \\ &= U\sigma_j \frac{1}{v_{ij}} \mathbf{x}_i^T. \end{aligned}$$

Taking norms gives

$$\begin{aligned} \|\Delta B\| &= \sigma_j \frac{1}{|v_{ij}|} k\epsilon \frac{|v_{ij}|}{\sigma_j} \\ &= k\epsilon. \end{aligned}$$

If (*bound*) is satisfied, then $k \leq 1$, hence we have exhibited an ϵ- perturbation to B that would zero a row of X. As for the lower bound, we note that of all matrices that would zero a row of X, the one with smallest norm would be

$$\Delta_\epsilon = \mathbf{e}_j \mathbf{x}_i^T,$$

whose norm satisfies

$$\|\Delta_\epsilon\| = \|\mathbf{x}_i^T\| = |v_{ij}| \|\Delta_i\|.$$

Hence, if $k \geq |v_{ij}| \geq 1$ then the change in B using Δ_ϵ would be

$$\begin{aligned} \|\Delta_\epsilon B\| &= \|U\Sigma V^{\mathrm{T}} \mathbf{e}_j \mathbf{x}_i^T\| \\ &\geq \sigma_{min} \|V^{\mathrm{T}} \mathbf{e}_j \mathbf{x}_i^T\| \\ &\geq \sigma_{min} \frac{\epsilon}{\sigma_j}. \end{aligned}$$

Since this is true for all j, the theorem follows. $$$

The algorithm that is derived from the above goes as follows:

Modal-B

(a) **compute** Y, the eigenvectors of A.

(b) **set** $X = Y^{-1}B$.

(c) **compute** $U\Sigma V^{\mathrm{T}} = \mathrm{SVD}(X)$.

(d) **for** $k = 1, \ldots, n$ **do**

(e) set $d_k = \left\| \mathbf{x}_k^{\mathrm{T}} \right\|$, the norm of the i-th row of X.

(f) **for** $i, j = 1, \ldots, n$ **do**

(g) **if** (*bound*) satisfied

(h) **then set** $d_k = 0$, and skip to next row k.

(i) **else let** $\hat{\epsilon} = d_k \sigma_n$, i.e. the smallest perturbation of B needed to zero out row k.

(j) set matrix $D = \mathrm{diag}(d_1, \ldots, d_n)$.

(k) **compute** $(Q_1\ Q_2)\Sigma[V]^{\mathrm{T}} = \mathrm{SVD}(YD)$, where the columns Q_1 correspond to the non-zero singular values. (The result is that Q_1 is an orthonormal basis for the space $\mathbf{span}(Q_1) \equiv \mathbf{span}(YD) \equiv S_c$.)

(l) **compute** $A^{(1)} = (Q_1\ Q_2)^{\mathrm{T}} A (Q_1\ Q_2)$. (We transform to a new coordinate system, in which $A^{(1)}$ is block upper triangular.)

(m) **define** the measure $\mu_B \equiv \min(\hat{\epsilon}_k)$, where we take the minimum over all k such that $d_k \neq 0$.

We also have a similar approach to estimate the effects of perturbations in the matrix A. Unfortunately, the bound for this case is *very* pessimitic, as can be seen from the results given in Chapter 4. We define the separation

$$\delta \equiv \left\| T^{-1} \right\|^{-1}, \qquad (lyapunov.a)$$

where, following [Stewart 1973b], the map T is defined in terms of A as

$$TX \equiv A_{11}^{(1)} X - X A_{22}^{(1)}. \qquad (lyapunov.b)$$

We need the following lemma.

Lemma 1.

Let X, B, A, Y, d_k, D, and δ be defined as above. Furthermore, let $Q = (Q_1\ Q_2)$ be an orthogonal matrix with $\mathbf{span}(Q_1) = S_c$. Then

(a) $\mathbf{span}(YD) = S_c = \mathbf{span}(Q_1)$. Hence, there is a matrix S such that $YD = Q_1 S$.

(b) If $\tilde{A} = A + \epsilon E$ is an ϵ perturbation of A, then there exists a $\tilde{P}$ such that $(I + \tilde{P})YD$ is an invariant subspace of $\tilde{A}$, where $\left\| \tilde{P} \right\| \leq \frac{2}{\delta}\epsilon$.

Proof.

(a) follows immediately. Note that $\|S\| = \|YD\|$ in the 2-norm.

(b) By theorem 4.11 of [Stewart 1973b], there is a P satisfying $\|P\| \leq \frac{3}{\delta}\epsilon$ such that $\tilde{Q}_1 = Q_1 + Q_2 P$ spans an invariant subspace of $\tilde{A}$. We can then write

$$\begin{aligned}\tilde{Q}_1 S &= Q_1 S + Q_2 P S \\ &= YD + \tilde{P} YD\end{aligned}$$

where $\tilde{P} = Q_2 P Q_1^{\mathrm{T}}$, since $S = Q_1^{\mathrm{T}} YD$.. $$$

For the purposes of this section, we define the condition number $\kappa(M)$ of a possibly singular matrix M to be the quantity $\kappa(M) = \frac{\sigma_{max}}{\sigma_\epsilon}$, where σ_ϵ is the smallest non-zero singular value. In the case where M is non-singular, this reduces to the usual definition, in the 2-norm. Then, it follows that in order to lower the rank of YD by a relative perturbation $\tilde{P}$, $\tilde{P}$ must satisfy [Stewart 1973a]

$$\left\|\tilde{P}\right\| \geq \frac{1}{\kappa(YD)} \geq \frac{1}{\kappa(Y)\kappa(D)}.$$

Using part (b) of Lemma 1 and solving for ϵ, we get the following measure of the sensitivity for perturbations in A.

$$\mu_A \equiv \frac{\delta}{2}\frac{1}{\kappa(Y)\kappa(D)}. \qquad (\mu.A)$$

The actual method used to compute the quantity μ_A involves finding estimates to all the items in the formula $(\mu.A)$. Computing $\kappa(D)$ is trivial; the singular values of a non-negative diagonal matrix are just the diagonal elements. So

$$\kappa(D) = \frac{d_{max}}{d_{min}}.$$

We scale Y so that $\|Y\| = 1$, and estimate $\left\|Y^{-1}\right\|$ from the growth involved in the solution of $X = Y^{-1}B$ in step (b) of algorithm *Modal-B* [Cline et al]. This is approximated by the quantity $\|X\| \approx \|D\|$, given that B is scaled to approximately norm 1. We then use the estimate

$$\kappa(Y) \approx d_{max}.$$

The quantity δ defined in (*lyapunov*) is computed by solving the system $T(X) = Z$ with a random right hand side Z, and estimating $\left\|T^{-1}\right\|$ as the growth in the solution. A variation of this procedure is used, which is based on the method of [Cline et al]. It is rather expensive, taking at least 25%of the running time.

Combining all these estimates, we arrive at the formula

$$\mu_A = \frac{d_{min}}{d_{max}^2 \delta}.$$

We can illustrate the algorithm with the following 4 by 4 example:

$$\dot{\mathbf{x}} = \begin{pmatrix} -\frac{1}{2} & 0 & \frac{5}{2} & 0 \\ -\sqrt{2} & -1 & \frac{8}{\sqrt{2}} & 0 \\ -\frac{3}{2} & 0 & \frac{7}{2} & 0 \\ \frac{1}{\sqrt{2}} & -1 & \frac{3}{\sqrt{2}} & -2 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \mathbf{u}. \qquad (sample)$$

The matrix A has the eigenvectors (all numbers given only to 3 decimal places)

$$Y = \begin{pmatrix} 0 & 0 & 0.492 & 0.639 \\ 0 & 0.707 & 0.696 & 0.633 \\ 0 & 0 & 0.492 & 0.383 \\ 1.000 & -0.707 & 0.174 & 0.211 \end{pmatrix},$$

which has a condition number of approximately 16.

The coefficients X, expressing the vector B as linear combinations of the eigenvectors Y, are

$$\begin{pmatrix} 2.000 \\ 1.414 \\ 0 \\ 0 \end{pmatrix}.$$

This is, of course, also the vector of row-norms $d_1, \ldots, d_4$ which form the diagonal of D.

The singular values σ_{ii} of Y from step (c) of *Modal-B* are to 4 places

$$(1.474 \quad 1.292 \quad 0.379 \quad 0.123),$$

and the right singular vectors V are

$$\begin{pmatrix} 0.017 & -0.749 & 0.659 & -0.070 \\ -0.383 & 0.606 & 0.687 & -0.119 \\ -0.660 & -0.166 & -0.095 & 0.727 \\ -0.647 & -0.209 & -0.293 & -0.673 \end{pmatrix}.$$

From these results, we determine that the controllable part is of dimension 2, and that the perturbation of B needed to reduce this further is

$$\mu_B = 1.74 \times 10^{-1},$$

which is certainly larger than the tolerance used: 10^{-7}.

The transformed system looks like

$$\dot{\mathbf{x}} = \left(\begin{array}{cc|cc} -1.500 & -1.207 & 0.205 & 1.194 \\ -0.207 & -1.500 & -6.038 & -1.036 \\ \hline 0 & 0 & 3.500 & 1.500 \\ 0 & 0 & -2.500 & -0.500 \end{array}\right)\mathbf{x} + \begin{pmatrix} -0.541 \\ -1.307 \\ 0 \\ 0 \end{pmatrix}\mathbf{u}, \qquad (sample.modal1)$$

and the transformation used to obtain this result is

$$(Q_1\ Q_2) = \left(\begin{array}{cc|cc} 0 & 0 & 0 & -1.000 \\ 0.383 & -0.924 & 0 & 0 \\ 0 & 0 & 1.000 & 0 \\ -0.924 & -0.383 & 0 & 0 \end{array}\right).$$

The map T defined in (*lyapunov*) is thus

$$T(X) \equiv \begin{pmatrix} -1.500 & -1.207 \\ -0.207 & -1.500 \end{pmatrix} X + X \begin{pmatrix} 3.500 & 1.500 \\ -2.500 & -0.500 \end{pmatrix},$$

whose inverse has norm

$$\delta^{-1} = .97\,.$$

Hence, the measure for the matrix A is computed to be

$$\mu_A = 6.03 \times 10^{-3}.$$

In the 11 by 11 example (*eleven*) of the previous section, the singular values of the eigenvector matrix Y are

$$\begin{pmatrix} 1.866 \\ 1.606 \\ 1.256 \\ 1.106 \\ 0.954 \\ 0.663 \\ 0.571 \\ 0.465 \\ 0.376 \\ 0.264 \\ 0.188 \end{pmatrix}$$

so that $\kappa(Y) \approx 10$.

The vector of row-norms of the matrix X, after the small entries have been zeroed according to (*bound*), are

$$\begin{pmatrix} 1.267 \\ 1.126 \\ 2.075 \\ 6.569 \\ 0.929 \\ 0 \\ 0 \\ 4.140 \\ 1.090 \\ 0 \\ 0 \end{pmatrix},$$

so that the controllable part has dimension 7.

The inverse of the map T has a norm of approximately 120, so that the two measures we get for this system are

$$\mu_B = 1.74 \times 10^{-2} \qquad \mu_A = 1.23 \times 10^{-6}.$$

We have here described two methods to compute the controllable part and defined the measures μ_s, μ_B and μ_A of the sensitivity of the results to perturbations in the coefficients of the original problem. Two examples, from actual computer runs, have been used to illustrate the methods.

Chapter III. Merging the Controllable and Observable Spaces

In the previous chapter, we described two methods to compute the controllable space S_c, and equivalently the observable space S_o. Once we have this information, we would like to compute the intersection of these two spaces and their complements. Specifically, we would like to compute the four spaces

$$\begin{aligned} S_{c\bar{o}} &\equiv S_c \bigcap S_{\bar{o}}, \\ S_{co} &\equiv S_c \bigcap S_o, \\ S_{\bar{c}\bar{o}} &\equiv S_{\bar{c}} \bigcap S_{\bar{o}}, \\ S_{\bar{c}o} &\equiv S_{\bar{c}} \bigcap S_o. \end{aligned}$$

Two methods are described in this chapter. The first is based on the idea of operating on the matrices themselves, hence the name Matrix Algorithm, and the second on the idea of operating on the spaces involved, hence the name State Space or Geometric Algorithm. In the first two sections of this chapter we describe the methods; in the last section we compare the methods, both from a theoretical and a numerical point of view.

Section III A. The Matrix Algorithm

The goal of this procedure is to convert the system

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u} \\ \mathbf{y} &= C\mathbf{x} \end{aligned} \qquad (start)$$

into the form

$$\begin{pmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \\ \dot{\mathbf{x}}_3 \\ \dot{\mathbf{x}}_4 \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ 0 & A_{22} & 0 & 0 \\ 0 & 0 & A_{33} & A_{34} \\ 0 & 0 & 0 & A_{44} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \end{pmatrix} + \begin{pmatrix} B_1 \\ B_2 \\ 0 \\ 0 \end{pmatrix} \mathbf{u}$$

$$\mathbf{y} = \begin{pmatrix} 0 & C_2 & 0 & C_4 \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \mathbf{x}_4 \end{pmatrix}$$

where the subsystem:

$$\begin{aligned} \begin{pmatrix} \dot{\mathbf{x}}_1 \\ \dot{\mathbf{x}}_2 \end{pmatrix} &= \begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} + \begin{pmatrix} B_1 \\ B_2 \end{pmatrix} \mathbf{u} \\ \mathbf{y} &= C\mathbf{x} \end{aligned}$$

is completely controllable; and the subsystem:

$$\begin{aligned} \begin{pmatrix} \dot{\mathbf{x}}_2 \\ \dot{\mathbf{x}}_4 \end{pmatrix} &= \begin{pmatrix} A_{22} & 0 \\ 0 & A_{44} \end{pmatrix} \begin{pmatrix} \mathbf{x}_2 \\ \mathbf{x}_4 \end{pmatrix} + \begin{pmatrix} B_2 \\ B_4 \end{pmatrix} \mathbf{u} \\ \mathbf{y} &= \begin{pmatrix} C_2 & C_4 \end{pmatrix} \begin{pmatrix} \mathbf{x}_2 \\ \mathbf{x}_4 \end{pmatrix} \end{aligned}$$

is completely observable. Once the system has been reduced to this form, the various parts can be read off by inspection.

The approach used in this section is to annihilate directly the appropriate entries in the matrices A, B, C. This algorithm depends on the eigenvalues being distinct. To review, we use the definition that an n-by-n system

$$\begin{aligned} \dot{\mathbf{x}} &= F\mathbf{x} + G\mathbf{u} \\ \mathbf{y} &= H\mathbf{x} \end{aligned} \qquad (prototype)$$

is completely controllable if $\mathbf{rank}(G \;\; FG \;\; F^2G \;\; \ldots) = n$, and is completely observable if

$$\mathbf{rank}\begin{pmatrix} H \\ HF \\ HF^2 \\ \vdots \end{pmatrix} = n$$

[Kalman]. These ranks are independent of basis, so we may apply similarity transformations freely without changing the structure, subject to numerical stability. Hence we say the two systems are *equivalent* if one can be obtained from the other by applying a similarity transformation. As was noted in the introduction, computing the controllable or observable spaces by actually forming the matrices used to define them can be numerically very unstable.

To achieve this end, we apply the following algorithm.

Matrix Algorithm.

Step 0. We start with system (*start*)

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u} \\ \mathbf{y} &= C\mathbf{x}. \end{aligned} \qquad (S.0)$$

We illustrate the algorithm step by step with the following 4 by 4 example:

$$\dot{\mathbf{x}}^{(0)} = \begin{pmatrix} -\frac{1}{2} & 0 & \frac{5}{2} & 0 \\ -\sqrt{2} & -1 & \frac{8}{\sqrt{2}} & 0 \\ -\frac{3}{2} & 0 & \frac{7}{2} & 0 \\ \frac{1}{\sqrt{2}} & -1 & \frac{3}{\sqrt{2}} & -2 \end{pmatrix} \mathbf{x}^{(0)} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \mathbf{u} \qquad (sample)$$

$$\mathbf{y} = (-\sqrt{2} \quad 1 \quad 0 \quad 0)\mathbf{x}^{(0)}.$$

Step 1. Controllability

We compute the complete controllability decomposition of $(S.0)$ using orthogonal similarity transformations, such as in the Staircase Algorithm (q.v.), to obtain a system of the form:

$$\begin{aligned} \begin{pmatrix} \dot{\mathbf{x}}_1^{(1)} \\ \dot{\mathbf{x}}_2^{(1)} \end{pmatrix} &= \begin{pmatrix} A_{11}^{(1)} & A_{12}^{(1)} \\ 0 & A_{22}^{(1)} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1^{(1)} \\ \mathbf{x}_2^{(1)} \end{pmatrix} + \begin{pmatrix} B_1^{(1)} \\ 0 \end{pmatrix} \mathbf{u} \\ \mathbf{y} &= (C_1^{(1)} \; C_2^{(1)}) \begin{pmatrix} \mathbf{x}_1^{(1)} \\ \mathbf{x}_2^{(1)} \end{pmatrix} \end{aligned} \qquad (S.1)$$

where $\mathbf{rank}(B_1^{(1)} \;\; A_{11}^{(1)}B_1^{(1)} \;\; A_{11}^{(1)2}B_1^{(1)} \;\; \ldots) = \mathbf{size}(A_{11}^{(1)})$ (the notation $\mathbf{size}(M)$ stands for the number of rows/columns in a square matrix M).

After this step (using the Staircase Algorithm), our example will have the form

$$\dot{\mathbf{x}}^{(1)} = \left(\begin{array}{cc|cc} -2.000 & 0 & 4.075 & -3.727 \\ 1.000 & -1.000 & 1.082 & -2.707 \\ \hline 0 & 0 & 2.516 & -3.793 \\ 0 & 0 & .206 & .484 \end{array}\right)\mathbf{x}^{(1)} + \left(\begin{array}{c} -1.414 \\ 0 \\ \hline 0 \\ 0 \end{array}\right)\mathbf{u} \qquad (sample.m1)$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} -.707 & -.707 & .856 & 1.125 \end{array}\right)\mathbf{x}^{(1)}.$$

The controllable part corresponds to the 1-1 (upper left) block of A, and thus has dimension 2.

Step 2. Observability in the Controllable Part

We compute the complete observability decomposition of the controllable part of $(S.1)$

$$\dot{\mathbf{x}}_1^{(1)} = A_{11}^{(1)}\mathbf{x}_1^{(1)} + B_1^{(1)}\mathbf{u}$$
$$\mathbf{y} = C_1^{(1)}\mathbf{x}_1^{(1)}$$

using orthogonal transformations, employing again the procedure used in Step 1. on $A_{11}^{(1)T}, C_1^{(1)T}$, but indexing the matrices in reverse. The entire system will then have the form

$$\left(\begin{array}{c} \dot{\mathbf{x}}_1^{(2)} \\ \hline \dot{\mathbf{x}}_2^{(2)} \\ \hline \dot{\mathbf{x}}_3^{(2)} \end{array}\right) = \left(\begin{array}{cc|c} A_{11}^{(2)} & A_{12}^{(2)} & A_{13}^{(2)} \\ 0 & A_{22}^{(2)} & A_{23}^{(2)} \\ \hline 0 & 0 & A_{33}^{(2)} \end{array}\right)\left(\begin{array}{c} \mathbf{x}_1^{(2)} \\ \hline \mathbf{x}_2^{(2)} \\ \hline \mathbf{x}_3^{(2)} \end{array}\right) + \left(\begin{array}{c} B_1^{(2)} \\ \hline B_2^{(2)} \\ \hline 0 \end{array}\right)\mathbf{u} \qquad (S.2)$$

$$\mathbf{y} = \left(\begin{array}{cc|c} 0 & C_2^{(2)} & C_3^{(2)} \end{array}\right)\left(\begin{array}{c} \mathbf{x}_1^{(2)} \\ \hline \mathbf{x}_2^{(2)} \\ \hline \mathbf{x}_3^{(2)} \end{array}\right)$$

where the solid line marks the split carried over from the previous stage $(S.1)$. (Note that we renumber the subblocks as the matrix becomes split further and further.)

At the end of this step, our example will have the form

$$\dot{\mathbf{x}}^{(2)} = \left(\begin{array}{cc|cc} -2.000 & -1.000 & -2.117 & .721 \\ 0 & -1.000 & -3.647 & 4.550 \\ \hline 0 & 0 & 2.516 & -3.793 \\ 0 & 0 & .206 & .484 \end{array}\right)\mathbf{x}^{(2)} + \left(\begin{array}{c} 1.000 \\ 1.000 \\ \hline 0 \\ 0 \end{array}\right)\mathbf{u} \qquad (sample.m2)$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} 0 & 1.000 & .856 & 1.125 \end{array}\right)\mathbf{x}^{(2)}.$$

Only the upper half of A and B have been affected, and C has been reduced to the form in $(S.2)$. Up to this point, we have applied only orthogonal transformations to the system

Step 3. Decoupling

We must now decouple the uncontrollable part (A_{33} in ($S.2$)) from the controllable-observable part (A_{22} in ($S.2$)). To do this, we compute a similarity transformation S of the form

$$S = \begin{pmatrix} I & 0 & 0 \\ 0 & I & T \\ 0 & 0 & I \end{pmatrix} \qquad (S.3a)$$

where the blocks are split as in (S.2), such that when applied in the manner

$$A \leftarrow SAS^{-1}, B \leftarrow SB, C \leftarrow CS^{-1}$$

we obtain a system equivalent to the original one ($S.0$) having the form

$$\left(\begin{array}{c} \dot{\mathbf{x}}_1^{(3)} \\ \hline \dot{\mathbf{x}}_2^{(3)} \\ \hline \dot{\mathbf{x}}_3^{(3)} \end{array}\right) = \left(\begin{array}{cc|c} A_{11}^{(2)} & A_{12}^{(2)} & A_{13}^{(3)} \\ \hline 0 & A_{22}^{(2)} & 0 \\ \hline 0 & 0 & A_{33}^{(3)} \end{array}\right) \left(\begin{array}{c} \mathbf{x}_1^{(3)} \\ \hline \mathbf{x}_2^{(3)} \\ \hline \mathbf{x}_3^{(3)} \end{array}\right) + \left(\begin{array}{c} B_1^{(2)} \\ \hline B_2^{(2)} \\ \hline 0 \end{array}\right) \mathbf{u}$$

$$\mathbf{y} = \left(\begin{array}{cc|c} 0 & C_2^{(2)} & C_3^{(3)} \end{array}\right) \left(\begin{array}{c} \mathbf{x}_1^{(3)} \\ \hline \mathbf{x}_2^{(3)} \\ \hline \mathbf{x}_3^{(3)} \end{array}\right) \qquad (S.3b)$$

using the same split as in ($S.2$). The object of this step is to annihilate the (2,3) block of $A^{(2)}$.

It can be easily shown that if A_{22} and A_{33} do not have any eigenvalues that are equal, then T is the unique solution of the Lyapunov equation

$$-A_{22}^{(2)}T + TA_{33}^{(2)} = A_{23}^{(2)} \qquad (Lyapunov)$$

and that the only blocks in ($S.2$) that are changed by the application of S are just the blocks denoted in ($S.3b$) with the superscript $^{(3)}$. To guarantee that a solution exists, A_{22} and A_{33} must not have any eigenvalues that match, otherwise the operator on the left hand side of (*Lyapunov*) will be singular. But this is already ensured by our assumption of distinct eigenvalues.

In our example, the decoupling transformation S in ($S.3a$) is

$$S = \left(\begin{array}{c|c|cc} 1.000 & 0 & 0 & 0 \\ \hline 0 & 1.000 & 1.058 & -.361 \\ \hline 0 & 0 & 1.000 & 0 \\ 0 & 0 & 0 & 1.000 \end{array}\right)$$

where T is the (2,3) subblock (1.058 —.361). When applied to the result of step 2, this

transformation will annihilate elements $a_{2,3}$ and $a_{2,4}$ of A to give

$$\dot{\mathbf{x}}^{(3)} = \left(\begin{array}{cc|cc} -2.000 & -1.000 & -1.058 & .361 \\ 0 & -1.000 & 0 & 0 \\ \hline 0 & 0 & 2.516 & -3.793 \\ 0 & 0 & .206 & .484 \end{array}\right)\mathbf{x}^{(3)} + \left(\begin{array}{c} 1.000 \\ \hline 1.000 \\ 0 \\ \hline 0 \end{array}\right)\mathbf{u} \qquad (sample.m3)$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} 0 & 1.000 & -.202 & 1.486 \end{array}\right)\mathbf{x}^{(3)}.$$

This step does not yield any new information on the structure of the observable parts, but it is essential if we mean to compute the observability split in the uncontrollable part in isolation.

The condition number of this transformation can often yield some information in the ill-posedness of the original problem, or a failure of the eigenvalues to be sufficiently well separated, but, as will be seen later, it is not the most reliable indicator of such problems.

Step 4. Observability in the non-Controllable Part

We compute the Observability decomposition for the non-controllable part by using the very method of step 2., this time on the non-controllable part of $(S.3b)$

$$\dot{\mathbf{x}}_3^{(3)} = A_{33}^{(3)}\mathbf{x}_3^{(3)}$$
$$\mathbf{y} = C_3^{(3)}\mathbf{x}_3^{(3)}.$$

The result is

$$\left(\begin{array}{c} \dot{\mathbf{x}}_1^{(4)} \\ \hline \dot{\mathbf{x}}_2^{(4)} \\ \dot{\mathbf{x}}_3^{(4)} \\ \dot{\mathbf{x}}_4^{(4)} \end{array}\right) = \left(\begin{array}{cc|cc} A_{11}^{(2)} & A_{12}^{(2)} & A_{13}^{(3)} & A_{14}^{(3)} \\ 0 & A_{22}^{(2)} & 0 & 0 \\ \hline 0 & 0 & A_{33}^{(4)} & A_{34}^{(4)} \\ 0 & 0 & 0 & A_{44}^{(4)} \end{array}\right)\left(\begin{array}{c} \mathbf{x}_1^{(4)} \\ \hline \mathbf{x}_2^{(4)} \\ \mathbf{x}_3^{(4)} \\ \mathbf{x}_4^{(4)} \end{array}\right) + \left(\begin{array}{c} B_1^{(2)} \\ \hline B_2^{(2)} \\ 0 \\ \hline 0 \end{array}\right)\mathbf{u} \qquad (S.4)$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} 0 & C_2^{(2)} & 0 & C_4^{(4)} \end{array}\right)\left(\begin{array}{c} \mathbf{x}_1^{(4)} \\ \hline \mathbf{x}_2^{(4)} \\ \mathbf{x}_3^{(4)} \\ \mathbf{x}_4^{(4)} \end{array}\right)$$

where the superscripts indicate whether each block was last changed in the step 2 or step 4. The transformation that is applied to the system $(S.3b)$ to achieve the above form is the block diagonal P:

$$P = \left(\begin{array}{cc|cc} I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ \hline 0 & 0 & P_3 & 0 \\ 0 & 0 & 0 & P_4 \end{array}\right).$$

The final form of our sample problem will be

$$\dot{\mathbf{x}}^{(4)} = \left(\begin{array}{cc|cc} -2.000 & -1.000 & -1.000 & .500 \\ 0 & -1.000 & 0 & 0 \\ \hline 0 & 0 & 2.000 & -4.000 \\ 0 & 0 & 0 & 1.000 \end{array}\right)\mathbf{x}^{(4)} + \left(\begin{array}{c} 1.000 \\ 1.000 \\ \hline 0 \\ 0 \end{array}\right)\mathbf{u} \qquad (sample.m4)$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} 0 & 1.000 & 0 & -1.500 \end{array}\right)\mathbf{x}^{(4)}.$$

Since this sample problem has only a single input and a single output, the controllable and observable parts can be determined by simple inspection of the final form (*sample.m4*), as can be easily seen by considering the rank of the controllability and observability matrices, respectively:

$$\mathbf{rank}(B \;\; AB \;\; A^2B \;\; \ldots), \quad \mathbf{rank}\begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \end{pmatrix}.$$

Theoretical Proof

Theorem 4. (*Matrix*)

If the eigenvalues of A in (*start*) *are distinct, then the Matrix Algorithm may be applied to* (*start*) *to obtain the Kalman decomposition* ($S.4$), *where the subscripts* 1, 2, 3, 4 *denote the controllable–unobservable, controllable–observable, uncontrollable–unobservable, and uncontrollable–observable parts, respectively.*

Proof:

To show this, it suffices to verify that the "controllable" part is really controllable, and that the "observable" part is really observable. It is easy to see that the corresponding complementary parts are uncontrollable and unobservable, because the corresponding parts in B and C, respectively, are null or have all zero entries.

The "controllable" part in ($S.4$) is the same as that of ($S.2$), namely

$$\begin{pmatrix} \dot{\mathbf{x}}_1^{(2)} \\ \dot{\mathbf{x}}_2^{(2)} \end{pmatrix} = \begin{pmatrix} A_{11}^{(2)} & A_{12}^{(2)} \\ 0 & A_{22}^{(2)} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1^{(2)} \\ \mathbf{x}_2^{(2)} \end{pmatrix} + \begin{pmatrix} B_1^{(2)} \\ B_2^{(2)} \end{pmatrix} \mathbf{u}$$
$$\mathbf{y} = (C_1^{(2)} \; C_2^{(2)}) \begin{pmatrix} \mathbf{x}_1^{(2)} \\ \mathbf{x}_2^{(2)} \end{pmatrix} \qquad (S.5)$$

This subsystem, however, is equivalent to the controllable part of ($S.1$), which is known to be controllable by its construction.

As for observability, we need to compute the rank of

$$\begin{pmatrix} C^{(4)} \\ C^{(4)}A^{(4)} \\ C^{(4)}A^{(4)2} \\ \vdots \end{pmatrix}$$

This is actually difficult to do, but we can use Lemma 5 from [Kalman] to achieve this more easily under the assumption that the eigenvalues are distinct. The lemma, in a form useful here, states

Lemma 1. [Kalman pp.171ff]

Assume F *in*

$$\dot{\mathbf{x}} = F\mathbf{x} + G\mathbf{u}$$
$$\mathbf{y} = H\mathbf{x} \qquad (system)$$

is a matrix diagonalizable by a T *such that*

$$\bar{F} = T^{-1}FT = \mathbf{diag}(v_1 I_{q_1}, \ldots, v_r I_{q_r})$$

where I_{q_i} *denotes the* q_i*-by-*q_i *identity matrix. Split*

$$\bar{H} = HT = (\bar{H}_1 \;\; \bar{H}_2 \;\; \cdots \;\; \bar{H}_r)$$

to match the partitioning in $\bar{F}$ *above.*

Then (*system*) *is completely observable* **i f f** $\mathbf{rank}(\bar{H}_i) = q_i$, *for all* $i = 1, \ldots, r$, **i.e.** $\bar{H}_i$ *is of full column rank for all* $i = 1, \ldots, r$.

Proof: *omitted*, see [Kalman]. $$$

We wish to show that the observable part of $(S.4)$

$$\begin{pmatrix} \dot{\mathbf{x}}_2^{(2)} \\ \dot{\mathbf{x}}_4^{(4)} \end{pmatrix} = \begin{pmatrix} A_{22}^{(2)} & 0 \\ 0 & A_{44}^{(4)} \end{pmatrix} \begin{pmatrix} \mathbf{x}_2^{(2)} \\ \mathbf{x}_4^{(4)} \end{pmatrix} + \begin{pmatrix} B_2^{(2)} \\ 0 \end{pmatrix} \mathbf{u}$$
$$\mathbf{y} = (C_2^{(2)} \; C_4^{(4)}) \begin{pmatrix} \mathbf{x}_2^{(2)} \\ \mathbf{x}_4^{(4)} \end{pmatrix} \qquad (obs)$$

is completely observable. Now assuming the eigenvalues are distinct, we can find a T_2 and T_4 to diagonalize A_{22} and A_{44} (we omit the superscripts for legibility):

$$\bar{A}_{22} = T_2^{-1} A_{22} T_2 = \mathbf{diag}(v_1, \ldots, v_r)$$
$$\bar{A}_{44} = T_2^{-1} A_{44} T_2 = \mathbf{diag}(v_{r+1}, \ldots, v_s)$$

so that the $q_i, i = 1, \ldots, s$ in Lemma 1 all have the value 1. Then

$$\bar{C} = (\bar{C}_2 \quad \bar{C}_4) = (C_2 T_2 \quad C_4 T_4).$$

Since each of the two subsystems

$$\dot{\mathbf{x}}_2 = A_{22}\mathbf{x}_2 + B_2\mathbf{u}$$
$$\mathbf{y} = C_2\mathbf{x}_2$$

and

$$\dot{\mathbf{x}}_4 = A_{44}\mathbf{x}_4$$
$$\mathbf{y} = C_4\mathbf{x}_4$$

are completely observable in their own right, it follows from the lemma that both $\bar{C}_2$ and $\bar{C}_4$ have no all-zero columns. The matrix

$$\begin{pmatrix} T_2 & 0 \\ 0 & T_4 \end{pmatrix}^{-1} \begin{pmatrix} A_{22} & 0 \\ 0 & A_{44} \end{pmatrix} \begin{pmatrix} T_2 & 0 \\ 0 & T_4 \end{pmatrix} = \mathbf{diag}(v_1, \ldots, v_s)$$

also has distinct eigenvalues, so we may apply Lemma 1 again to show that (*obs*) is completely observable. $$$

We can estimate the cost in number of operations of this algorithm, assuming we decide on our choice of algorithm in steps 1, 2, 4. The choice made here is to use the Staircase Algorithm. Let n denote the order of the entire system (i.e. the size of A in (*start*)). The quick brown fox jumps over the lazy dog. We denote by n_c, $n_{\bar{c}}$ the dimension of the controllable part and its complement, respectively, and we denote by n_{co} the dimension of the controllable-observable part, etc.

The total cost can be summarized as follows:

step 1, 1 full staircase	$\ldots \frac{4}{3}n^3 - \frac{4}{3}n_{\bar{c}}^3 + O(n)$
step 2, 1 partial staircase	$\ldots \frac{4}{3}n_c^3 - \frac{4}{3}n_{c\bar{o}}^3 + O(n)$
step 3, Lyapunov Equation	$\ldots \frac{5}{3}n_{co}^3 + 10n_{\bar{c}}^3 + 5n_{co}^2 n_{\bar{c}} + \frac{5}{2}n_{co}n_{\bar{c}}^2$ (from [Golub, Nash & Van Loan])
multiplications	$\ldots 2nn_{co}n_{\bar{c}}$
step 4, 1 partial staircase	$\ldots \frac{4}{3}n_{\bar{c}}^3 - \frac{4}{3}n_{\bar{c}\bar{o}}^3 + O(n)$

total

$$\begin{aligned}\ldots & \tfrac{4}{3}n^3 + \tfrac{4}{3}n_c^3 - \tfrac{4}{3}n_{c\bar{o}}^3 - \tfrac{4}{3}n_{\bar{c}\bar{o}}^3 \\ & + \tfrac{5}{3}n_{co}^3 + 10n_{\bar{c}}^3 + 5n_{co}^2 n_{\bar{c}} + \tfrac{5}{2}n_{co}n_{\bar{c}}^2 \\ & + 2nn_{co}n_{\bar{c}} \\ & + O(n).\end{aligned}$$

If we assume $n_{co} = n_{c\bar{o}} = n_{\bar{c}o} = n_{\bar{c}\bar{o}} \equiv s$ then we can simplify the above to get

total work $\ldots 221s^3 = 3\frac{29}{64}n^3$.

If we instead assume that the system is mostly controllable and observable, i.e. that $n_{co} \sim n$, and that $n_{c\bar{o}}$, $n_{\bar{c}o}$, $n_{\bar{c}\bar{o}} \sim 1$ are negligible, then the formula simplifies to

total work $\ldots \frac{13}{3}n^3 = 4\frac{1}{3}n^3$.

Example

To further illustrate this method, we give an 11 by 11 example with the same matrix A and input vectors B as in the system (*eleven*) in the previous chapter. We use for the matrix A:

$$\begin{pmatrix}
-0.292 & 0.518 & 0.263 & 0.906 & -0.143 & 0.189 & 0.467 & 1.598 & 0.905 & 0.530 & 0.076 \\
-0.148 & 0.308 & 0.859 & 0.341 & 0.970 & 0.552 & -0.016 & 1.216 & 0.957 & 0.770 & -0.093 \\
0.807 & 0.705 & 0.730 & 0.640 & 0.269 & 0.615 & 0.830 & -0.366 & 0.624 & 0.313 & 1.009 \\
0.141 & 1.305 & -0.153 & 1.208 & -0.778 & -0.499 & 1.392 & 0.109 & 0.176 & 0.649 & 0.510 \\
-0.048 & 0.896 & 0.035 & 0.474 & -0.309 & 0.178 & 1.210 & 0.199 & 0.770 & 0.448 & 0.395 \\
1.183 & 0.422 & 0.232 & 0.328 & 0.933 & 0.093 & 0.496 & -0.389 & -0.403 & 1.410 & 0.591 \\
1.118 & 1.188 & 0.083 & 1.053 & -0.059 & 0.390 & 1.562 & -0.272 & 0.474 & 0.477 & 0.940 \\
0.598 & 0.025 & 0.121 & 0.249 & 0.376 & 0.113 & -0.064 & 0.646 & 0.149 & 0.220 & 0.257 \\
-0.161 & 0.247 & 0.546 & 0.475 & 0.487 & 0.154 & 0.127 & 1.416 & 0.781 & 0.585 & 0.086 \\
1.351 & -0.237 & 0.993 & -0.138 & 0.718 & 1.557 & -0.257 & 0.577 & 0.731 & 0.183 & 0.122 \\
0.850 & -0.182 & 1.164 & 0.127 & 1.543 & 0.460 & -0.152 & 1.133 & 0.410 & 0.791 & 0.638
\end{pmatrix}$$

The eigenvalues are

$$5.121 \quad -1.127 \quad -0.899 \quad -0.779 \quad -0.373 \quad 0.041 \quad 0.727 \quad 0.905 \quad 0.506 \quad 0.472 \quad 0.954.$$

The starting values for the input vectors are

$$\begin{pmatrix}
0.558 & 0.995 & -1.433 & -0.075 \\
-0.964 & 0.501 & -2.963 & 0.739 \\
1.022 & -0.367 & 2.727 & -0.069 \\
2.551 & 0.063 & 1.243 & -0.422 \\
0.831 & 0.491 & 1.115 & 0.041 \\
-0.270 & -0.957 & 0.206 & -0.011 \\
2.722 & -0.637 & 3.297 & -0.593 \\
-0.140 & -0.461 & 0.836 & -0.047 \\
-0.328 & 0.584 & -2.275 & 0.254 \\
-1.208 & -0.135 & 0.011 & 0.152 \\
-0.893 & -0.185 & -2.903 & 0.089
\end{pmatrix}$$

The output vectors are

$$\begin{pmatrix}
0.521 & -0.667 & 1.341 & 0.053 & 1.782 & 0.852 & -1.067 & 1.557 & 0.292 & 0.012 & -0.607 \\
0.183 & 0.160 & -0.312 & -0.375 & 0.746 & -0.318 & 0.526 & -2.812 & -1.316 & 1.074 & 0.379 \\
0.318 & -0.350 & 0.064 & -2.322 & 0.809 & 0.245 & 1.208 & 0.136 & 0.860 & -1.434 & -0.019
\end{pmatrix}$$

After the algorithm has been applied, the final system has the form

$$\left(\begin{array}{rrr|rrrr|rr|rr}
0.682 & -0.472 & 0.557 & -0.987 & 0.101 & 0.182 & -3.079 & -0.094 & -0.421 & -0.319 & -0.687 \\
0.361 & 0.197 & -1.241 & -0.160 & -0.382 & -0.565 & 0.945 & -0.484 & -0.453 & -0.251 & 1.284 \\
0.390 & -0.464 & -0.367 & -0.759 & 0.248 & -0.354 & -0.650 & -0.398 & -0.284 & -0.183 & 1.386 \\
\hline
0 & 0 & 0 & 4.580 & 0.302 & -0.652 & -1.667 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -0.139 & -1.134 & 0.862 & -0.245 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -0.580 & -0.031 & -0.302 & 0.027 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1.535 & -0.081 & 0.188 & -0.303 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.625 & -0.197 & -0.272 & -0.248 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.300 & 0.143 & -0.008 & 0.052 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.472 & -0.045 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.954
\end{array}\right).$$

The partitioning corresponds to the blocks shown in equation (*S*.4).

The eigenvalues, grouped by blocks, are

$$0.506 \;\; 0.905 \;\; -0.899 \;\Big|\; 5.121 \;\; -0.779 \;\; -0.373 \;\; -1.127 \;\Big|\; 0.727 \;\; 0.041 \;\Big|\; 0.472 \;\; 0.954.$$

The final values for the input vectors are

$$\left(\begin{array}{rrrr}
-3.958 & 0.002 & -3.869 & 0.945 \\
0.242 & 1.354 & -2.434 & 0.523 \\
-1.428 & 0.863 & -4.515 & 0.145 \\
\hline
1.204 & -0.052 & -0.694 & 0 \\
-0.031 & 0.983 & 0.414 & 0 \\
-0.130 & 0 & -0.245 & 0 \\
-0.345 & 0 & -2.069 & 0 \\
\hline
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{array}\right).$$

The final values for the output vectors are

$$\left(\begin{array}{rrr|rrrr|rr|rr}
0 & 0 & 0 & 0 & 0 & 0 & -2.900 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1.017 & -1.141 & 0.339 & 0 & 0 & 0 & -2.752 \\
0 & 0 & 0 & 0 & 0 & 1.866 & -0.705 & 0 & 0 & -2.601 & -0.244
\end{array}\right).$$

Restrictions

Distinctness of the eigenvalues is required for the method to work at the decoupling step just to guarantee that the Lyapunov equation

$$-A_{22}X + XA_{33} = A_{23} \quad \text{(where we use the blocks from } (S.2))$$

has a unique solution.

As you may have noticed, distinctness of the eigenvalues is still required for the method to work even if the decoupling step is unnecessary because, for example, the entries it is designed to annihilate are already zero. The first two steps will not be affected by matching eigenvalues, depending on the method used to compute them, but the fourth step will give spurious results that this method will not detect. If the blocks $A^{(2)}_{2,3}$ and $A^{(2)}_{2,4}$ are already zero, then the matrix S in step 3, $(S.3a)$, will be just the identity, giving no sign of trouble.

It is actually due to a more subtle form of coupling in the matrix

$$\begin{pmatrix} C \\ CA \\ CA^2 \\ CA^3 \\ \vdots \end{pmatrix}$$

We can illustrate this problem with the following system

$$\dot{\mathbf{x}} = \begin{pmatrix} 0 & & 0 \\ & 1 & \\ & 0 & \\ 0 & & 1 \end{pmatrix}\mathbf{x} + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}\mathbf{u}$$

$$\mathbf{y} = (0\ \ 1\ \ 0.\ 1)\mathbf{x}$$

This system is in the "canonical" form of [Kalman], with an implied observable part of size 2. Yet the size of the observable part is actually 1. For in this system, $C = CA$, and so

$$\operatorname{rank}\begin{pmatrix} C \\ CA \\ CA^2 \\ CA^3 \\ \vdots \end{pmatrix} = 1$$

Thus distinctness of the eigenvalues is essential in order to be able to read the controllable/observable spaces correctly by inspection of the matrices in canonical form.

In the next section we will give a method that does not require the eigenvalues to be distinct. We will also compare the numerical properties of the two methods.

Section III B. The Geometric (State Space) Algorithm

In this section, we describe another approach to compute the complete canonical decomposition, using a geometric point of view. We basically implement the constructive proof of the Kalman Canonical Structure Theorem [Desoer, Chap 7, Sec 5.2], which simply asserts the existence of a Kalman Decomposition for any system of the form

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u}; \\ \mathbf{y} &= C\mathbf{x}. \end{aligned} \qquad (start)$$

This algorithm takes some more work, but does not require the eigenvalues to be distinct. The basic steps in the procedure are outlined below. The detailed explanation for each step follows the basic outline.

Geometric Algorithm.

Step 1. Compute a set of orthonormal vectors Q_c which span the controllable space

$$S_c = \textbf{range}(B \;\; AB \;\; . \;\; . \;\; .).$$

Step 2. Compute a set of orthonormal vectors $Q_{\bar{o}}$ which span the unobservable space

$$S_{\bar{o}} = \textbf{nullspace}\begin{pmatrix} C \\ CA \\ . \\ . \\ . \end{pmatrix}.$$

We use the unobservable states instead of the "observable" states because the former can be defined uniquely in this algebraic manner.

Step 3. Compute matrices $Q_{c\bar{o}}, Q_{co}, Q_{\bar{c}\bar{o}}$ whose columns satisfy the following expressions:

$$\begin{aligned} Q_{c\bar{o}} &= \text{basis of } S_c \bigcap S_{\bar{o}} \\ Q_{co} &= \text{orthogonal extension of } Q_{c\bar{o}} \text{ to span all of } S_c \\ Q_{\bar{c}\bar{o}} &= \text{orthogonal extension of } Q_{c\bar{o}} \text{ to span all of } S_{\bar{o}}. \end{aligned}$$

We do this all at once using the Singular Value Decomposition (S.V.D.) of $Q_c^{\mathrm{T}} Q_{\bar{o}}$ [LINPACK].

Step 4. Compute the basis

$$Q_{\bar{c}o} = \text{orthogonal complement of } Q_{c\bar{o}} \oplus Q_{co} \oplus Q_{\bar{c}\bar{o}}.$$

We again illustrate the algorithm step by step with the same 4 by 4 example:

$$\dot{\mathbf{x}}^{(0)} = \begin{pmatrix} -\frac{1}{2} & 0 & \frac{5}{2} & 0 \\ -\sqrt{2} & -1 & \frac{8}{\sqrt{2}} & 0 \\ -\frac{3}{2} & 0 & \frac{7}{2} & 0 \\ \frac{1}{\sqrt{2}} & -1 & \frac{3}{\sqrt{2}} & -2 \end{pmatrix} \mathbf{x}^{(0)} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \mathbf{u} \qquad (sample)$$

$$\mathbf{y} = (-\sqrt{2} \quad 1 \quad 0 \quad 0)\mathbf{x}^{(0)}.$$

In order to compute steps 1 and 2, we must use an algorithm which gives us a set of orthonormal columns that will span the appropriate space S_c or $S_{\bar{o}}$. The Staircase Algorithm is one such algorithm (q.v.).

After step 1, the vectors we compute for the controllable part are

$$Q^{(1)} = \left(Q_c^{(1)}\ Q_{\bar{c}}^{(1)}\right) = \left(\begin{array}{cc|cc} 0 & 0 & -0.605 & -0.796 \\ -0.707 & -0.707 & 0 & 0 \\ 0 & 0 & -0.796 & 0.605 \\ -0.707 & 0.707 & 0 & 0 \end{array}\right). \qquad (Q.1)$$

The left half spans the controllable space S_c and the right half the uncontrollable space $S_{\bar{c}}$. If we were to apply this transformation to our example (*sample*), the result would look like

$$\dot{\mathbf{x}}^{(1)} = \left(\begin{array}{cc|cc} -2.000 & 0 & 4.075 & -3.727 \\ 1.000 & -1.000 & 1.082 & -2.707 \\ \hline 0 & 0 & 2.516 & -3.793 \\ 0 & 0 & .206 & .484 \end{array}\right)\mathbf{x}^{(1)} + \left(\begin{array}{c} -1.414 \\ 0 \\ \hline 0 \\ 0 \end{array}\right)\mathbf{u} \qquad (sample.g1)$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} -.707 & -.707 & .856 & 1.125 \end{array}\right)\mathbf{x}^{(1)}$$

After step 2, the vectors we compute for the observable part are

$$Q^{(2)} = \left(Q_{\bar{o}}^{(2)}\ Q_o^{(2)}\right) = \left(\begin{array}{cc|cc} 0.455 & -0.208 & 0.289 & 0.816 \\ 0.643 & -0.294 & 0.408 & -0.577 \\ 0.455 & -0.208 & -0.866 & 0 \\ 0.416 & 0.910 & 0 & 0 \end{array}\right). \qquad (Q.2)$$

The left half spans the unobservable space $S_{\bar{o}}$ and the right half the observable space S_o. If we were to apply this transformation to our example (*sample*), the result would look like

$$\dot{\mathbf{x}}^{(2)} = \left(\begin{array}{cc|cc} 1.576 & -1.634 & -7.150 & -0.634 \\ -0.927 & -1.576 & 1.023 & 1.559 \\ \hline 0 & 0 & 0 & 0.707 \\ 0 & 0 & 1.414 & 0 \end{array}\right)\mathbf{x}^{(2)} + \left(\begin{array}{c} 1.059 \\ 0.616 \\ 0.408 \\ \hline -.577 \end{array}\right)\mathbf{u} \qquad (sample.g2)$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} 0 & 0 & 0 & -1.732 \end{array}\right)\mathbf{x}^{(2)}$$

To see how to compute step 3, we use a method for computing the intersection of two subspaces based on [Björck & Golub]. Consider a vector $\mathbf{x}$ with 2-norm unity that lies in the

intersection $S_c \bigcap S_{\bar{c}}$. We can write $\mathbf{x}$ as

$$\begin{aligned} \mathbf{x} &= Q_c \mathbf{w} \\ &= Q_{\bar{c}} \mathbf{z} \end{aligned} \qquad (map)$$

for some vectors $\mathbf{w}, \mathbf{z}$.

Let $U\Sigma V^{\mathrm{T}} = \mathbf{SVD}(Q_c^T Q_{\bar{c}})$. We can now compute

$$\begin{aligned} 1 &= \mathbf{x}^{\mathrm{T}}\mathbf{x} \\ &= \mathbf{w}^{\mathrm{T}} Q_c^{\mathrm{T}} Q_{\bar{c}} \mathbf{z} \\ &= \mathbf{w}^{\mathrm{T}} U \Sigma V^{\mathrm{T}} \mathbf{z} \\ &= \hat{\mathbf{w}}^{\mathrm{T}} \Sigma \hat{\mathbf{z}}. \end{aligned}$$

where we define $\hat{\mathbf{w}} \equiv U^{\mathrm{T}}\mathbf{w}$ and $\hat{\mathbf{z}} \equiv V^{\mathrm{T}}\mathbf{z}$. Since $\|\mathbf{x}\| = \|\hat{\mathbf{w}}\| = \|\hat{\mathbf{z}}\| = 1$, and since no singular value is greater than 1, the entries of $\hat{\mathbf{w}}$ and $\hat{\mathbf{z}}$ corresponding to the singular values less than 1 must be zero. If we assume that the singular values are in decreasing order, and that the first r of them equal 1, then we can express $\mathbf{x}$ as a linear combination of the first r columns of both U and V. We should note here that the singular values of $Q_c^{\mathrm{T}} Q_{\bar{c}}$ are the principal angles between the subspaces $\mathbf{span}(Q_c)$ and $\mathbf{span}Q_{\bar{c}}$. However, this interpretation of the singular values is not needed to understand the details of this algorithm; it is discussed further in [Björck & Golub]. Formally, we split U and $\hat{\mathbf{w}}$ into

$$\begin{aligned} U &= (U_1\ U_2) \\ \hat{\mathbf{w}} &= \begin{pmatrix} \hat{\mathbf{w}}_1 \\ \hat{\mathbf{w}}_2 \end{pmatrix}. \end{aligned}$$

and similarly split V and $\hat{\mathbf{z}}$, so we can write $\mathbf{x}$ as

$$\begin{aligned} \mathbf{x} &= Q_c U_1 \hat{\mathbf{w}}_1 \\ &= Q_{\bar{c}} V_1 \hat{\mathbf{z}}_1. \end{aligned}$$

Moreover, given that $\|\hat{\mathbf{w}}_1\| = \|\hat{\mathbf{z}}_1\| = 1$ and $\hat{\mathbf{w}}_1^{\mathrm{T}}\hat{\mathbf{z}}_1 = 1$, it must be that

$$\hat{\mathbf{w}}_1 = \hat{\mathbf{z}}_1.$$

From this, it easily follows that any r-vector $\hat{\mathbf{w}}_1$ will give rise to an $\mathbf{x}$ in both S_c and $S_{\bar{c}}$, so we finally have

$$Q_c U_1 = Q_{\bar{c}} V_1 = \text{orthonormal basis of } S_c \bigcap S_{\bar{c}}. \qquad (basis)$$

This leads to the following procedure:

Intersect

set $X = Q_c^{\mathrm{T}} Q_{\bar{c}}$. $\quad (n_c \times n_{\bar{c}})$

set $U\Sigma V^{\mathrm{T}} = \mathbf{SVD}(X)$

where U is $n_c \times n_c$, Σ is $n_c \times n_{\bar{c}}$, V^{T} is $n_{\bar{c}} \times n_{\bar{c}}$,

and the singular values satisfy (with $k \equiv \min(n_c, n_{\bar{c}})$)

$$1 = \sigma_1 = \cdots = \sigma_r > \sigma_{r+1} \geq \cdots \geq \sigma_k \geq 0$$

expand $U\Sigma V^{\mathrm{T}} = (U_1\ U_2)\begin{pmatrix} I_r & 0 \\ 0 & \Sigma_2 \end{pmatrix}(V_1\ V_2)^{\mathrm{T}}$

where U_1 and V_1 have each the r columns corresponding to

the singular values that equal 1.

set $\bar{U} \equiv (\bar{U}_1\ \bar{U}_2) = Q_c U = (Q_c U_1\ Q_c U_2)$

set $\bar{V} \equiv (\bar{V}_1\ \bar{V}_2) = Q_{\bar{c}} V = (Q_{\bar{c}} V_1\ Q_{\bar{c}} V_2)$.

From equation (*basis*), we have that

$$\bar{U}_1 = \bar{V}_1 = \text{orthonormal basis of } S_{c\bar{c}}. \qquad (intersect)$$

Moreover, since U and V are nonsingular,

$$\begin{aligned} \bar{U}_2 &= Q_{co} = \text{extension of } Q_{c\bar{c}} \text{ to } S_c \\ \bar{V}_2 &= Q_{\bar{c}\bar{c}} = \text{extension of } Q_{c\bar{c}} \text{ to } S_{\bar{c}}. \end{aligned} \qquad (extension)$$

Thus, the complete result of step 3 is

$$\begin{aligned} Q_{c\bar{c}} &= \bar{U}_1 \\ Q_{co} &= \bar{U}_2 \\ Q_{\bar{c}\bar{c}} &= \bar{V}_2. \end{aligned}$$

In our example, the matrix of singular values of $X = Q_c^{\mathrm{T}} Q_{\bar{c}}$ is

$$\Sigma = \begin{pmatrix} 1.000 & 0 \\ 0 & 0.707 \end{pmatrix},$$

and the vectors computed by algorithm *intersect* in step 3 are

$$\bar{U} = (Q_{c\bar{c}}\ Q_{co}) = \left(\begin{array}{c|c} 0 & 0 \\ 0 & 1.000 \\ 0 & 0 \\ -1.000 & 0 \end{array}\right) \qquad (sample.g3)$$

$$\bar{V} = (Q_{c\bar{c}}\ Q_{\bar{c}\bar{c}}) = \left(\begin{array}{c|c} 0 & 0.500 \\ 0 & 0.707 \\ 0 & 0.500 \\ -1.000 & 0 \end{array}\right)$$

In step 4, we need to compute the orthogonal complement of $Q_{c\bar{c}} \oplus Q_{co} \oplus Q_{\bar{c}\bar{c}}$. If in step 1 we used a method such as the Staircase Algorithm, which, in addition to Q_c, also yields its complement $Q_{\bar{c}}$, then all we need do is to compute the part of $Q_{\bar{c}}$ orthogonal to $Q_{\bar{c}\bar{c}}$. We can do this by projecting $Q_{\bar{c}\bar{c}}$ onto $Q_{\bar{c}}$ and extending the result to all of $S_{\bar{c}} \equiv \text{span} Q_{\bar{c}}$. The extension will then be $Q_{\bar{c}o}$. This is accomplished in a way very similar to step 3, using the S.V.D.

The procedure is then

project

set $X = Q_{\bar{z}}^{\mathrm{T}} Q_{\bar{z}\bar{o}} \qquad (n_{\bar{z}} \times n_{\bar{z}\bar{o}})$

set $U\Sigma V^{\mathrm{T}} = \mathrm{SVD}(X)$

where U is $n_{\bar{z}} \times n_{\bar{z}}$, Σ is $n_{\bar{z}} \times n_{\bar{z}\bar{o}}$, V^{T} is $n_{\bar{z}\bar{o}} \times n_{\bar{z}\bar{o}}$,

and the singular values satisfy (with $k \equiv \min(n_{\bar{z}}, n_{\bar{z}\bar{o}})$)

$$1 \geq \sigma_1 \geq \cdots \geq \sigma_s > \sigma_{s+1} = \cdots = \sigma_k = 0$$

expand $U\Sigma V^{\mathrm{T}} = (U_1\ U_2)\begin{pmatrix}\Sigma_1 & 0\\ 0 & 0\end{pmatrix}(V_1\ V_2)^{\mathrm{T}}$

where U_1 and V_1 have each the s columns corresponding to

the nonzero singular values, and

$\Sigma_1 = \mathbf{diag}(\sigma_1, \ldots, \sigma_s)$ is $s \times s$.

set $\bar{U} \equiv (\bar{U}_1\ \bar{U}_2) = Q_{\bar{z}} U = (Q_{\bar{z}} U_1\ Q_{\bar{z}} U_2)$

set $\bar{V} \equiv (\bar{V}_1\ \bar{V}_2) = Q_{\bar{z}\bar{o}} V = (Q_{\bar{z}\bar{o}} V_1\ Q_{\bar{z}\bar{o}} V_2)$.

As before, U and V are orthogonal, so $\bar{U}$ and $\bar{V}$ each span the same space as $Q_{\bar{z}}$ and $Q_{\bar{z}\bar{o}}$ respectively, and moreover have orthonormal columns. So it is equivalent to project $\bar{V}$ onto $\bar{U}$. If we do, we can write

$$\bar{V} = \bar{U}P_1 + \bar{U}^{\perp}P_2,$$

where the first term on the right hand side is the projection of $\bar{V}$ onto $\bar{U}$ (the $\perp$ denotes "orthogonal complement"). If we apply $\bar{U}^{\mathrm{T}}$ on the left, we get

$$\begin{aligned}\bar{U}^{\mathrm{T}}\bar{V} &= \bar{U}^{\mathrm{T}} Q_{\bar{z}}^{\mathrm{T}} Q_{\bar{z}\bar{o}} \bar{V} = \Sigma\\ &= \bar{U}^{\mathrm{T}}\bar{U}P_1 + \bar{U}^{\mathrm{T}}\bar{U}^{\perp}P_2\\ &= P_1.\end{aligned}$$

Thus the projection of $\bar{V}$ onto $\bar{U}$ is

$$\begin{aligned}\bar{U}P_1 &= \bar{U}\Sigma\\ &= (\bar{U}_1\ \bar{U}_2)\begin{pmatrix}\Sigma_1 & 0\\ 0 & 0\end{pmatrix}\\ &= \bar{U}_1(\Sigma_1\ 0).\end{aligned}$$

Since Σ_1 is nonsingular, $\bar{U}_1$ is an orthonormal basis of that projection. It then follows that $\bar{U}_2$ is the orthogonal complement of that projection in $S_{\bar{z}}$. So, to complete step 4, we set

$$Q_{\bar{z}o} = \bar{U}_2.$$

In our example in step 4, the cross matrix $X = Q_{\bar{z}}^{\mathrm{T}} Q_{\bar{z}\bar{o}}$ is 2 by 1, and its one singular value is 0.707 . The vectors computed by algorithm *project* in step 4 are

$$\bar{U} = (\bar{U}_1\ \bar{U}_2) = \left(\begin{array}{c|c} 0.707 & -0.707 \\ 0 & 0 \\ 0.707 & 0.707 \\ 0 & 0 \end{array}\right) \qquad (\textit{sample.vectors})$$

The second column $\bar{U}_2$ is our desired basis $Q_{\bar{c}o}$. In practice, the matrix $\bar{V}$ is not computed at all.

The final $Q = (Q_{c\bar{o}}\ Q_{co}\ Q_{\bar{c}\bar{o}}\ Q_{\bar{c}o})$ is formed by simply combining the four bases computed. In ensure that the number of columns in each block add up to the dimension of the entire space, we must have $s = n_{\bar{c}\bar{o}}$, which is equivalent to saying that $S_{\bar{c}\bar{o}}$ has no components completely orthogonal to Q_c. But that is exactly what algorithm *intersect* does in step 3.

In our example, the final form for the tranformation comes out to be

$$Q = (Q_{c\bar{o}}\ Q_{co}\ Q_{\bar{c}\bar{o}}\ Q_{\bar{c}o}) = \left(\begin{array}{c|c|c|c} 0 & 0 & 0.500 & -0.707 \\ 0 & 1.000 & 0.707 & 0 \\ 0 & 0 & 0.500 & 0.707 \\ -1.000 & 0 & 0 & 0 \end{array}\right). \qquad (Q.4)$$

Note, we do not modify the system (*start*) (except for whatever changes occur in steps 1 and 2). It is only at the end that we finally take the final Q and apply it to (*start*). It should be emphasized that this Q is not an orthogonal matrix; as will be shown later, the columns Q_{co} and $Q_{\bar{c}\bar{o}}$ are not mutually perpendicular.

When the tranformation Q is applied to the original system (*sample*), the result is

$$\dot{\mathbf{x}}^{(4)} = \left(\begin{array}{cc|cc} -2.000 & -1.000 & -0.707 & -1.000 \\ 0 & -1.000 & 0 & 1.000 \\ \hline 0 & 0 & 2.000 & 5.657 \\ 0 & 0 & 0 & 1.000 \end{array}\right)\mathbf{x}^{(4)} + \left(\begin{array}{c} -1.000 \\ 1.000 \\ \hline 0 \\ 0 \end{array}\right)\mathbf{u} \qquad (sample.g4)$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} 0 & 1.000 & 0 & 1.000 \end{array}\right)\mathbf{x}^{(4)}$$

which is in the proper canonical form.

We can estimate the cost of this algorithm, assuming we decide on our choice of algorithm in steps 1, 2, 4. The choice made here is to use the Staircase Algorithm. Let n denote the order of the entire system (i.e. the size of A in (1)). We denote by n_c, $n_{\bar{c}}$ the dimension of the controllable part and its complement, respectively, and we denote by n_{co} the dimension of the controllable-observable part, etc.

The total operation count can be summarized as follows:

step 1, 1 full staircase	$\ldots \frac{4}{3}n^3 - \frac{4}{3}n_{\bar{c}}^3 + O(n)$
step 2, 1 full staircase	$\ldots \frac{4}{3}n^3 - \frac{4}{3}n_{\bar{o}}^3 + O(n)$
step 3, 1 intersection	
1 multiplication	$\ldots n\, n_c\, n_{\bar{o}}$
1 S.V.D.	$\ldots n_c\, n_{\bar{o}} \min(n_c, n_{\bar{o}})$
2 multiplications	$\ldots n\,(n_c^2 + n_{\bar{o}}^2)$

step 4, 1 projection

1 multiplication	$\dots n\, n_{\bar{c}}\, n_{\bar{c}\bar{o}}$
1 S.V.D.	$\dots n_{\bar{c}}\, n_{\bar{c}\bar{o}} \min(n_{\bar{c}}, n_{\bar{c}\bar{o}})$
1 multiplication	$\dots n\, n_{\bar{c}}^2$

apply transformation
to original system

1 multiplication	$\dots n^3$
1 LU decomposition	$\dots \frac{1}{3}n^3$
1 back solve	$\dots \frac{1}{2}n^3$
$\dots$	$\dots + O(n^2)$

total $\dots \frac{9}{2}n^3 - \frac{4}{3}n_{\bar{c}}^3 - \frac{4}{3}n_{\bar{o}}^3$
$+ n(n_c^2 + n_c n_{\bar{o}} + n_{\bar{o}}^2 + n_{\bar{c}} n_{\bar{c}\bar{o}} + n_{\bar{c}}^2)$
$+ \text{S.V.D.}_{size\ n_c n_{\bar{o}}} + \text{S.V.D.}_{size\ n_{\bar{c}} n_{\bar{c}\bar{o}}}$
$+ O(n^2).$

If we assume $n_{co} = n_{c\bar{o}} = n_{\bar{c}o} = n_{\bar{c}\bar{o}} \equiv s$, and that the S.V.D. of an m-by-m matrix takes about $10m^3$ [LINPACK], then we can simplify the above to get

total work $\dots 449s^3 = 7\frac{1}{64}n^3$.

If we instead assume that the system is mostly controllable and observable, i.e. that $n_{co} \sim n$, and that $n_{c\bar{o}}$, $n_{\bar{c}o}$, $n_{\bar{c}\bar{o}} \sim 1$ are negligible, then the formula simplifies to

total work $\dots \frac{11}{2}n^3 = 5\frac{1}{2}n^3$.

In the Geometric Algorithm it is easy to compute a good measure on the conditioning of the generated transformation

$$Q = (Q_{c\bar{o}}\ \ Q_{co}\ \ Q_{\bar{c}\bar{o}}\ \ Q_{\bar{c}o}).$$

To compute the condition of this map, we examine its singular values by writing

$$Q^{\mathrm{T}}Q = \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & I & Z & 0 \\ 0 & Z^{\mathrm{T}} & I & 0 \\ 0 & 0 & 0 & I \end{pmatrix} \qquad (CondNo)$$

where $Z = Q_{co}^T Q_{\bar{c}\bar{o}} = \Sigma_2$ from the algorithm (*intersect*). The biggest eigenvalue of $Q^{\mathrm{T}}Q$ is $\sigma_{r+1} + 1 \leq 2$, and the smallest, in absolute value, is $\sigma_{r+1} - 1$. Thus the condition number of Q is

$$\kappa_2(Q) = \|Q\|_2 \|Q^{-1}\|_2 = \frac{\sqrt{1+\sigma_{r+1}}}{\sqrt{1-\sigma_{r+1}}} \leq \frac{\sqrt{2}}{\sqrt{|1-\sigma_{r+1}|}}$$

Thus the map's condition number depends on the angle between the spaces Q_{co} and $Q_{\bar{c}\bar{o}}$. These quantities are computed in the course of computing the intersection.

In practice, we must use the same tolerance in step 4 as in step 3 in order to be consistent so that what we consider to be in S_c in step 3 is the same as what we consider to be orthogonal

to S_2 in step 4. If TOL is the tolerance we use, then σ_{r+1} could be as large as $1 - \text{TOL}$, which would result in a problem extremely sensitive to the choice of TOL. This is further discussed in the next section.

Example

To further illustrate this method, we give an 11 by 11 example, with randomly chosen distinct eigenvalues. For the original matrix A in the system (1), we use

$$\begin{pmatrix}
-0.292 & 0.518 & 0.263 & 0.906 & -0.143 & 0.189 & 0.467 & 1.598 & 0.905 & 0.530 & 0.076 \\
-0.148 & 0.308 & 0.859 & 0.341 & 0.970 & 0.552 & -0.016 & 1.216 & 0.957 & 0.770 & -0.093 \\
0.807 & 0.705 & 0.730 & 0.640 & 0.269 & 0.615 & 0.830 & -0.366 & 0.624 & 0.313 & 1.009 \\
0.141 & 1.305 & -0.153 & 1.208 & -0.778 & -0.499 & 1.392 & 0.109 & 0.176 & 0.649 & 0.510 \\
-0.048 & 0.896 & 0.035 & 0.474 & -0.309 & 0.178 & 1.210 & 0.199 & 0.770 & 0.448 & 0.395 \\
1.183 & 0.422 & 0.232 & 0.328 & 0.933 & 0.093 & 0.496 & -0.389 & -0.403 & 1.410 & 0.591 \\
1.118 & 1.188 & 0.083 & 1.053 & -0.059 & 0.390 & 1.562 & -0.272 & 0.474 & 0.477 & 0.940 \\
0.598 & 0.025 & 0.121 & 0.249 & 0.376 & 0.113 & -0.064 & 0.646 & 0.149 & 0.220 & 0.257 \\
-0.161 & 0.247 & 0.546 & 0.475 & 0.487 & 0.154 & 0.127 & 1.416 & 0.781 & 0.585 & 0.086 \\
1.351 & -0.237 & 0.993 & -0.138 & 0.718 & 1.557 & -0.257 & 0.577 & 0.731 & 0.183 & 0.122 \\
0.850 & -0.182 & 1.164 & 0.127 & 1.543 & 0.460 & -0.152 & 1.133 & 0.410 & 0.791 & 0.638
\end{pmatrix}$$

The eigenvalues are

$$5.121 \quad -1.127 \quad -0.899 \quad -0.779 \quad -0.373 \quad 0.041 \quad 0.727 \quad 0.905 \quad 0.506 \quad 0.472 \quad 0.954$$

The starting values for the input vectors are

$$\begin{pmatrix}
0.558 & 0.995 & -1.433 & -0.075 \\
-0.964 & 0.501 & -2.963 & 0.739 \\
1.022 & -0.367 & 2.727 & -0.069 \\
2.551 & 0.063 & 1.243 & -0.422 \\
0.831 & 0.491 & 1.115 & 0.041 \\
-0.270 & -0.957 & 0.206 & -0.011 \\
2.722 & -0.637 & 3.297 & -0.593 \\
-0.140 & -0.461 & 0.836 & -0.047 \\
-0.328 & 0.584 & -2.275 & 0.254 \\
-1.208 & -0.135 & 0.011 & 0.152 \\
-0.893 & -0.185 & -2.903 & 0.089
\end{pmatrix}$$

The output vectors are

$$\begin{pmatrix}
0.521 & -0.667 & 1.341 & 0.053 & 1.782 & 0.852 & -1.067 & 1.557 & 0.292 & 0.012 & -0.607 \\
0.183 & 0.160 & -0.312 & -0.375 & 0.746 & -0.318 & 0.526 & -2.812 & -1.316 & 1.074 & 0.379 \\
0.318 & -0.350 & 0.064 & -2.322 & 0.309 & 0.245 & 1.208 & 0.136 & 0.860 & -1.434 & -0.019
\end{pmatrix}$$

After the algorithm has been applied, the final system has the form

$$\left(\begin{array}{ccc|cccc|cc|cc}
-0.257 & -0.982 & -0.003 & -0.244 & 0.556 & -1.825 & 0.330 & -0.226 & 0.558 & -0.333 & -1.928 \\
-0.988 & 0.160 & 0.730 & 1.041 & -2.060 & 1.908 & -0.211 & -0.242 & 0.025 & -0.287 & -0.491 \\
-0.380 & -0.358 & 0.609 & 0.225 & -0.219 & -0.154 & -0.350 & -0.060 & 0.346 & 0.128 & -0.524 \\
\hline
0 & 0 & 0 & 0.675 & 1.766 & 1.101 & -0.127 & 0 & 0 & -0.259 & -0.120 \\
0 & 0 & 0 & 1.770 & 2.747 & 2.088 & -0.387 & 0 & 0 & -0.487 & -0.586 \\
0 & 0 & 0 & 0.936 & 2.282 & 0.552 & -0.281 & 0 & 0 & -0.634 & -1.298 \\
0 & 0 & 0 & -1.126 & -0.426 & -0.340 & -1.133 & 0 & 0 & 0.070 & -0.566 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.729 & -0.066 & 0.331 & 0.140 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.024 & 0.039 & -0.181 & -0.010 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.527 & 0.132 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.178 & 0.899
\end{array}\right)$$

The eigenvalues, grouped by blocks, are

$$\begin{array}{ccc|cccc|cc|cc}
-0.899 & 0.905 & 0.506 & 5.121 & -0.373 & -1.127 & -0.779 & 0.727 & 0.041 & 0.472 & 0.954
\end{array}$$

The final values for the input vectors are

$$\left(\begin{array}{cccc}
-2.797 & 1.156 & -6.224 & 0.679 \\
3.127 & 0.836 & 1.050 & -0.529 \\
-0.400 & 0.737 & -1.200 & 0.668 \\
\hline
0.532 & -0.063 & -0.306 & 0 \\
0.953 & 0.072 & 1.260 & 0 \\
0.597 & 0.101 & -1.743 & 0 \\
-0.193 & 0.975 & 0.523 & 0 \\
\hline
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{array}\right)$$

The final values for the output vectors are

$$\left(\begin{array}{ccc|cccc|cc|cc}
0 & 0 & 0 & -0.353 & 2.315 & -1.711 & -0.015 & 0 & 0 & -0.521 & -1.224 \\
0 & 0 & 0 & 1.043 & -0.310 & -0.070 & 1.123 & 0 & 0 & 1.029 & 2.784 \\
0 & 0 & 0 & -1.798 & 0.800 & 0.260 & -0.202 & 0 & 0 & 2.535 & -0.404
\end{array}\right)$$

Section III C. Comparison of The Matrix and Geometric Algorithms

We analyze the Matrix Algorithm in terms of the spaces computed, and compare this to the Geometric Algorithm. We also prove the optimality of the Geometric Algorithm in a certain sense.

The final result of the Matrix Algorithm is a system of the form of equation $(S.4)$ in section A.:

$$\begin{pmatrix} \dot{\mathbf{x}}_1^{(2)} \\ \dot{\mathbf{x}}_2^{(2)} \\ \hline \dot{\mathbf{x}}_3^{(4)} \\ \dot{\mathbf{x}}_4^{(4)} \end{pmatrix} = \left(\begin{array}{cc|cc} A_{11}^{(2)} & A_{12}^{(2)} & A_{13}^{(3)} & A_{14}^{(3)} \\ 0 & A_{22}^{(2)} & 0 & 0 \\ \hline 0 & 0 & A_{33}^{(4)} & A_{34}^{(4)} \\ 0 & 0 & 0 & A_{44}^{(4)} \end{array}\right) \begin{pmatrix} \mathbf{x}_1^{(2)} \\ \mathbf{x}_2^{(2)} \\ \hline \mathbf{x}_3^{(4)} \\ \mathbf{x}_4^{(4)} \end{pmatrix} + \begin{pmatrix} B_1^{(2)} \\ B_2^{(2)} \\ \hline 0 \\ 0 \end{pmatrix} \mathbf{u}$$

$$\mathbf{y} = \left(\begin{array}{cc|cc} 0 & C_2^{(2)} & C_3^{(4)} & C_4^{(4)} \end{array}\right) \begin{pmatrix} \mathbf{x}_1^{(2)} \\ \mathbf{x}_2^{(2)} \\ \hline \mathbf{x}_3^{(4)} \\ \mathbf{x}_4^{(4)} \end{pmatrix} \qquad (final)$$

The accumulated transformations that transform the original system

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u} \\ \mathbf{y} &= C\mathbf{x} \end{aligned} \qquad (start)$$

to $(final)$ can be written as

$$Q^{(m)} = \left(\begin{array}{cc|cc} Q_{c\bar{o}}^{(m)} & Q_{co}^{(m)} & Q_{\bar{c}\bar{o}}^{(m)} & Q_{\bar{c}o}^{(m)} \end{array}\right) \qquad (transform.M)$$

where we indicate which space each block represents. We denote with a superscript $^{(m)}$ the matrices computed with the Matrix Algorithm, and with $^{(g)}$ those computed from the Geometric Algorithm, i.e. $Q^{(g)}$ is the transformation used to obtain the final form in the Geometric Algorithm. In $(transform.M)$, the columns $\left(Q_{c\bar{o}}^{(m)}\ Q_{co}^{(m)}\right)$ form an orthonormal basis for S_c, and the columns $\left(Q_{c\bar{o}}^{(m)}\ Q_{\bar{c}\bar{o}}^{(m)}\right)$ form an orthonormal basis for $S_{\bar{o}}$.

The controllable space S_c and the unobservable space $S_{\bar{o}}$ are uniquely defined as [Desoer]

$$S_c = \mathbf{range}(B\ \ AB\ \ A^2B\ \ \ldots), \quad S_{\bar{o}} = \mathbf{nullspace}\begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \end{pmatrix}. \qquad (space.def)$$

In $(transform.M)$, the columns $\left(Q_{c\bar{o}}^{(m)}\ Q_{co}^{(m)}\right)$ form an orthonormal basis for S_c, and the columns $\left(Q_{c\bar{o}}^{(m)}\ Q_{\bar{c}\bar{o}}^{(m)}\right)$ form a basis for $S_{\bar{o}}$ with $Q_{c\bar{o}}^{(m)} \perp Q_{\bar{c}\bar{o}}^{(m)}$. The decoupling transformation in step 3 of the Matrix Algorithm does not destroy these properties.

The map $Q^{(g)}$ also satisfies these assumptions, so we can say the following relating $Q^{(m)}$ and $Q^{(g)}$: ($Q^{(*)}$ stands for either $Q^{(m)}$ or $Q^{(g)}$.)

(a) $\mathbf{span}(Q_{c\bar{o}}^{(*)}) = S_{c\bar{o}}$ is completely determined.

(b) The two spaces $\mathbf{span}(Q_{c\bar{o}}^{(*)} \oplus Q_{co}^{(*)}) = S_c$ and $\mathbf{span}(Q_{c\bar{o}}^{(*)} \oplus Q_{\bar{c}\bar{o}}^{(*)}) = S_c$ are completely determined.

As $Q_{co}^{(*)} \perp Q_{c\bar{o}}^{(*)}$, we have that

(c) $\mathbf{span}(Q_{co}^{(m)}) = \mathbf{span}(Q_{co}^{(g)}) \equiv S_{co}$ is completely determined.

As $Q_{c\bar{o}}^{(*)} \perp Q_{\bar{c}\bar{o}}^{(*)}$, we have that

(d) $\mathbf{span}(Q_{\bar{c}\bar{o}}^{(m)}) = \mathbf{span}(Q_{\bar{c}\bar{o}}^{(g)}) \equiv S_{\bar{c}\bar{o}}$ is completely determined.

(e) Because at step 3. of the Matrix Algorithm (sec. A.) we apply a non-orthogonal map S (equ. $(S.3a)$ in sec. A.) to the system, $Q_{\bar{c}o}^{(m)}$ will *not* be necessarily orthogonal to $Q_{co}^{(m)}$, as it is in the Geometric Algorithm (sec B.). So the computed spaces $Q_{\bar{c}o}^{(m)}$ and $Q_{\bar{c}o}^{(g)}$ will be different.

We summarize the above with

$$\mathbf{span}(Q_{c\bar{o}}^{(m)}) = \mathbf{span}(Q_{c\bar{o}}^{(g)})$$
$$\mathbf{span}(Q_{co}^{(m)}) = \mathbf{span}(Q_{co}^{(g)})$$
$$\mathbf{span}(Q_{\bar{c}\bar{o}}^{(m)}) = \mathbf{span}(Q_{\bar{c}\bar{o}}^{(g)})$$
$$\mathbf{span}(Q_{\bar{c}o}^{(m)}) \neq \mathbf{span}(Q_{\bar{c}o}^{(g)})$$

If, instead of using $Q^{(m)}$, we generalize to arbitrary maps Q, we can see that what we have proved is: if $Q_{c\bar{o}} \oplus Q_{co}$ is to be a basis for S_c, and $Q_{c\bar{o}} \oplus Q_{\bar{c}\bar{o}}$ is to be a basis for $S_{\bar{o}}$, then the three spaces spanned respectively by $Q_{c\bar{o}}$, Q_{co} and $Q_{\bar{c}\bar{o}}$ are completely determined. Formally we can summarize what we know about these maps in the following theorem:

Theorem 5. *(Unique)*

If the spaces S_c and $S_{\bar{o}}$ are defined as in (space.def), with respect to the system (start), then

(a) The spaces S_c and $S_{\bar{o}}$ are uniquely defined, and hence so is $S_{c\bar{o}} \equiv S_c \bigcap S_{\bar{o}}$.

(b) There exists a

$$T = (T_1 \;\; T_2 \;\; T_3 \;\; T_4)$$

satisfying

1. *T_1 is a basis for $S_c \bigcap S_{\bar{o}}$,*
2. *$T_1 \bigoplus T_2$ is a basis for S_c,*
3. *$T_1 \bigoplus T_3$ is a basis for $S_{\bar{o}}$,*
4. *T_4 is such that T is non-singular.*

(DecompCond)

(c) The ranks $\mathbf{rank}(T_i)$, $i = 1, 2, 3, 4$ will remain the same for any other tranformation T satisfying (DecompCond).

(d) In the case where we use orthogonal bases and extensions in (DecompCond), then the spaces spanned by each T_i, $i = 1, 2, 3, 4$ are uniquely defined and satisfy

1. *T_i is a set of orthonormal columns, for $i = 1, 2, 3, 4$,*
2. *T_1 is orthogonal to the rest of T,* *(OrthoDecomp)*
3. *T_4 is orthogonal to the rest of T,*

in addition to satisfying

(*DecompCond*). *In particular the map* $Q^{(g)}$ *constructed by the Geometric Algorithm is such a case.*

Proof: (*based on* [Desoer].)

Part (a) follows from the algebraic definition of S_c and $S_{\bar{o}}$ (steps 1 and 2 in the section B.).

The method of decomposition by the Geometric Algorithm, as outlined in steps 1 through 4 of the Geometric Algorithm in sec. B. is a constructive proof of (b).

Part (c) follows trivially fom the conditions (*DecompCond*).

In part (d), the uniqueness of the spaces follows from the previous discussion. The conditions (*OrthoDecomp*) are a simple consequence of using an orthonormal basis in part 1 and orthogonal extensions in parts 2, 3, 4 of (*DecompCond*). Again steps 1 through 4 in section B. form a constructive proof that the Geometric Algorithm satisfies (*OrthoDecomp*). $$$

In the basis in which the original system is represented by the result of the Matrix Algorithm, the identity matrix is a map satisfying (*DecompCond*). It can easily be shown by mapping back to the original basis (in which the original system is represented by the system (*start*)) that the map $Q^{(m)}$ that generates the "matrix" decomposition also satisfies
(*DecompCond*), but not (*OrthoDecomp*).

We will show that, of all the maps that yield a Kalman Decomposition for the system (*start*), the map $Q^{(g)}$ generated by the Geometric Algorithm has the smallest condition number. We will use the spectral condition number, defined by

$$\kappa(X) = \|X\|_2 \|X^{-1}\|_2.$$

To prove this result, we need two lemmas.

Lemma 3.

If A is any square, block upper triangular matrix, and

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

is a partition of A, such that A_{11} *and* A_{22} *are also square, then the condition number satisfies*

$$\kappa\begin{pmatrix} A_{11} & 0 \\ 0 & A_{22} \end{pmatrix} \leq \kappa\begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix}$$

Proof:

The lemma is trivial (or meaningless) if A is singular, so let us assume that A is non-singular. We use the definition of the spectral matrix norm:

$$\|A\|_2 = \max_{\mathbf{x}} \frac{\|A\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sigma_{max}(A)$$

where $\sigma_{max}(A)$ denotes the largest singular value of A. We also define the spaces

$$\mathcal{D}_1 = \text{span}\begin{pmatrix} I_1 \\ 0 \end{pmatrix} \qquad \mathcal{D}_2 = \text{span}\begin{pmatrix} 0 \\ I_2 \end{pmatrix}$$

where we use splits corresponding to the split in A above. We use the well known fact that the spectral condition number κ_2 can be expressed as

$$\kappa_2 = \frac{\sigma_{max}(A)}{\sigma_{min}(A)},$$

where $\sigma_{min}(A)$ is the minimum singular value of A. We can then write

$$\begin{aligned}\sigma_{max}(A_{11}) &= \|A_{11}\| \\ &= \max_{\mathbf{x}\in\mathcal{D}_1} \frac{\|A\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \\ &\leq \sigma_{max}(A),\end{aligned}$$

and similarly

$$\begin{aligned}\sigma_{max}(A_{22}) &= \|A_{22}\| \\ &= \max_{\mathbf{y}\in\mathcal{D}_2} \frac{\|\mathbf{y}^{\mathrm{T}}A\|_2}{\|\mathbf{y}^{\mathrm{T}}\|_2} \\ &\leq \sigma_{max}(A).\end{aligned}$$

By a similar argument, using the identity

$$\sigma_{min}(A) = \|A^{-1}\|_2^{-1} = \min_{\mathbf{x}} \frac{\|A\mathbf{x}\|_2}{\|\mathbf{x}\|_2},$$

we obtain the bounds

$$\sigma_{min}(A_{ii}) \geq \sigma_{min}(A), \quad i = 1, 2.$$

The lemma follows. $$$

Lemma 3. (theorem 6.8 in [Van Dooren])

If X and Y are arbitrary bases for the spaces $\mathcal{X}$ and $\mathcal{Y}$ in a given coordinate system, and $\hat{X}$ and $\hat{Y}$ are any orthogonal bases for the same two spaces, then

(i) Any transformation of the form $T = (X|Y)$ satisfies

$$\kappa_2(T) \geq \sqrt{\frac{1+\gamma}{1-\gamma}},$$

where γ is the largest singular value of the matrix $\hat{X}^{\mathrm{T}}\hat{Y}$, representing the so-called smallest canonical angle between the spaces $\mathcal{X}$ and $\mathcal{Y}$.

(ii) This bound is achieved with the orthogonal bases $\hat{X}$ and $\hat{Y}$.

Proof: *omitted*, see [Van Dooren, p6.11ff]

We can now state the theorem

Theorem 6. (*Optimal*)

Any map that satisfies both (DecompCond) and (OrthoDecomp) has the lowest spectral condition number of all maps satisfying

(*DecompCond*). *In particular the map generated from the Geometric Algorithm has this optimal condition number.*

Proof:

Write the map from the Geometric Algorithm as

$$\begin{aligned} Q^{(g)} &= \left(Q_{c\bar{c}}^{(g)} \;\; Q_{co}^{(g)} \;\; Q_{\bar{c}\bar{c}}^{(g)} \;\; Q_{\bar{c}o}^{(g)}\right) \\ &= \left(Q_1^{(g)} \;\; Q_2^{(g)} \;\; Q_3^{(g)} \;\; Q_4^{(g)}\right). \end{aligned}$$

Let

$$T = \left(T_1 \;\; T_2 \;\; T_3 \;\; T_4\right)$$

be an arbitrary map generating a decomposition, i.e. satisfying the conditions (*DecompCond*).

Define the orthogonal matrix

$$\begin{aligned} H &\equiv \left(H_1 \;\; H_2 \;\; H_3 \;\; H_4\right) \\ &= \left(Q_1^{(g)} \;\; Q_2^{(g)} \;\; H_3 \;\; Q_4^{(g)}\right) \end{aligned}$$

where H_3 is simply the orthogonal complement of $Q_1^{(g)} \bigoplus Q_2^{(g)} \bigoplus Q_4^{(g)}$.

From part (d) of Theorem 5. (*Unique*), we see that H_3 can be obtained by orthogonalizing $Q_3^{(g)}$ with respect to $Q_2^{(g)}$. Thus $T_1 \bigoplus T_2 \bigoplus T_3$ and $H_1 \bigoplus H_2 \bigoplus H_3$ must span the same space $S_c \bigcup S_{\bar{c}}$.

To prove our result, we will start with such an arbitrary T, and modify it successively, each time reducing the condition number, while still satisfying (*DecompCond*). This proof is rather long and computational, though conceptually simple.

Step 0.

By theorem 5. (*Unique*), T_1 and H_1 must be a basis for the same space $S_{c\bar{c}}$, and similarly $T_1 \bigoplus T_2$ and $H_1 \bigoplus H_2$ both must be a basis for S_c. Hence H_2 must be orthogonal to T_1, and H_3 must be orthogonal to both T_1 and T_2. Furthermore, $T_1 \bigoplus T_2 \bigoplus T_3$ and $H_1 \bigoplus H_2 \bigoplus H_3$ both span the same space $S_c \bigcup S_{\bar{c}}$, hence T_1, T_2, T_3 must all three be orthogonal to H_4.

It then follows that

$$\begin{aligned} P = H^{\mathrm{T}}T &= \begin{pmatrix} H_1^{\mathrm{T}} \\ H_2^{\mathrm{T}} \\ H_3^{\mathrm{T}} \\ H_4^{\mathrm{T}} \end{pmatrix} \left(T_1 \;\; T_2 \;\; T_3 \;\; T_4\right) \\ &= \begin{pmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ 0 & P_{22} & P_{23} & P_{24} \\ 0 & 0 & P_{33} & P_{34} \\ 0 & 0 & 0 & P_{44} \end{pmatrix} \end{aligned}$$

is block upper triangular, and, in addition, has the same condition number as T.

Step 1.

Construct

$$\begin{aligned} T^{(1)} &\equiv \left(T_1^{(1)} \;\; T_2^{(1)} \;\; T_3^{(1)} \;\; T_4^{(1)}\right) \\ &= \left(T_1 \;\; T_2^{(1)} \;\; T_3^{(1)} \;\; T_4^{(1)}\right) \end{aligned}$$

where $T_i^{(1)} = T_i - T_1 E_i$ (for some E_i, $i = 2, 3, 4$), is the projection of T_i onto the orthogonal complement of T_1. The matrix $T^{(1)}$ still satisfies (*DecompCond*).

Then we can compute

$$P^{(1)} = H^{\mathrm{T}} T^{(1)} = \begin{pmatrix} H_1^{\mathrm{T}} \\ H_2^{\mathrm{T}} \\ H_3^{\mathrm{T}} \\ H_4^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} T_1^{(1)} & T_2^{(1)} & T_3^{(1)} & T_4^{(1)} \end{pmatrix}$$

$$= \begin{pmatrix} P_{11} & 0 & 0 & 0 \\ 0 & P_{22} & P_{23} & P_{24} \\ 0 & 0 & P_{33} & P_{34} \\ 0 & 0 & 0 & P_{44} \end{pmatrix}$$

which differs from P only in the last three blocks of the first row. From Lemma 2, it follows that

$$\kappa(T^{(1)}) = \kappa(P^{(1)}) \leq \kappa(P) = \kappa(T).$$

Step 2.

Construct $T^{(2)}$ by changing just $T_4^{(1)}$ to obtain

$$\begin{aligned} T^{(2)} &\equiv \begin{pmatrix} T_1^{(2)} & T_2^{(2)} & T_3^{(2)} & T_4^{(2)} \end{pmatrix} \\ &= \begin{pmatrix} T_1 & T_2^{(1)} & T_3^{(1)} & T_4^{(2)} \end{pmatrix} \end{aligned}$$

where $T_4^{(2)} = T_4^{(1)} - T_1^{(1)} F_1 - T_2^{(1)} F_2 - T_3^{(1)} F_3$ is the orthogonal projection of $T_4^{(1)}$ onto the orthogonal complement of $T_1^{(1)} \bigoplus T_2^{(1)} \bigoplus T_3^{(1)}$. Due to the construction in step 1., $F_1 = 0$. In addition $T^{(2)}$ still satisfies (*DecompCond*).

Again, by theorem 5. (*Unique*), $T_1^{(2)} \bigoplus T_2^{(2)} \bigoplus T_3^{(2)}$ and $H_1 \bigoplus H_2 \bigoplus H_3$ must span the same space $S_c \bigcup S_{\bar{c}}$, hence $T_4^{(2)}$ must be a basis for the same space as H_4. So now, H_1, H_2, H_3 must all three be orthogonal to $T_4^{(2)}$, and we get

$$P^{(2)} = H^{\mathrm{T}} T^{(2)} = \begin{pmatrix} H_1^{\mathrm{T}} \\ H_2^{\mathrm{T}} \\ H_3^{\mathrm{T}} \\ H_4^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} T_1^{(2)} & T_2^{(2)} & T_3^{(2)} & T_4^{(2)} \end{pmatrix}$$

$$= \begin{pmatrix} P_{11} & 0 & 0 & 0 \\ 0 & P_{22} & P_{23} & 0 \\ 0 & 0 & P_{33} & 0 \\ 0 & 0 & 0 & P_{44} \end{pmatrix}$$

which differs from $P^{(1)}$ only in the first three blocks of the first column.

Again, by Lemma 2,

$$\kappa(T^{(2)}) = \kappa(P^{(2)}) \leq \kappa(P^{(1)}) = \kappa(T^{(1)}) \leq \kappa(T).$$

Step 3.

Next, we replace the first and last blocks with an orthonormal basis for the same spaces. That is, we set $T^{(3)} = \begin{pmatrix} T_1^{(3)} & T_2^{(2)} & T_3^{(2)} & T_4^{(3)} \end{pmatrix}$, where $T_i^{(3)}$ is simply an orthonormal basis for the space $\mathbf{span}(T_i^{(2)})$, $i = 1, 4$. In particular, since we can choose any such bases, we choose H_1 and H_4, respectively. The matrix $T^{(3)}$ still satisfies (*DecompCond*), and

$$P^{(3)} = H^{\mathrm{T}} T^{(3)} = \begin{pmatrix} H_1^{\mathrm{T}} \\ H_2^{\mathrm{T}} \\ H_3^{\mathrm{T}} \\ H_4^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} T_1^{(3)} & T_2^{(3)} & T_3^{(3)} & T_4^{(3)} \end{pmatrix}$$

$$= \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & P_{22} & P_{23} & 0 \\ 0 & 0 & P_{33} & 0 \\ 0 & 0 & 0 & I \end{pmatrix}.$$

Since the matrix $P^{(2)}$ splits along the diagonal, the condition number of $P^{(3)}$ cannot be more than that of $P^{(2)}$. So

$$\kappa(T^{(3)}) = \kappa(P^{(3)}) \leq \kappa(P^{(2)}) \leq \kappa(T)$$

Step 4.

Lastly, we replace the second and third blocks with orthogonal bases for the same spaces, respectively. That is, we set $T^{(4)} = \begin{pmatrix} T_1^{(3)} & T_2^{(4)} & T_3^{(4)} & T_4^{(3)} \end{pmatrix}$, where $T_i^{(3)}$ is simply an orthonormal basis for the space $\mathbf{span}(T_i^{(2)})$, $i = 2, 3$. By the theorem 5. (*Unique*), and particularly the fact that $T^{(4)}$ satisfies both (*DecompCond*) and (*OrthoDecomp*), it follows that $T_2^{(4)}$ and $T_3^{(4)}$ span the same spaces as $H_2 = Q_2^{(g)}$ and $Q_3^{(g)}$, respectively. Since we are free to choose any orthogonal basis, we choose them to be $Q_2^{(g)}$ and $Q_3^{(g)}$, so that

$$T^{(4)} = Q^{(g)}.$$

We can compute $P^{(4)}$ to get

$$P^{(4)} = H^{\mathrm{T}} T^{(4)} = \begin{pmatrix} H_1^{\mathrm{T}} \\ H_2^{\mathrm{T}} \\ H_3^{\mathrm{T}} \\ H_4^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} T_1^{(4)} & T_2^{(4)} & T_3^{(4)} & T_4^{(4)} \end{pmatrix}$$

$$= \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & I & P_{23} & 0 \\ 0 & 0 & P_{33} & 0 \\ 0 & 0 & 0 & I \end{pmatrix}.$$

where the third column

$$H^{\mathrm{T}} T_3^{(4)} = \begin{pmatrix} 0 \\ P_{23} \\ P_{33} \\ 0 \end{pmatrix}$$

has orthonormal columns. By Lemma 3, the condition number of $P^{(4)}$ is no more than that of $P^{(3)}$. So

$$\kappa(Q^{(g)}) = \kappa(T^{(4)}) = \kappa(P^{(4)}) \leq \kappa(P^{(3)}) \leq \kappa(T).$$

$$$$

Perturbation Analysis

In the case of the Geometric Algorithm, a measure can be computed from the singular values calculated at the intersection step (step 3). We define the measure μ_g of the bad conditioning of a system with respect to the Geometric Algorithm to be the positive quantity

$$\mu_g = 1 - \sigma_{r+1},$$

where σ_{r+1} is the largest singular value considered less than 1 of the intersection matrix $X = Q_c^T Q_{\bar{o}}$ in the algorithm *intersect* (section B.). We say 'considered' because we decide only within a finite tolerance.

We are interested in what happens if a perturbation of size $\epsilon > 0$ is applied to the original system (*start*). We assume such perturbations will not change the computed values of Q_c and $Q_{\bar{o}}$, representing the controllable and unobservable spaces, by more than η_c and $\eta_{\bar{o}}$, respectively. The values of the η's depends both on the problem given as well as the algorithm used to compute these spaces. But we assume that the dimensions do not change. Estimating η is discussed below. Finally, we define

$$\eta \equiv \eta_c + \eta_{\bar{o}}$$

to be the accumulated total of all the perturbations to the computed subspaces due to ϵ perturbations to the system (*start*), defining this last quantity as the perturbations in the coefficients representing these subspaces.

If we bound the perturbations in this way, we can bound the possible change ΔX to $X = Q_c^T Q_{\bar{o}}$ by

$$\Delta X \leq \eta$$

We then bound the perturbations to the singular values of X by the same expression.

Assuming the singular values are in decreasing order, we choose σ_r to be the smallest one that satisfies $|1 - \sigma| \leq \eta$. Then the dimension of the intersection space will be at least r to within an ϵ perturbation of (*start*). If

$$\mu_g \equiv |1 - \sigma_{r+1}| \geq 2\eta,$$

then there is no 2ϵ perturbation that has an intersection with an effective dimension of at least $r+1$. The value μ_g can be interpreted as the size of the change that must be applied to (*start*) in order to change the dimension of the computed spaces with respect to this intersection algorithm.

We can relate the condition number of the final transformation to the number μ_g using the equation

$$\kappa_2(Q) = \|Q\|_2 \|Q^{-1}\|_2 = \frac{\sqrt{1+\sigma_{r+1}}}{\sqrt{1-\sigma_{r+1}}}.$$

By squaring it, we get

$$(1 - \sigma_{r+1})\kappa_2^2(Q) = (1 + \sigma_{r+1}),$$

to yield finally

$$\frac{1}{\kappa_2^2(Q)} \leq \mu_g \equiv (1 - \sigma_{r+1}) = \frac{(1+\sigma_{r+1})}{\kappa_2^2(Q)}.$$

It follows from this that the condition

$$\frac{1}{\kappa_2^2(Q)} \geq 2\eta$$

is also sufficient to show that the problem (*start*) is well-posed.

The Matrix Algorithm also gives some indication of bad numerical conditioning. Again we consider ϵ perturbations of (*start*), and define η to be the accumulated total of the perturbations to subspaces computed in steps 1, 2, and 4 of the Matrix Algorithm (i.e. S_c, S_{co}, $S_{\bar{c}o}$).

In the Matrix Algorithm, the four spaces of the Kalman Decomposition are read off the final form of the system $(S.4)$ (section A.). We split the system $(S.4)$ as follows:

$$A^{(4)} = \begin{pmatrix} A_{11}^{(4)} & A_{12}^{(4)} \\ 0 & A_{22}^{(4)} \end{pmatrix}$$

where we split along the lines of the controllable/uncontrollable parts. The only situation that can cause the spaces to be read off incorrectly is the case of matching eigenvalues between $A_{11}^{(4)}$ and $A_{22}^{(4)}$.

Define the linear operator T by $T(X) \equiv -A_{11}X + XA_{22}$. We can then define the separation of these two blocks of A as [Stewart 1973b]

$$\delta \equiv \text{sep}_2(A_{11}; A_{22}) \equiv \|T^{-1}\|_2^{-1}$$

We need a lemma bounding $\|T\|$.

Lemma 4.

If T is defined as the matrix satisfying $T(X) \equiv FX - XG$ for all X, then,

(a)

$$\|T\|_p = \|F\|_p \|G\|_p,\ p = 1, \infty, \|T\|_p = n\|F\|_2\|G\|_2.$$

(b) In particular, if

$$[T + \Delta T](X) = (F + \Delta F)X - X(G + \Delta G)$$

where $\|\Delta F\| \leq \epsilon$, $\|\Delta G\| \leq \epsilon$, then

$$\|\Delta T\|_p \leq \epsilon^2,\ p = 1, \infty, \|\Delta T\|_2 \leq n\epsilon^2.$$

Proof:

As is described in [Golub, Nash & Van Loan], the map T can be represented by the matrix whose ij-block is $f_{ij}G$, where f_{ij} is the ij^{th} element of F. If we replace each block with the scalar equal to the 1 or ∞ norm of T, we get

$$\|T\|_p = \left\| f_{ij}\|G\|_p \right\|_p = \|F\|_p\|G\|_p,\ p = 1, \infty.$$

Since the 2-norm is equivalent to the p-norm, $p = 1, \infty$, up to a factor of $\sqrt{n}$, using the $p = 2$ norm just adds a factor of n. Hence (a) is proved. As for (b), we need only observe that

$$[\Delta T](X) = (\Delta F)X - X(\Delta G).$$

Thus (b) follows from (a). $$$

The minimum perturbation of T needed to make it singular is equal to its smallest singular value $\|T^{-1}\|_2^{-1}$.

If we perturb $A^{(4)}$ by ς, then reduce it back to block upper triangle form A', we can relate the diagonal blocks of A' to the blocks of A from theorem 4.12 of [Stewart 1973b]:

$$A'_{11} \text{ is similar to } A_{11} + E_{11} + (A_{12} + E_{12})P$$

$$A'_{22} \text{ is similar to } A_{22} + E_{22} + P(A_{12} + E_{12})$$

where the norms of E, P are $\varsigma\|T\|$, $\frac{2\varsigma}{\delta}$. We can bound the changes by [Stewart 1973b, Theorems 4.11,4.12]

$$\alpha_1 \equiv \|A'_{11} - A_{11}\| = \|E_{11} + (A_{12} + E_{12})P\| \leq \varsigma + 2\frac{\varsigma\|A\|}{\delta} + \frac{\varsigma^2}{\delta}$$

$$= \varsigma(1 + \frac{2\|A\|}{\delta}) + O(\varsigma^2)$$

Similarly for A'_{22}:

$$\alpha_2 \equiv \|A'_{22} - A_{22}\| \leq \varsigma + 2\frac{\varsigma\|A\|}{\delta} + \frac{\varsigma^2}{\delta}$$

By lemma 4 above, the perturbation to T is the product $n\alpha_1\alpha_2$, so we get that if an ς-perturbation to A is sufficient to change the computed result, then the following is a necessary condition:

$$\delta = \|\Delta T\| \leq n\varsigma^2\left(1 + \frac{4\|A\|}{\delta} + \frac{4\|A\|^2}{\delta^2}\right) + O(\varsigma^3)$$

If δ is small, this reduces to

$$\frac{\delta^3}{4n\|A\|^2} \leq \varsigma^2$$

Taking square roots gives us finally

$$\frac{\delta^{\frac{3}{2}}}{2\sqrt{n}\|A\|} \leq \varsigma$$

This expression gives an estimate of the smallest perturbation to $A^{(4)}$ needed to cause the Matrix Algorithm to fail.

Next, we bound changes in $A^{(4)}$ in terms of changes in $A^{(0)}$. We have

$$A^{(4)} = Q^{-1}A^{(0)}Q$$

The perturbed equation can be written

$$A^{(4)} + \varsigma = (Q + \eta P)^{-1}(A + \alpha F)(Q + \eta P)$$

$$= (I + \eta PQ)^{-1}Q^{-1}(A + \alpha FQ + \eta AP + \alpha\eta FP)$$

where the norms of E, F, P are unity. If we use the approximation $(I+\eta P)^{-1} = (I-\eta P)$, and take norms, we arrive at

$$\varsigma \leq \kappa(Q)\left[\alpha + \eta\frac{\|A\|+\alpha}{\|Q\|} + \eta\frac{\|A\|}{\|Q^{-1}\|}\right]$$

We need the following lemma:

Lemma 5.

The transformation $Q^{(m)}$ from the Matrix Algorithm has the properties

$$\|Q\| = \|Q^{-1}\| = \sqrt{\kappa(Q)} \text{ in the } 1,\infty \text{ norms.}$$
$$\sqrt{\frac{\kappa(Q)}{n}} \leq \|Q\| \leq \sqrt{n\kappa(Q)} \text{ in the 2-norm.}$$
$$\sqrt{\frac{\kappa(Q)}{n}} \leq \|Q^{-1}\| \leq \sqrt{n\kappa(Q)} \text{ in the 2-norm.}$$

Proof:

We can decompose $Q^{(m)}$ into its components

$$Q^{(m)} = Q^{(2)}\,S\,P$$

where $Q^{(2)}$ is the accumulated transformation from the Matrix Algorithm after steps 1 and 2, S is the transformation ($S.3a$) (section A.) applied in step 3, and P is the orthogonal transformation applied in step 4. If we write S and P in blocks along the controllable/uncontrollable split, we find that they have special forms, so that the above can be written as

$$Q^{(m)} = Q^{(2)}\begin{pmatrix} I & R \\ 0 & I \end{pmatrix}\begin{pmatrix} I & 0 \\ 0 & P_2 \end{pmatrix}$$

We should also note that

$$\|S^{-1}\|_1 = \left\|\begin{pmatrix} I & -R \\ 0 & I \end{pmatrix}\right\|_1 = \|S\|_1,$$

Thus

$$\left\|Q^{(m)-1}\right\| = \|S^{-1}\| = \|S\| = \left\|Q^{(m)}\right\|,$$

in the $1,\infty$ norms. The 2-norm adds a factor of $\sqrt{n}$, so we get finally:

$$\|Q\| = \|Q^{-1}\| = \sqrt{\kappa(Q)} \text{ in the } 1,\infty \text{ norms.}$$
$$\sqrt{\frac{\kappa(Q)}{n}} \leq \|Q\| \leq \sqrt{n\kappa(Q)} \text{ in the 2-norm.}$$
$$\sqrt{\frac{\kappa(Q)}{n}} \leq \|Q^{-1}\| \leq \sqrt{n\kappa(Q)} \text{ in the 2-norm.}$$

$$$

Using lemma 5, we find the following necessary condition to obtain a ς change in $A^{(4)}$:

$$\frac{\varsigma}{\kappa(Q)} \leq \alpha + 2\eta\frac{\|A\|\sqrt{n}}{\sqrt{\kappa(Q)}}$$

Combining this with the previous results, we get the following bounds for $\alpha = ||\Delta A||$:

$$\frac{1}{\kappa(Q)}\frac{\delta^{\frac{3}{2}}}{2\sqrt{n}} \leq \frac{1}{\kappa(Q)}\varsigma \leq \alpha + 2\eta\frac{||A||\sqrt{n}}{\sqrt{\kappa(Q)}}$$

This expression is a *necessary* condition for the problem to be sensitive to α-perturbations. We can thus define the sensitivity measure μ_m for the Matrix Algorithm to be the following estimate of the perturbation to (*start*) needed to change the computed results:

$$\mu_m \equiv \frac{1}{\kappa(Q)}\frac{\delta^{\frac{3}{2}}}{2\sqrt{n}}$$

We summarize the above results in the following theorem, turning the statement of the results around to become a *sufficient* condition for *well*-posedness.

Theorem 7. (*Perturb*)

Suppose we compute the Kalman Decomposition of the system

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u}; \\ \mathbf{y} &= C\mathbf{x}. \end{aligned} \qquad (start)$$

Assume the entries of A, B, C are perturbed by at most α, and, as a consequence, the orthogonal transformations computed in the process of computing the individual controllable or observable decompositions (i.e. where, for example the Staircase Algorithm is used) are perturbed by no more than η (leaving the dimensions unchanged).

Then *the following condition is sufficient to show that (start) is well-posed, i.e. the computed decomposition will not change under such perturbations: for the Matrix Algorithm:*

$$\mu_m \equiv \frac{1}{\kappa(Q)}\frac{\delta^{\frac{3}{2}}}{2\sqrt{n}} \geq \alpha + 2\eta\frac{||A||\sqrt{n}}{\sqrt{\kappa(Q)}}$$

for the Geometric Algorithm:

$$\frac{1}{\kappa_2^2(Q)} \geq 2\eta$$

or

$$\mu_g \equiv |1 - \sigma_{r+1}| \geq 2\eta.$$

Proof: *follows from above discussion.* \$\$\$

We can use the theory of [Stewart 1973b, thm 4.11] to estimate the value of η. A rather direct application of that theorem gives the estimate

$$\eta \leq 2\frac{\alpha}{\delta}. \qquad (\eta estimate)$$

Using this estimate, we get the following corollary:

Corollary 4a. (*Perturb'*)

The following estimates are sufficient to guarantee that the computed decomposition will not change under perturbations to the coefficients of order α:

$$\mu_m \equiv \frac{1}{\kappa(Q)} \frac{\delta^{\frac{3}{2}}}{2\sqrt{n}} \geq \alpha \left(1 + 4 \frac{\|A\|\sqrt{n}}{\delta \sqrt{\kappa(Q)}} \right)$$

for the Geometric Algorithm:

$$\frac{1}{\kappa_2^2(Q)} \geq 4\frac{\alpha}{\delta},$$

or

$$\mu_g \equiv |1 - \sigma_{r+1}| \geq 4\frac{\alpha}{\delta}.$$

Note that μ_m is proportional to $(\kappa(Q^{(m)}))^{-1}$, whereas μ_g is proportional to $(\kappa(Q^{(g)}))^{-2}$. But, before yconcluding that the Matrix Algorithm is better, consider the following theorem, which shows that the bound from the Geometric Algorithm is no worse.

Theorem 8. (*Compare*)

If $Q^{(g)}$, $Q^{(m)}$ are the final computed transformations from the Geometric and Matrix Algorithms, respectively, then there is a transformation $\hat{Q}$ satisfying (DecompCond) (i.e. generating a valid decomposition) such that, in the 1-norm

$$\kappa_1^2(\hat{Q}) \leq 4\kappa_1(Q^{(m)}) \qquad (compare.1)$$

Hence, by Theorem 6. (optimal), it follows that, in the 2-norm

$$\kappa(Q^{(g)})_2 \leq 2n^{\frac{3}{2}} \sqrt{\kappa_1(Q^{(m)})}, \qquad (compare.2)$$

using the equivalence of the norms. Furthermore

$$\mu_g \geq \frac{1}{2n^{\frac{3}{2}}\delta^{\frac{3}{2}}} \mu_m. \qquad (compare.3)$$

Note that in a badly conditioned case, δ will be small, and the coeficient of μ_m will be greater than 1. Thus the Geometric Algorithm would be a more reliable method to use on the nasty examples.

Proof:

We can decompose $Q^{(m)}$ into its components

$$Q^{(m)} = Q^{(2)} S P$$

where $Q^{(2)}$ is the accumulated transformation from the Matrix Algorithm after steps 1 and 2, S is the transformation ($S.3a$) (section A.) applied in step 3, and P is the orthogonal transformation applied in step 4. If we write S and P in blocks along the controllable/uncontrollable split, we find that they have special forms, so that the above can be written as

$$Q^{(m)} = Q^{(2)} \begin{pmatrix} I & R \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & P_2 \end{pmatrix}$$

Define $\hat{Q}$ to be

$$\hat{Q} = Q^{(m)}\begin{pmatrix} I & 0 \\ 0 & \frac{1}{c}I \end{pmatrix} \qquad (hat)$$

where c satisfies

$$c \equiv \|S\|_1 = \left\|\begin{pmatrix} R \\ I \end{pmatrix}\right\|_1 = 1 + \|R\|_1.$$

Since $\mathbf{diag}(I, \frac{1}{c}I)$ commutes with P, we can expand (hat) to get

$$\begin{aligned} Q^{(m)} &= Q^{(2)}\begin{pmatrix} I & R \\ 0 & I \end{pmatrix}\begin{pmatrix} I & 0 \\ 0 & \frac{1}{c}I \end{pmatrix}\begin{pmatrix} I & 0 \\ 0 & P_2 \end{pmatrix} \\ &= Q^{(2)}\begin{pmatrix} I & \frac{1}{c}R \\ 0 & \frac{1}{c}I \end{pmatrix}\begin{pmatrix} I & 0 \\ 0 & P_2 \end{pmatrix} \\ &\equiv Q^{(2)}\hat{S}P \end{aligned}$$

It is evident that $\left\|\hat{S}\right\|_1 = 1$. It also easy to verify that the inverse of $\hat{S}$ is

$$\hat{S}^{-1} = \begin{pmatrix} I & -R \\ 0 & cI \end{pmatrix}$$

so that

$$\begin{aligned} \left\|\hat{S}^{-1}\right\|_1 &= \mathbf{max}\left(1;\ \left\|\begin{pmatrix} -R \\ cI \end{pmatrix}\right\|_1\right) \\ &= \mathbf{max}(1;\ \ c + \|R\|_1) \\ &\leq 2\|S\|_1, \end{aligned}$$

since $\|R\|_1 \leq c = \|S\|_1$. Thus we can conclude that, in the 1-norm,

$$\kappa_1(\hat{Q}) = \kappa_1(\hat{S}) = \left\|\hat{S}\right\|_1\left\|\hat{S}^{-1}\right\|_1 \leq 2\|S\|_1.$$

We should also note that

$$\|S^{-1}\|_1 = \left\|\begin{pmatrix} I & -R \\ 0 & I \end{pmatrix}\right\|_1 = \|S\|_1,$$

so that we can conclude that

$$\kappa_1(Q^{(m)}) = \kappa_1(S) = \|S\|_1^2 \geq \frac{1}{4}\kappa_1^2(\hat{Q}).$$

We must still show that $\hat{Q}$ satisfies $(DecompCond)$. But if we write

$$Q^{(m)} = (Q_{c\bar{c}}^{(m)} \quad Q_{co}^{(m)} \quad Q_{\bar{c}\bar{c}}^{(m)} \quad Q_{\bar{c}o}^{(m)}),$$

then

$$\hat{Q} = (Q_{c\bar{c}}^{(m)} \quad Q_{co}^{(m)} \quad \tfrac{1}{c}Q_{\bar{c}\bar{c}}^{(m)} \quad \tfrac{1}{c}Q_{\bar{c}o}^{(m)}),$$

which clearly satisfies $(DecompCond)$.

The formula $(compare.3)$ follows directly from Theorem 7. $(Perturb)$ and the definition of the measures μ, resulting in the following expression:

$$\mu_g \geq \frac{1}{\kappa^2(Q^{(g)})} \geq \frac{1}{4n^3\kappa(Q^{(m)})} = \frac{1}{2n^{\frac{5}{2}}\delta^{\frac{3}{2}}}\mu_m,$$

where a factor of $\sqrt{n}$ has been added due to the use of the 2-norm instead of the 1-norm. $$$

In the numerical examples used above to illustrate the methods, we obtain the following values for the various parameters:

Parameter	4 by 4	11 by 11
General		
$\|\|A\|\|$	13.778	7.920
Staircase Algorithm		
μ_s	7.450×10^{-3}	5.055×10^{-4}
Matrix Algorithm		
$\kappa_2(Q)$	2.906	2.694
μ_m	1.61×10^{-1}	4.24×10^{-5}
Geometric Algorithm		
$\kappa_2(Q)$	2.414	2.157
μ_g	.2929	.3539

Most of these parameters are computed in the process of computing the decomposition. The two major exceptions are μ_m and δ. The parameter δ is particularly expensive to compute and difficult to estimate. In the programs used to obtain these values, the value of $\frac{1}{\delta} = \left\|T^{-1}\right\|$ was estimated using a scheme described in [Cline et al] that is used in LINPACK (see [LINPACK]) to estimate the condition numbers of linear operators. (Recall that $T(X) \equiv -A_c X + X A_z$.) Once we have computed an estimate for δ, we can then proceed to compute an estimate for μ_m. Estimating η is just as difficult, since it also involves computing δ. Since α is on the order of the machine ϵ, the programs used an arbitrary over-estimate for η of 10^{-7}. This value was then used as the zero-tolerance. Further discussion of the numerical results accompanies the complete tables carried in the appendix.

Epilogue - Summary of Results

In this thesis, we have discussed methods that are used to compute the Kalman Decomposition of

$$\begin{aligned} \dot{\mathbf{x}} &= A\mathbf{x} + B\mathbf{u} \\ \mathbf{y} &= C\mathbf{x} \end{aligned} \qquad (start)$$

where $\mathbf{x}$, $\mathbf{y}$, $\mathbf{u}$ are respectively the state vector, the vector of outputs, and the vector of inputs, all functions of time. The problem is therefore to compute the four parts of the system: controllable and observable, not controllable and observable, controllable and not observable, not controllable and not observable. We have examined four algorithms, two that compute the controllable part, or equivalently the observable part, and two that combine these parts to form the complete Kalman Decomposition for (*start*). The first method to compute the controllable part was the Staircase Algorithm, first fully described in [Van Dooren, Emami-Naeini & Silverman], which is based on the same techniques as the process of reducing a matrix to Upper Hessenberg form by orthogonal similarity transformations. A bound on the sensitivity to possible perturbations was shown to exist. Based on this result, the quantity μ_s, defined as the product of the subdiagonal elements in the upper Hessenberg form, was found to be a measure of the sensitivity of the result to perturbations in the coefficients as estimated by this method, although the computer experiments showed it to be sometimes very pessimistic.

The Modal Method, based on computing the eigen- decomposition of the matrix A in (*start*), has been also described, and measures μ_B and μ_A were defined based on this method. From the computer experiments and the cost analysis, it was discovered that the Modal Method as implemented was more robust than the Staircase Algorithm, and the measure μ_B was less pessimistic than μ_s, but that the Modal Method was almost an order of magnitude more expensive. The difficulties we encountered will be investigated in a future paper.

Next, two methods were discussed which combine the controllable and observable partitions to form the four combination spaces: controllable and observable, not controllable and observable, controllable and not observable, not controllable and not observable. The first of the methods, the Matrix Algorithm, follows the method described in [Kalman] and [Boley]. It involves annihilating the appropriate elements in the coefficient matrices A, B, C in (*start*) by applying similarity transformations so that the four parts may be read off by inspection.

The second method, the Geometric Algorithm, follows the proof of the Canonical Decomposition Theorem in [Desoer], and is described in [Boley, Emami-Naeini & Franklin]. The method is based on the idea of intersecting and extending orthogonal bases for the controllable/ observable spaces involved. It was shown that this method gives the transformation with the smallest spectral condition number of all maps yielding the Kalman Decomposition.

The condition number of the map from the Geometric Algorithm was used to define a sensitivity measure μ_g. In the Matrix Algorithm as well, the controllable and uncontrollable parts must be decoupled by solving an algebraic Lyapunov equation during the reduction process, and the results are used to define a sensitivity measure μ_m. From the computer experiments, it appears that the measures μ_m and μ_g are somewhat pessimistic, but not nearly as much as

μ_s. The Matrix Algorithm breaks down if the eigenvalues of A are not distinct or well separated, but the Geometric Algorithm is not affected by this condition. On the other hand the Matrix Algorithm uses almost half the operations.

In general, computing the Kalman Decomposition depends on successfully computing the rank of certain matrices, and hence is as ill-posed as the rank problem for those matrices. Unfortunately, no method that has been found for computing the Kalman Decomposition is as robust as the Singular Value Decomposition is for determining the rank of a matrix (see discussion on obtaining rank with the S.V.D. in e.g. [Stewart 1973a]).

Appendix. Summary of Numerical Experiments

In this section we summarize the main numerical results in two tables. The first table consists of an example already in the final canonical form (denoted by "final"), the two examples fom the body of the text of this thesis (denoted "e.g.") and four examples with identical splits in increasing order of ill-conditioning (denoted "0443"). In all the tables, the out of range entries are shown in the format $xxx_{\pm}yy$, which is short for $xxx \times 10^{\pm yy}$.

The examples marked "0443" were constructed by taking a system already in canonical form and applying a series of random similarity transformations. The "0443" comes from the splitting of the system, for which the four parts ($C\bar{O}$, CO, $\bar{C}\bar{O}$, $\bar{C}O$) have sizes 0, 4, 4, 3, respectively. The direct effect of the ill-conditioning is the progressively increasing values for the norm of the matrix A, though the eigenvalues remain the same, bounded by 10.

The increasing ill-conditioning shows up in higher and higher condition numbers for the transformation matrices $Q^{(m)}$ and $Q^{(g)}$ and smaller values for the various measures, especially μ_g, μ_m, μ_B, indicating that the problems are becoming more and more sensitive to perturbations.

At the bottom of the table, we show the effects on the modal measures of perturbing the coeficients B or A by 10^{-5} so that the computed dimensions of the controllable and observable spaces are changed. In two cases, the answer was not changed; these are indicated by a star. The values for μ_B and μ_A for these two cases were the same as in the unperturbed case. In the remaining cases, the answer obtained was different (all was controllable/observable), but the measures were small, indicating that the answer was suspect, very sensitive to perturbations.

One can see that some of the parameters are so pessimistic as to be next to useless. In particular, μ_A is smaller than the machine ϵ. This is an artifact of the poor quality of the theoretical estimation rather than possible ill-conditioning in the actual problem. As will be seen in the second table, this also occurs for the parameter μ_s for large systems.

In table 2, we have a series of systems of increasing size, all with four parts ($C\bar{O}$, CO, $\bar{C}\bar{O}$, $\bar{C}O$) of equal size, and all with approximately the same conditioning. All of these examples have only a single input and a single output. Some measure become uselessly pessimistic, like μ_A and μ_s. In fact for large systems, the Staircase Algorithm itself failed to find a distinguishably small subdiagonal element where the matrix could be split. It seems to be an empirical result that this method as presently envisioned will not work for large problems.

The other methods converged on all the problems. The measures μ_B and μ_m both decreased as we go to larger sizes, and the condition number $\kappa(Q^{(m)})$ of the transformation from the Matrix Algorithm increased in an analogous manner. However, these effects seem to be less marked in these cases than in the case of badly conditioned problems. The geometric measure μ_g was only minimally affected by the increasing sizes.

The times shown are for solving the given problems as well as computing the more robust of the measures μ_g and μ_B and all the condition numbers. One can see that they increase approximately as n^3 and that the Geometric Algorithm costs somewhat less than twice the Matrix Algorithm, about as predicted. The positions marked with a star denote cases where the methods

failed to obtain the correct answer, namely four parts of equal size. All the failures were caused by a failure in the Staircase Algorithm, resulting in a longer running time. It is unfortunate that the more reliable method for computing the Controllable Space, the Modal Method, took almost an order of magnitude more time than the Staircase Algorithm took, when they both worked. This is understandable when one considers that the Staircase Algorithm has almost the same cost as a simple QR-decomposition, whereas the Modal Method consists of computing the complete eigensystem and the singular value decomposition of the matrix A, as well as solving a set of linear equations of the same size as A.

case:	final	e.g.	e.g.	0443	0443	0443	0443
size:	4	4	11	11	11	11	11
general:							
$\|\|A\|\|$	7.00	13.8	7.92	13.05	200.7	5751	8424
$\|\|B\|\|$				55.4	298.6	7.468	21.140
$\|\|C\|\|$				5757	4660	120.5	94.98
μ_s	2.88-2	7.44-3	5.05-4	2.09-4	5.43-5	3.63-4	1.76-4
Matrix Algorithm:							
$\kappa_2(Q^m)$	1.64	2.91	2.69	1	48.9	478	6031
$\kappa_1(Q^m)$	2.25	5.85	6.25	1	229	1761	1.37+4
μ_m	9.01-2	1.61-1	4.24-5	7.99-3	7.44-5	1.04-11	9.75-14
Geometric Algorithm:							
$\kappa_2(Q^g)$	1	2.414	2.157	1	5.01	17.5	30.7
μ_g	1	.2929	.354	1	7.65-2	6.52-3	2.12-3
from Modal Method:							
$\frac{1}{\delta}$	.965	.969	120.5	7.09	11.97	9.73+4	4.03+5
μ_B	2.33-1	1.74-1	1.74-1	3.14-1	2.99-2	8.43-3	5.76-3
μ_A	9.83-3	6.03-3	1.23-6	4.16-8	8.56-11	1.01-16	2.66-16
measures when B or A perturbed by 1.0-5:							
$\mu_B(^*B)$				1.44-5	3.06-5	*1	*1
$\mu_A(^*B)$				8.39-12	1.51-15	*1	*1
$\mu_B(^*A)$				5.15-6	3.53-5	1.92-4	3.14-5
$\mu_A(^*A)$				2.88-12	1.73-14	3.53-16	9.88-18

Table 1.

case: equal parts.....									
size:	4	8	16	24	32	40	48	56	64
general:									
$\|\|A\|\|$	51.7	158.6	3190	793.1	970.7	5709	5269	13470	15512
$\|\|B\|\|$	3.05	5.75	30.30	11.32	5.28	8.09	8.52	8.16	2.82
$\|\|C\|\|$	1.79	2.93	5.54	7.92	10.52	12.55	15.00	17.08	18.97
μ_s	2.72-2	7.23-12	8.08-21	2.18-31	1.73-54	9.21-49	1.82-58	*	*
Matrix Algorithm:									
$\kappa_2(Q^m)$	13.8	43.8	28.1	37.7	26.1	38.0	61.6	*	*
μ_m	4.34-3	3.58-4	3.56-6	1.79-5	1.015-5	5.27-6	3.26-6	*	*
Geometric Algorithm:									
$\kappa_2(Q^g)$	2.43	3.55	5.17	5.61	2.82	5.38	4.52	5.11	6.22
μ_g	2.89-1	1.46-1	7.23-2	6.16-2	2.23-1	6.67-2	9.32-2	7.37-2	5.03-2
from Modal Method:									
$\frac{1}{\delta}$	2.59	5.03	115.0	28.4	48.1	53.8	50.6	90.31	60.04
μ_B	2.84-1	1.202-1	1.58-2	1.66-2	2.14-2	1.45-2	1.36-2	5.05-3	7.12-3
μ_A	1.61-5	2.06-6	2.37-12	3.18-10	5.11-10	1.15-10	8.25-12	2.02-13	1.06-12
Times:(secs)									
step:									
stair	.01	.01	.03	.08	.17	.32	.54	.85	1.26
Lyapunov	.00	.005	.024	.06	.14	.23	.38	*.61	*1.70
intersect	.01	.01	.03	.08	.15	.25	.40	.59	.87
modal	.03	.07	.29	.81	1.78	3.28	5.33	7.70	11.18
totals by method:									
Matrix	.02	.03	.09	.21	.47	.82	*1.34	*2.27	*3.32
Geometric	.035	.04	.14	.36	.75	1.42	1.96	*4.28	*4.82
"+Modal	.06	.16	.68	1.88	4.07	7.49	12.24	18.07	26.21

Table 2.

Bibliography

[Björck & Golub]
Björck Å., Golub G. H., "Numerical methods for computing angles between subspaces", *Math. Comp.*, vol 27, pp. 579-594, July 1973.

[Boley]
Boley D. L., "On Kalman's procedure for computing the controllability/observability canonical form", *SIAM j. Control*, Nov. 1980

[Boley, Emami-Naeini & Franklin]
Boley D. L., Emami-Naeini A., Franklin G. F., "A New Algorithm for Canonical Decomposition of Linear Systems", presented at IEEE Conference on Decision and Control, Albuquerque N.M., Dec 10-12, 1980.

[Cline et al]
Cline A. K., Moler C. B., Stewart G. W., Wilkinson J. H., "An estimate for the condition number of a matrix", *SIAM j. Num Anal*, vol 16, pp. 368-375, 1979.

[Daly]
Daly K. C., "The computation of Luenberger canonical forms using elementary similarity transformations", *Int. j. Sys. Sci.*, vol 7, pp 1-15, January 1976.

[Desoer]
Desoer C. A., *A Second Course in Linear Systems*, Van Nostrand Reinhold, 1970.

[Emami-Naeini & Franklin]
Emami-Naeini A., Franklin G. F., "A new algorithm for the canonical decomposition of linear systems", internal memo no. 8001, I. S. L., Stanford University, Dec. 1979.

[Golub]
Golub G. H., "Numerical Methods for solving linear least square problems", *Numer Math*, vol 7, pp. 206-216, 1965.

[Golub & Kahan]
Golub G. H., Kahan W., "Calculating the singular values and pseudo-inverse of a matrix", *SIAM j. Num. Anal. Ser. B*, vol 2, pp 205-224.

[Golub, Nash & Van Loan]
Golub G. H., Nash S., Van Loan C., "A Hessenberg-Schur method for the problem $AX + XB = C$", *IEEE Trans. Auto. Cont.*, vol AC-24, pp. 909-913, Dec. 1979.

[Golub & Reinsch]
Golub G. H., Reinsch C., "Singular value decomposition and least square solutions", *Numer Math*, vol 14, pp. 403-420, 1970.

[Kahan & Davis]
Davis C., Kahan W., "The rotation of eigenvectors by a perturbation", *SIAM j. Num Anal*, vol 7, pp. 1-70, 1970.

[Kailath]
Kailath T., *Linear Systems*, Prentice Hall, Englewood Cliffs N.J., 1980.

[Kalman]
Kalman R. E., "Mathematical description of linear systems", *SIAM j. Control*, Vol 1, no. 2, pp. 152-192, 1963.

[Kalman, Ho & Narendra]
Kalman R. E., Ho Y. C., Narendra K. S., "Controllability of linear dynamic systems", *Contributions to Differential Equations*, vol 1, pp. 189-213, 1961.

[Klema & Laub]
Klema V. C., Laub A. J., "The Singular Value Decomposition: its computation and some applications", *IEEE Trans. Auto. Contr.*, vol 25, no. 2, pp. 164-176, April 1980.

[Konstantinov et al]
Konstantinov M., Petkov P., Christov N., "Synthesis of linear systems with desired equivalent form", j. Comp. and Appl. Math. (Bulgaria), vol 6, pp. 27-35, 1980.

[Laub 1978a]
Laub A. J., "Linear multivariable control, numerical considerations", Invited paper at A.M.S. Short Course on Control and System Theory, Providence, R.I., Aug 1978; also M.I.T. Electronic Systems Lab. rep. ESL-P-833, July 1978.

[Laub 1978b]
Laub A. J., "Computational aspects of the singular value decomposition and some applications", *Sixteenth Annual Allerton Conference on Communication, Control, and Computing*, Oct. 1978.

[Laub & Moore]
Laub A. J., Moore B. C., "Computation of supremal (A, B)-invariant and controllability subspaces", *IEEE Trans. Auto. Contr.*, vol 23, no. 5, pp. 783-792, Oct 1978.

[LINPACK]
Dongarra J. J., Bunch J. R., Moler C. B., Stewart G. W., *LINPACK Users' Guide*, S.I.A.M., Philadelphia 1979.

[Luenberger]
Luenberger D. G., *Introduction to Dynamic Systems*, Wiley, New York 1979.

[Mayne]
Mayne D., *Computational procedure for the numerical realization of transfer function matrices*, Proc IEE, 115, pp. 1363-1368, Sept. 1968.

[Mayne 1973]
Mayne D., "An Elementary Derivation of Rosenbrock's minimal realization algorithm",*IEEE Trans Auto Control*, pp. 306-307, June 1973.

[Paige]
Paige C. C., *Properties of Numerical Algorithms Related to Computing Controllability*, Tech report SOCS 80.4, McGill Univ., Montreal, March 1980.

[Rosenbrock]
Rosenbrock, H. H., *State Space and Multivariable Theory*, Nelson, London 1970.

[Stewart 1973a]
Stewart G. W., *Introduction to Matrix Computations*, Academic Press, New York, 1973

[Stewart 1973b]
Stewart G. W., "Error and perturbation bounds for subspaces associated with certain eigenvalue problems", *SIAM Review*, vol 15 pp. 727-764, 1973.

[Tse, Medanic & Perkins]
Tse E. C. Y., Medanic J. V., Perkins W. R., "Generalized Hessenberg transformations for reduced-order modelling of large-scale systems", *Int. J. Contr.*, vol 27, pp. 493-512, 1978.

[Van Dooren]
Van Dooren P., *The Generalized Eigenstructure Problem*, Ph.D. thesis, Univ So. Cal., Jan 1979.

[Van Dooren, Emami-Naeini & Silverman]
Van Dooren P., Emami-Naeini A., Silverman L., "Stable extraction of the Kronecker construction of pencils", *IEEE Trans. Auto. Contr.*, vol 23, pp. 521-524, 1978.

[Varah]
Varah J. M., "Computing the invariant subspaces of a general matrix when the eigensystem is poorly conditioned", *Math Comp.*, vol 24, pp. 137-149, 1970.

[Wilkinson]
Wilkinson J. H., *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford 1965.

[Wilkinson & Reinsch]
Wilkinson J. H., Reinsch C., *Handbook for Automatic Computation, Vol II. Linear Algebra*, Springer Verlag, Berlin 1971.

[Wonham]
Wonham M. W., *Linear Multivariable Theory: A Geometric Approach*, Springer Verlag, Berlin 1974.

This is Chaotic Confusion and Bluff
That hung on the Turn of a Plausible Phrase
That thickened the Erudite Verbal Haze
Cloaking Constant K
That saved the Summary
Based on the Mummery
Hiding the Flaw
That lay in the Theory Dan *built.*

– adapted from Frederick Winsor