Technische Universität Berlin
Institut für Mathematik

Sensitivity of Computational Control Problems

Nicholas J. Higham, Mihail Konstantinov,
Volker Mehrmann, and Petko Petkov

Preprint 5-2003

Preprint-Reihe des Instituts für Mathematik
Technische Universität Berlin
http://www.math.tu-berlin.de/preprints

Report 5-2003 2003

Sensitivity of Computational Control Problems

Nicholas J. Higham Mihail Konstantinov Volker Mehrmann
Petko Petkov

Introduction

Although numerical methods have been used for many centuries to solve problems in science and engineering, the importance of computation grew tremendously with the advent of digital computers. It became immediately clear that many of the classical analytic and numerical methods and algorithms could not be implemented directly as computer codes, although they were well-suited for hand computations. What was the reason? When doing computations by hand a person can choose the accuracy of each elementary calculation and then estimate, based on intuition and experience, its influence on the final result. In contrast, when computations are done automatically, intuitive error control is usually not possible and the effect of errors on the intermediate calculations must be estimated in a more systematic way. Due to this observation, starting essentially with the works of J. Von Neumann and A. Turing, modern numerical analysis evolved as a fundamental component of machine computation. One of the central themes of this analysis is the solution of computational problems in finite precision (or machine) arithmetic taking into account the properties of both the mathematical problem and the numerical algorithm for its solution. On the basis of such an analysis, numerical methods may be evaluated and compared with respect to the accuracy that can be achieved.

When solving a computational problem on a digital computer, the accuracy of the computed solution generally depends on three major factors:

1. The properties of the *machine arithmetic*—in particular, the rounding unit (or the relative machine precision) and the range of this arithmetic.

2. The properties of the computational problem—in particular, *the sensitivity* of its solution relative to changes in the data, often estimated by the *conditioning* of the problem.

3. The properties of the computational algorithm—in particular, the *numerical stability* of this algorithm.

It should be noted that only by taking into account all three factors are we able to estimate the accuracy of the computed solution.

In this article we will discuss the sensitivity of three important problems in linear control theory that are solved frequently in a number of applications. These problems are pole placement, linear-quadratic optimal control, and optimal H_∞ control. Let us briefly recall these problems.

Consider a linear time invariant dynamical system in state space form

$$\dot{x} = Ax + Bu, \qquad x(t_0) = x^0, \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state at the time t, x^0 is an initial vector, $u(t) \in \mathbb{R}^m$ is the control input of the system and the matrices $A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$ are constant. (Here $\mathbb{R}^{n,m}$ denotes the set of real $n \times m$ matrices). The classical *pole placement problem* is to find a state feedback control law $u = Kx$ such that the closed loop system $\dot{x} = (A + BK)x$ has prescribed poles or, in linear algebra terminology, that the spectrum of the closed loop system matrix $A+BK$ is a given collection $\mathcal{P}$ of complex numbers symmetric with respect to the real axis. For a discussion of the theory of pole placement and related problems, we refer the reader to classical monographs in linear control theory such as [1]. Here, we discuss the conditioning of the pole placement problem. This topic has generated some controversy in the literature and we will bring different viewpoints together.

Another important basic problem in control is the *linear quadratic control problem.* The objective of this problem is to find a control u such that the closed-loop system is asymptotically stable and the performance index

$$\mathcal{S}(u) = \int_0^\infty \begin{bmatrix} x(t) \\ u(t) \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} dt \tag{2}$$

is minimized. Here $Q = Q^T \in \mathbb{R}^{n,n}$, $R = R^T \in \mathbb{R}^{m,m}$ is positive definite and $\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix}$ is positive semidefinite. An important feature of this problem is that the optimal control can be realized as a linear state feedback $u = Kx$. The classical theory for this problem can be found in the monographs [2]–[4]. In the present paper we show that the classical approach of using Riccati equations is not generally the best way to solve this problem.

The third problem included in our discussion is the *optimal H_∞ control problem*, which arises in the context of robust control in the frequency domain, see [5]. In this problem one studies the linear system

$$\begin{aligned} \dot{x} &= Ax + B_1u + B_2w, \quad x(t_0) = x^0, \\ z &= C_1x + D_{11}u + D_{12}w, \\ y &= C_2x + D_{21}u + D_{22}w, \end{aligned} \tag{3}$$

where $A \in \mathbb{R}^{n,n}$, $B_k \in \mathbb{R}^{n,m_k}$, $C_k \in \mathbb{R}^{p_k,n}$ for $k = 1, 2$, and $D_{ij} \in \mathbb{R}^{p_i,m_j}$ for $i, j = 1, 2$. Here $u(t) \in \mathbb{R}^{m_1}$ is the control, $w(t) \in \mathbb{R}^{m_2}$ represents noise, modeling errors or an unknown part of the system, $y(t) \in \mathbb{R}^{p_2}$ describes measured outputs, and $z(t) \in \mathbb{R}^{p_1}$ describes the regulated outputs. The objective of optimal H_∞ control is to find a controller

$$\begin{array}{rcl} \dot{q} & = & \widetilde{A}q + \widetilde{B}y, \\ u & = & \widetilde{C}q + \widetilde{D}y, \end{array} \tag{4}$$

that internally stabilizes the system and minimizes the H_∞-norm of the closed-loop transfer function T_{zw} from w to z. For an explicit formula of the transfer function, see [5]. Although this problem is frequently solved in practice, the sensitivity analysis and the development of reliable numerical methods are far from mature. Consequently, we highlight some of the questions that need to be studied.

The sensitivity of computational problems and its impact on the results of computations are discussed in several textbooks and monographs such as [3],[6],[7].

Basic concepts of numerical analysis

In this section we discuss the three factors that determine the accuracy of the results of a numerical computation in further detail. Readers familiar with floating point arithmetic, conditioning, and stability may proceed to the next section.

Floating point arithmetic

In this subsection we recall some of the basics of floating point arithmetic. A digital computer has only a finite number of internal states and hence it can operate with a finite, although possibly very large, set of numbers called *machine numbers*. As a result, we have the so-called *machine arithmetic*, which consists of the set of machine numbers together with the rules for performing algebraic operations on these numbers.

There are different machine arithmetics, the most widely used being the ANSI/IEEE 754-1985 Standard for Binary Floating Point Arithmetic [8],[6, Chap. 2]. In the following we consider several issues that are essential in every computing environment. For a detailed treatment of this topic see [6]. For simplicity we consider real arithmetic.

Let $\mathbb{M} \subset \mathbb{R}$ be the set of machine numbers. The set $\mathbb{M}$ is finite, contains the zero 0 and is symmetric with respect to 0, i.e., if $x \in \mathbb{M}$ then $-x \in \mathbb{M}$.

In order to map $x \in \mathbb{R}$ into $\mathbb{M}$, *rounding* is used to represent x by the number $\widehat{x} \in \mathbb{M}$ (denoted also as $\mathrm{rd}(x)$), which is closest to x, including a rule for breaking ties when x is equidistant from two machine numbers. Of course, $\widehat{x} = x$ if and only if $x \in \mathbb{M}$. We shall use the hat notation to denote quantities computed in machine arithmetic.

Some strange things may happen in $\mathbb{M}$: an arithmetic operation may not be performed even if the operands are from $\mathbb{M}$; the associative law is violated;and the distributive law may be violated.

Since $\mathbb{M}$ is finite, there is a large positive number $L \in \mathbb{M}$ such that any $x \in \mathbb{R}$ can be approximated in $\mathbb{M}$ if and only if $|x| \leq L$. Moreover, there is a very small positive number

$l \in \mathbb{M}$ such that if $|x| < l$ then $\widehat{x} = 0$ even when $x \neq 0$. We say that a number $x \in \mathbb{R}$ is in the *standard range* of $\mathbb{M}$ if $l \leq |x| \leq L$. In the IEEE double precision arithmetic we have $L \approx 10^{308}, \quad l \approx 10^{-324}$, [8],[6, Chap. 2].

If a number x with $|x| > L$ appears as initial data or as an intermediate result in a computational procedure realized in $\mathbb{M}$, then the computations are usually terminated. This event is called an *overflow* and must be avoided. If a number $x \neq 0$ with $|x| < l$ appears during the computations then it is rounded to $\widehat{x} = 0$, and this event is known as *underflow*. Although not so destructive as overflow, underflow should also be avoided. Over- and underflow may be avoided by appropriate *scaling* of the data.

Example 1 Consider the computation of the norm $y = \|x\| = \sqrt{x_1^2 + x_2^2}$ of the vector $x = [x_1, x_2]^T$, where the data x_1, x_2 and the result y are in the standard range $[l, L]$ of $\mathbb{M}$. In particular, we have $l \leq |x_i| \leq L$. If, however, we have $x_1^2 > L$ then the direct calculation of y will give overflow. Another difficulty arises when $x_1^2 < l$ and $x_2^2 < l$. Then we have the underflow $\mathrm{rd}(x_1^2) = \mathrm{rd}(x_2^2) = 0$ resulting in the wrong answer $\widehat{y} = 0$, while the correct answer is $y \geq l\sqrt{2}$. Overflow may be avoided by using the scaling $\xi_i := x_i/s$, $s := |x_1| + |x_2|$ (provided $s \leq L$) and computing the result from $y = s\sqrt{\xi_1^2 + \xi_2^2}$. Underflow can also be avoided by this scaling (we shall have at least $\widehat{y} \geq l$ when $x_1^2 < l$ and $x_2^2 < l$).

Another important characteristic of $\mathbb{M}$ is the *rounding unit* (*relative machine precision*, or *machine epsilon*), denoted by ε, which is half the distance from 1 to the next larger floating point number. If $1/L \leq |x| \leq L$ then the relative error in the approximation of x by its machine analogue $\widehat{x}$ satisfies the bound

$$\frac{|x - \widehat{x}|}{|x|} \leq \varepsilon.$$

In IEEE double precision arithmetic we have $\varepsilon \approx 1.1 \times 10^{-16}$, which implies that rounding is performed with a small relative error. Most machine arithmetics, including IEEE arithmetic, are built to satisfy the property that arithmetic operations on two numbers are performed accurately in $\mathbb{M}$, with a relative error of order of the rounding unit ε.

Computational problems

The second important feature in assessing the results of computations in finite arithmetic is the formulation of the computational problem. Most problems can be written in explicit form as $y = f(x)$ or in implicit form by means of the equation $\varphi(x, y) = 0$. Here typically the *data* x and the *result* y are elements of vector spaces $\mathcal{X}$ and $\mathcal{Y}$, respectively, and $f : \mathcal{X} \to \mathcal{Y}$, $\varphi : \mathcal{X} \times \mathcal{Y} \to \mathcal{Y}$ are given functions.

Suppose that the data x are perturbed to $x + \delta x$, where the perturbation may result from measurement, modeling, or rounding errors. Then the result y is changed to $y + \delta y$, where $\delta y = f(x + \delta x) - f(x)$. Thus, δy depends on both the data x and its perturbation δx.

The estimation of the *sensitivity* of the problem, i.e., of some quantitative measure $\mu(\delta y)$ of the size of δy as a function of the corresponding measure $\mu(\delta x)$ of δx is the aim of *perturbation analysis* of computational problems. If $x = [x_1, \dots, x_m]^T$ and $y = [y_1, \dots, y_n]^T$ are vectors, then we may use a vector norm, $\mu(x) = \|x\|$ as the quantitative measure.

To illustrate the idea of perturbation analysis we consider the solution of the Lyapunov equation, which is another basic problem in computational control.

Example 2 Consider the Lyapunov equation $A^T X + XA = C$, where A, C, and the solution X are real 6×6 matrices with $C = C^T$ and $X = X^T$. For a particular example of this equation we generated 10000 additive perturbations δc_{11}, δc_{12}, δc_{22} in the corresponding entries of the right hand side C, each of size $10^{-6} \times \|C\|$, and computed the variations δx_{11}, δx_{12}, δx_{22} in the entries of the solution X. In Figure 1 we show the perturbations in the right hand side, the corresponding variations in the solution, and an appropriate sensitivity estimate. The sensitivity estimate is an upper bound on the size of perturbations in the solution and is in the form of a linear estimate $\|\delta X\| \le \beta \|\delta C\|$ for some positive constant β. Clearly, for some directions the corresponding perturbations δC lead to relatively small changes in the solution. Hence the norm-based sensitivity estimate is pessimistic for these particular perturbations.

To derive sensitivity estimates, we need some basic mathematical concepts. Recall that a function $f : \mathcal{X} \to \mathcal{Y}$ is *Lipschitz continuous* at a point $x \in \mathcal{X}$ if there exists $r > 0$ and $M > 0$ such that $\|f(x + \delta x) - f(x)\| \le M \|\delta x\|$ for all $\|\delta x\| \le r$. The smallest quantity

$$M = M(x, r) := \inf \left\{ \frac{\|f(x + \delta x) - f(x)\|}{\|\delta x\|} : \delta x \neq 0,\ \|\delta x\| \le r \right\} \tag{5}$$

is the *Lipschitz constant* of f in the r-neighborhood of x. Lipschitz continuous functions satisfy the perturbation bound

$$\|\delta y\| \le M(x, r) \|\delta x\| \quad \text{for all} \quad \|\delta x\| \le r.$$

A computational problem $y = f(x)$, where f is Lipschitz continuous at x, is *regular* at x; otherwise the problem is *singular*. If x is not in the domain of f, then the problem is singular.

Example 3 Consider the polynomial equation

$$(y - 1)^p = y^p - py^{p-1} + \cdots + (-1)^p = 0,$$

which has a multiple solution $y = 1$. If the constant term $(-1)^p$ is perturbed to $(-1)^p - 10^{-p}$, then the perturbed equation will have p different roots $y_i = 1 + 0.1\varepsilon_i$, $i = 1, \dots, p$, where $\varepsilon_1, \dots, \varepsilon_p$ are the primitive pth roots of 1. Thus a relative change of 10^{-p} in one of the coefficients leads to a relative change of 0.1 in the solution.

In order to characterize when a problem has the property that small changes in the data can lead to large changes in the result, we introduce the concept of condition number. For a regular problem, let $M(x,r)$ be as in (5). Then the number $K(x) := \lim_{r\to 0} M(x,r)$ is called the *absolute condition number* of the computational problem $y = f(x)$. For singular problems we set $K(x) = \infty$.

We have

$$\|\delta y\| \le K(x)\|\delta x\| + \Omega(\delta x), \tag{6}$$

where the scalar quantity $\Omega(h) \ge 0$ satisfies $\Omega(h)/\|h\| \to 0$ for $h \to 0$.

Suppose now that $x \ne 0$ and $y = f(x) \ne 0$. Then setting $\delta_x := \|\delta x\|/\|x\|$, $\delta_y := \|\delta y\|/\|y\|$ we have the bound

$$\delta_y \le k(x)\delta_x + \omega(\delta x), \quad \omega(h) := \Omega(h)/\|y\|,$$

where $\|\omega(h)\|/\|h\| \to 0$ for $h \to 0$ and $k(x) := K(x)\frac{\|x\|}{\|y\|}$ is the *relative condition number* of the problem $y = f(x)$.

Condition numbers can be defined analogously for implicit problems of the form $\varphi(x,y) = 0$, where x is the data and y is the solution.

A regular problem $y = f(x)$ is *well-conditioned* (respectively, *ill-conditioned*) if the relative condition number $k(x)$ is small (respectively, large) in the context of the given machine arithmetic.

The computer solution of an ill-conditioned problem may lead to large errors. In practice, the following rule of thumb may be used for the computational problem $y = f(x)$.

Suppose that $\varepsilon k(x) < 1$. Then one can expect approximately $-\log_{10}(\varepsilon k(x))$ correct decimal digits in the largest components of the computed solution vector y.

Indeed, as a result of rounding the data x we work with $\widehat{x} = x + \delta x$, where $\|\delta x\| \le \varepsilon\|x\|$. If no additional errors are made during the computation, then the computed result is $\widehat{y} = f(\widehat{x})$ and we have

$$\|f(\widehat{x}) - f(x)\| \le K(x)\|\delta x\| + \Omega(x) \le \varepsilon K(x)\|x\| + \Omega(x).$$

Thus the relative error in the computed result satisfies the approximate inequality

$$\frac{\|\widehat{y} - y\|}{\|y\|} \le \frac{\varepsilon K(x)\|x\|}{\|y\|} = \varepsilon k(x).$$

However, this rule of thumb may give pessimistic results, since it describes a worst case situation [9].

Closely related to the sensitivity is the problem of estimating the *distance to the nearest singular problem.* Consider a computational problem $y = f(x)$. The quantity

$$\mathrm{Dist}(f,x) = \min\{\|h\| : \text{the problem } y = f(x+h) \text{ is singular}\},$$

is the *absolute distance* to singularity of the problem $y = f(x)$. Similarly, for $x \ne 0$, the quantity $\mathrm{Dist}(f,x)/\|x\|$ is the *relative distance* to singularity of the problem. For many problems the relative distance to singularity and the relative condition number of the problem are inversely proportional [9].

Example 4 The problem of solving the linear system $Ay = b$ with a square matrix A and data $x = (A, b)$ is regular if and only if the matrix A is nonsingular. The relative distance to singularity for an invertible matrix A is $1/\text{cond}(A)$, where $\text{cond}(A) := \|A\| \, \|A^{-1}\|$ is the relative condition number of A relative to inversion [6, Thm. 6.5].

Another difficulty is the mathematical representation of the computational problem that needs to be solved. In particular, in control theory, several different frameworks are used. A classical example for such different frameworks is the representation of linear systems via matrices and vectors, as in the classical state space form (1), as rational matrix functions (via the Laplace transform), or even in a polynomial setting [10],[11]. These different approaches have different mathematical properties and taste often determines which framework is preferred.

From a numerical point of view, however, the chosen approach is typically not a matter of taste, since the sensitivity is drastically different. Numerical analysts usually prefer the matrix/vector setting over polynomial or rational functions, while for users of computer algebra systems the polynomial or rational approach is often more attractive. The reason for the preference for the matrix/vector approach in numerical methods is that the sensitivity of the polynomial or rational representation is usually higher than that of a matrix/vector representation. This fact is often ignored in choosing frameworks that are mathematically more elegant but numerically inadequate.

Example 5 [12] Consider the computation of the eigenvalues of the matrix $A = Q^T \text{diag}(1, 2, \ldots, 20)Q$, where Q is a random orthogonal matrix. Clearly the matrix is symmetric and therefore diagonalizable with nicely separated eigenvalues $1, 2, \ldots, 20$. The problem of computing the eigenvalues of A is well-conditioned, and numerical methods such as the symmetric QR algorithm lead to highly accurate results, see [13]. For example, `eig` from MATLAB [14] yields all eigenvalues to at least 15 correct digits.

The usual textbook approach for computing eigenvalues taught in first year linear algebra is that the eigenvalues of A are the roots of the characteristic polynomial $\det(\lambda I - A) = (\lambda - 1)(\lambda - 2) \cdots (\lambda - 20)$. Using a numerical method such as `roots` from MATLAB to compute the roots of this polynomial, however, yields highly inaccurate large eigenvalues 20.0003, 18.9970, 18.0117, 16.9695, 16.0508, 14.9319, 14.0683, 12.9471, 12.0345, 10.9836, 10.0062, 8.9983, 8.0003. The accuracy of the small eigenvalues is slightly better. There are several reasons for the inaccuracy. First, the coefficients of the polynomial range in the interval $[1, 20!] \approx [1, 2.4 \times 10^{18}]$ and cannot all be represented accurately in IEEE double precision arithmetic, while the entries of the matrix range in the ball of radius 20 around the origin. Second, the sensitivity of the larger roots with respect to perturbations in the coefficients is very large in this case.

In this section we have discussed the sensitivity and conditioning of a computational problem. This sensitivity is a property of the problem and its mathematical representation in the context of the machine arithmetic used, and should not be confused with the properties of the computational method that is implemented to solve the problem. In practice, linear sensitivity estimates of the type $\delta_y \le k(x)\delta_x$ are usually used, occasionally leading to underestimation of the actual perturbation in the solution. Rigorous perturbation bounds can be derived by using non-linear perturbation analysis [15].

Computational algorithms

In this subsection we discuss properties of computational algorithms and the accuracy of the computed result.

An algorithm for computing $y = f(x)$ is a decomposition

$$f = F_r \circ F_{r-1} \circ \cdots \circ F_1, \tag{7}$$

which gives a sequence $x_k = F_k(x_{k-1})$, $k = 1, \ldots, r$, with $x_0 = x$ and $y = x_r$. Although the computation of $F_k(\xi)$ requires simple algebraic operations on ξ such as arithmetic operations or taking roots, the computation may also be a more complicated subproblem such as solving a system of linear equations or computing the eigenvalues of a matrix.

The algorithm either gives the exact answer in exact arithmetic or, for some problems, such as eigenvalue problems or the solution of differential equations, gives an approximate answer in exact arithmetic. We will not analyze the latter case here, but rather will investigate what happens with the computed value of x_r when the computations are done in machine arithmetic.

It is important to mention that two different algorithms, say (7) and $f = \Phi_s \circ \Phi_{s-1} \circ \cdots \circ \Phi_1$ for computing $y = f(x)$, may give completely different results in machine arithmetic, although in exact arithmetic they are equivalent.

In what follows, we suppose that the data x is in the standard range of the machine arithmetic with characteristics L, l, ε, and that the computations do not lead to overflow or to a destructive underflow. As a result, the answer computed by the algorithm (7) is $\widehat{y}$. Our goal is to estimate the absolute error $E := \|\widehat{y} - y\|$ and the relative error $e := E/\|y\|$ (for $y \neq 0$) of the computed solution $\widehat{y}$ in the case of a regular problem $y = f(x)$ when x belongs to a given set X_0.

Definition 1 *[12],[6],[3] The algorithm (7) is* numerically stable *on the set X_0 if the computed quantity $\widehat{y}$ for $y = f(x)$, $x \in X_0$, is close to the solution $f(\widehat{x})$ of a problem with data $\widehat{x}$ near to x in the sense that*

$$\|\widehat{y} - f(\widehat{x})\| \le \varepsilon a \|y\|, \quad \|\widehat{x} - x\| \le \varepsilon b \|x\|, \tag{8}$$

where the constants $a, b > 0$ do not depend on $x \in X_0$.

For a problem with inexact data, perhaps itself being subject to rounding errors, numerical stability is in general the most we can ask of an algorithm. If in Definition 1 we take $a = 0$, then the algorithm is *numerically backward stable.* Backward error analysis, introduced by Wilkinson [12], can be used to show that the solution computed by an algorithm is the exact solution of a perturbed problem, where the perturbation is the *equivalent data error.*

As in [3], using the inequalities (8) and $\|f(x+\delta x) - f(x)\| \leq K(x)\|\delta x\| + \Omega(\delta x)$ (see (6)), we obtain the absolute error estimate

$$\begin{aligned} E := \|\widehat{y} - y\| &= \|\widehat{y} - f(\widehat{x}) + f(\widehat{x}) - f(x)\| \\ &\leq \|\widehat{y} - f(\widehat{x})\| + \|f(\widehat{x}) - f(x)\| \\ &\leq \varepsilon a\|y\| + K(x)\|\widehat{x} - x\| + \Omega(\widehat{x} - x) \\ &\leq \varepsilon a\|y\| + \varepsilon b K(x)\|x\| + \Omega(\widehat{x} - x). \end{aligned}$$

Dividing by $\|y\|$ yields the relative error estimate

$$e := \frac{\|\widehat{y} - y\|}{\|y\|} \leq \varepsilon \left(a + bK(x)\frac{\|x\|}{\|y\|} + \frac{\omega(\widehat{x} - x)}{\varepsilon} \right).$$

Since $\omega(\widehat{x} - x)/\varepsilon \to 0$ for $\varepsilon \to 0$, by ignoring this term, we have the approximate estimate

$$e \leq \varepsilon \left(a + bK(x)\frac{\|x\|}{\|y\|} \right) = \varepsilon(a + bk(x)) \tag{9}$$

for the relative error in the computed solution.

Inequality (9) clearly shows the influence of the three major factors that determine the accuracy of the computed solution:

- the machine arithmetic (the rounding unit ε and implicitly the range of $\mathbb{M}$ through the requirement to avoid over- and underflow);
- the computational problem (the relative condition number $k(x)$);
- the computational algorithm (the constants a and b).

Inequality (9) is an example of a *condition number based accuracy estimate* for the solution, computed in machine arithmetic. In order to assess and trust the accuracy of results, condition and accuracy estimates should accompany every computational procedure. Many modern software packages provide such estimates [16],[17]. However, it is unfortunately common practice in industrial use to turn these facilities off, even though this service will warn the user of numerical methods about possible failure.

As we have seen in (7), computational problems are typically modularized, for example, they can be decomposed and solved as a sequence of subproblems. This decomposition facilitates the use of computational modules and is one of the reasons for the success of numerical analysis. One should be aware, however, that modularization can lead to substantial numerical difficulties. Such difficulties arise if one or more of the created subproblems F_i is ill-conditioned or singular.

Example 6 The scalar identity function $y = f(x) = x$ may be decomposed as $f = F_2 \circ F_1$, where $F_1(x) = x^3$ and $F_2(z) = z^{1/3}$. Here the function F_2 is not Lipschitz continuous at 0.

But even if the functions F_1, F_2 are Lipschitz continuous with constants K_1, K_2 respectively, then it may happen that one (or both) of these constants is large. We obtain the estimate

$$\begin{array}{rcl} \|f(x+h)-f(x)\| & = & \|F_2(F_1(x+h))-F_2(F_1(x))\| \\ & \leq & K_2\|F_1(x+h)-F_1(x)\| \leq K_2K_1\|h\|, \end{array}$$

where the quantity K_2K_1 may be much larger than the actual Lipschitz constant K of f.

Example 7 Consider the identity function $y = f(x) = x$ in $\mathbb{R}^2$. Define $F_1(x) = A^{-1}x$ and $F_2(z) = Az$, where the matrix $A \in \mathbb{R}^{2,2}$ is nonsingular. Then $K = 1$ while both $K_1 = \|A^{-1}\|$ and $K_2 = \|A\|$ may be arbitrarily large. If the computations are carried out with maximum achievable accuracy, then the computed value for $A^{-1}x$ is $\widehat{F}_1(x) = (I_2 + E_1)A^{-1}x$, where $E_1 := \mathrm{diag}(\varepsilon_1, \varepsilon_2)$ and $|\varepsilon_1|, |\varepsilon_2| \leq \varepsilon$. Similarly, the computed value for $A(A^{-1}x)$ becomes $(I_2 + E_2)A\widehat{F}_1(x) = (I_2 + E_2)A(I_2 + E_1)A^{-1}x$, where $E_2 := \mathrm{diag}(\varepsilon_3, \varepsilon_4)$ and $|\varepsilon_3|, |\varepsilon_4| \leq \varepsilon$. Suppose that $\varepsilon_1 = -\varepsilon_2 = \varepsilon \simeq 10^{-16}$, $\varepsilon_3 = \varepsilon_4 = 0$ and

$$A = \left[\begin{array}{cc} a & a+1 \\ a-1 & a \end{array}\right], \; x = \left[\begin{array}{c} 1 \\ -1 \end{array}\right],$$

where $a = 10^8$. Then the computed result is $\widehat{x} = x + \varepsilon\xi$, where $\xi = [\xi_1, \xi_2]^T$ and $\xi_1 = 4a^2 + 2a - 1$, $\xi_2 = 4a^2 - 2a - 1$. Thus, the actual relative error in the solution of the decomposed problem is $\varepsilon\frac{\|\xi\|}{\|x\|} \simeq 4a^2\,\varepsilon \simeq 4$, and there are no correct digits in the computed result.

In this section we reviewed some of the general principles of numerical analysis. In the following sections we look at three basic problems in control theory and analyze their sensitivity.

Pole Placement

Pole placement is an important tool for many applications in modern control theory. In linear algebra terminology, the pole placement problem is as follows.

Problem 1 *For a given pair of matrices $S = (A, B)$ with $A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$ and a given collection of n complex numbers $\mathcal{P} = \{\lambda_1, \ldots, \lambda_n\} \subset \mathbb{C}$ (closed under conjugation), find a matrix $K \in \mathbb{R}^{m,n}$ such that the collection of eigenvalues of $A + BK$ is equal to $\mathcal{P}$.*

It is well known, see [1], that a *feedback gain* matrix K exists for all collections $\mathcal{P} \subset \mathbb{C}$, (symmetric relative to the real axis) if and only if (A, B) is *controllable*, i.e., $\mathrm{rank}[A - \lambda I_n, B] = n$, for all $\lambda \in \mathbb{C}$. There is a large literature on the numerical solution of this problem, see [18]–[21]. Even though numerical backward stability has been shown for some of these methods, see [19],[20],[22], it is often observed that the numerical results are inaccurate. In view of our discussion, if a numerically stable method yields highly inaccurate results, then this inaccuracy must be due to the ill-conditioning of the problem. The analysis of the conditioning of the pole placement problem, however, led to differing conclusions, see [23]–[26].

Since controllability is a requirement for the ability to assign arbitrary sets of poles, it must be expected that numerical difficulties arise when the problem is very near to an uncontrollable problem. The *distance to uncontrollability* is defined as the minimum of the quantity $\|[\delta A, \delta B]\|$, where the pair $(A+\delta A, B+\delta B)$ is uncontrollable, see [27]. A bound for this distance may be determined by computing $\min_{\lambda \in \mathcal{C}} \sigma_n[A-\lambda I, B]$, see [27], where $\sigma_n[A-\lambda I, B]$ denotes the smallest singular value of the matrix $[A-\lambda I, B]$.

As we have seen, numerical problems can also arise when a problem is approached by means of a multi-step procedure, where an intermediate step is ill-conditioned. For example, pole placement is usually a two-step procedure, which first brings the pair (A, B) to a simpler form [3],[28] and then assigns the poles in this simpler form. To evaluate of a particular numerical method, the conditioning of both subproblems needs to be analyzed.

If one studies the literature of the pole placement problem, this ill-conditioning is only partially reflected and the discussion is quite controversial. This controversy has several sources which have to do with the non-uniqueness of the solution in the multi-input case, and also with the representation of the data. Another reason for confusion in the analysis of the pole placement problem is that one has to define what the *solution* of the problem is. Theoretically, this is the feedback matrix K, or the set of all such matrices. But relative to the computed solution, there are three different issues. First, there is the computed value $\widehat{K}$ of K. Second, we have the closed-loop matrix $A+B\widehat{K}$ or its rounded value $\mathrm{rd}(A+B\widehat{K})$. And third, we have the resulting spectrum of $A+B\widehat{K}$, which should be equal to $\mathcal{P}$ but usually differs from $\mathcal{P}$. Although all of these quantities are computed "solutions" of the pole placement problem, they exhibit largely different perturbation behavior. We will now summarize these different viewpoints.

Perturbation analysis for the gain matrix consists of determining bounds on the change δK in the gain matrix K as a function of the changes $\delta A, \delta B$ in the system matrices A, B and the changes $\delta\lambda_1, \ldots, \delta\lambda_n$ in the desired poles. In this case, whether or not the closed-loop system matrix $A+BK$ or its spectrum is sensitive is not the subject of sensitivity analysis. Of course, in the multi-input case it is possible to use the $n(m-1)$-parametric freedom (if the rank of B is m) in the gain matrix K to minimize some measure of the sensitivity of the eigenstructure of $A+BK$, or to achieve other design purposes, such as minimizing $\|K\|$, or maximizing the stability radius of $A+BK$, see [18],[21],[29]. Since for $m > 1$ the gain matrix K lies in an unbounded $n(m-1)$–dimensional algebraic variety in $\mathbb{R}^{m,n}$, the sensitivity analysis must guarantee that there exists at least one solution to the perturbed problem for which the perturbation bounds for δK hold. At the same time both the original and perturbed problems may have solutions of arbitrary large norm. Explicit perturbation bounds for K, both local and non-local, have been derived in [24].

Let $\Lambda := \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ and $\delta\Lambda := \mathrm{diag}(\delta\lambda_1, \ldots, \delta\lambda_n)$. An estimate in terms of relative perturbations $\delta_K := \|\delta K\|_F/\|K\|_F$ is given by

$$\delta_K \leq c_A\delta_A + c_B\delta_B + c_\Lambda\delta_\Lambda + \mathrm{O}(\|\delta\|^2),$$

where $c_A := C_A\|A\|_F/\|K\|_F$, $c_B := C_B\|B\|_F/\|K\|_F$, and $c_\Lambda := C_\Lambda\|\Lambda\|_F/\|K\|_F$ are the relative condition numbers with respect to the perturbations in A, B, Λ, respectively, $\delta_A := \|\delta A\|/\|A\|_F$, $\delta_B := \|\delta B\|_F/\|B\|_F$, and $\delta_\Lambda = \|\delta\Lambda\|_F/\|\Lambda\|_F$. Here C_A, C_B, C_Λ are the corresponding absolute condition numbers and $\delta := [\delta_A, \delta_B, \delta_\Lambda]^T$. This analysis shows that the problem of computing the feedback gain K is well- or ill-conditioned if the overall relative condition number

$$c_{PA} := c_A + c_B + c_\Lambda \tag{10}$$

is small or large in the context of the machine arithmetic used [24].

In general, the sensitivity of the computation of K does not depend substantially on the desired spectrum $\mathcal{P}$. At the same time, the eigenstructure (the eigenvalues in particular) of the matrix $A + BK$ may be sensitive to perturbations in the data. As a result, the spectrum of the perturbed closed-loop system matrix $A + B\widehat{K}$ may be far from $\mathcal{P}$, even if $\widehat{K}$ (the computed value for K) is obtained by a numerically stable algorithm or even exactly.

Example 8 [25] Let $A = \mathrm{diag}(1, \ldots, 20)$, $\mathcal{P} = \{-1, \ldots, -20\}$, let B be formed from the first m columns of a random 20×20 orthogonal matrix. The MATLAB pole placement code `place` of the *Control System Toolbox* Version 4.1, which is an implementation of the method given in [18], was used to compute the feedback gain K. For m from 1 to 10 the feedback was computed 20 times with 20 random matrices B with orthonormal columns. In Table 1 the geometric means (over the 20 experiments) of the norm of the computed feedback matrix $\widehat{K}$ and err=$\max_{1\leq i\leq 20}|\widehat{\lambda}_i - \lambda_i|$ are listed, with λ_i and the real parts of the resulting poles $\widehat{\lambda}_i$ arranged in increasing order.

For all 400 tests the pair (A, B) was controllable with a large distance to uncontrollability. Nevertheless, for $m = 1$ the method produced an error message "Can't place eigenvalues there" and, for $m = 2, 3$, a warning "Pole locations are more than 10% in error" was displayed. Other pole placement algorithms have similar difficulties for small m, see [25],[26]. The eigenvalues of the closed-loop system are highly sensitive and their computed values may have positive real parts regardless of how the feedback is computed. If the data of the problem are slightly perturbed, for example due to measurement errors, then the resulting feedback design may fail completely.

Analysis of the sensitivity of the spectrum and the eigenvectors of the closed-loop matrix $A+BK$ has been carried out in [25],[26]. The major factors in the conditioning of the closed loop spectrum include the norm of K, the distance to uncontrollability, and the condition number of the closed loop eigenvector matrix. We have the following possibilities.

- The gain matrix K is very sensitive, for example, since the distance to uncontrollability is small. A small change in A, B may lead to a large difference between $\widehat{K}$ and K. In general, this difference will result in large errors for the eigenvalues of the computed closed-loop system matrix.

Table 1: Norms of feedback gain matrix and error in assigned spectrum for different values of m.

m	$\widehat{K}$	err
2	2.5×10^6	2.0×10^1
3	1.3×10^6	1.2×10^1
4	2.3×10^5	1.2×10^{-3}
5	3.4×10^5	1.6×10^{-6}
6	1.0×10^4	3.1×10^{-8}
7	4.2×10^3	1.3×10^{-9}
8	2.1×10^3	1.3×10^{-10}
9	1.1×10^3	1.9×10^{-11}
10	8.9×10^2	6.3×10^{-12}

– The norm of the gain matrix K is very large. Then the difference $\|\widehat{K} - K\|$, which is of order at least $\varepsilon\|K\|$, may also be large, and this gain perturbation will perturb the eigenvalues of $A + B\widehat{K}$.

– The eigenvalues of $A + BK$ are very sensitive to perturbations for any (or for the particular) choice of K. This situaton occurs, for example, in the case of dead-beat control of discrete-time systems $x(t + 1) = Ax(t) + Bu(t)$, where the closed-loop poles are all equal to zero and contained in the same Jordan block. Here, the perturbations in the eigenvalues of $A + BK$ may be of order $\eta^{1/n}$, where η is the size of the perturbations in the data.

These three factors are all independent and may appear alone or in some combination. Moreover, in some cases the minimum sensitivity of the gain matrix is achieved exactly when the eigenstructure of the closed-loop system matrix is maximally sensitive.

Example 9 Consider the pole placement problem for the case $n = 2$, $m = 1$ with

$$A = \begin{bmatrix} \lambda_1 & 1 \\ \beta & \lambda_2 \end{bmatrix}, \; B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

If the desired poles are λ_1, λ_2, then $K = [-\beta, \; 0]$ yields

$$C_\Lambda = \sqrt{1 + 2\mu + \sqrt{1 + 4\mu^4}}, \; C_A = \sqrt{2 + 4\mu^2}, \; C_B := \mu + \sqrt{1 + \mu^2},$$

where $\mu := |\lambda_1 - \lambda_2|/2$. Here the minimum sensitivity of K is achieved for $\lambda_1 = \lambda_2$, which corresponds to maximum sensitivity of the closed-loop poles, since this is the worst case in the perturbation theory for eigenvalues [13].

Example 10 In this example based on [24], we study the overall relative condition number c_{PA} in (10) for computing K for the controllable pair of matrices

$$A = \begin{bmatrix} 0 & 3 & 0 & 4 & 0 & -7 \\ -9 & 0 & -3 & 0 & 7 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 \\ -4 & 0 & -1 & 0 & 4 & 0 \\ 0 & 3 & 0 & 4 & 0 & -7 \\ -9 & 0 & -2 & 0 & 8 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ -1 \\ 0 \end{bmatrix}.$$

We take $\lambda_3 = \cdots = \lambda_6 = -1$ and vary λ_1, λ_2. In Figure 2 we show the dependence of c_{PA} on the real part σ and imaginary part ω of $\lambda_1 = \sigma + j\omega, \lambda_2 = \sigma - j\omega$. We see that the computation of K remains well conditioned for large variations in λ_1, λ_2. The minimum of the overall conditioning is achieved for λ_1, λ_2 near to -1. Choosing all desired poles equal to -1 yields the gain matrix $K = [-6.3,\ -2.3,\ -0.7,\ -3.1,\ 5.3,\ 5.35]$ and the relative condition numbers $c_\Lambda = 1.420$, $c_A = 37.27$, and $c_B = 2.360$.

In Figure 3 we show the distribution of the closed-loop poles (the so-called *pseudospectrum*) for 2000 perturbations in $A + BK$ of norm 10^{-8}, computed by the function `ps` from Matrix Computation Toolbox [30]. Clearly, the large sensitivity of the closed-loop poles is not related to the conditioning of computing K.

So far, we have mentioned only the nonuniqueness of the choice of K in the multi-input case. There are several possibilities for using this freedom to optimize a robustness measure: one could minimize $\|K\|$, see [19],[31], or the stability radius of $A + BK$, or the condition number of the closed-loop eigenvector matrix as in [18] (in this case the poles must be pairwise distinct), or the feedback norm and the eigenvalue sensitivity together [21]. In general, one should ask the following question.

Does one really have a fixed collection of poles or rather, does one have a specific region in the complex plane where one wants the closed loop poles to be?

If the latter is the case, then not only the minimization over the freedom in K but also a minimization over the position of the poles in the given set should be used, leading to the *optimized pole placement problem* [29],[32], see [33] for such an approach.

Problem 2 For given matrices $A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$ and a given set $\mathcal{P} \subset \mathbb{C}$, find a matrix $K \in \mathbb{R}^{m,n}$, such that the eigenvalues of $A + BK$ are contained in $\mathcal{P}$ and at the same time some robustness measure is optimized.

A clear and practical formulation of a general robustness measure, as well as suitable algorithms for determining the optimal pole assignment, depend on both the application and the set $\mathcal{P}$. In the stabilization problem $\mathcal{P}$ is the left half plane, or in the case of damped stabilization a subset of the left half plane. If the set $\mathcal{P}$ is too small, such as when it has exactly n points, then optimizing a robustness measure may still yield a sensitive closed loop spectrum, but if the set $\mathcal{P}$ is large, then better results may be obtained. The general sensitivity analysis for this optimized pole placement problem is an open problem.

Linear-quadratic control

In this section we discuss the linear-quadratic control problem of minimizing (2) subject to (1). Application of the maximum principle [2] leads to the equivalent problem of finding an asymptotically stable solution to the two-point boundary value problem of Euler-Lagrange equations

$$\mathcal{E}_c \begin{bmatrix} \dot{x} \\ \dot{\mu} \\ \dot{u} \end{bmatrix} = \mathcal{A}_c \begin{bmatrix} x \\ \mu \\ u \end{bmatrix}, \qquad x(t_0) = x^0, \qquad \lim_{t\to\infty} \mu(t) = 0, \tag{11}$$

with the matrix pencil

$$\alpha\mathcal{E}_c - \beta\mathcal{A}_c := \alpha \begin{bmatrix} I & 0 & 0 \\ 0 & -I & 0 \\ 0 & 0 & 0 \end{bmatrix} - \beta \begin{bmatrix} A & 0 & B \\ Q & A^T & S \\ S^T & B^T & R \end{bmatrix} \tag{12}$$

and the Lagrange multiplier (costate) μ.

If R is well-conditioned with respect to inversion, then (11) may be reduced to the two-point boundary value problem

$$\begin{bmatrix} \dot{x} \\ -\dot{\mu} \end{bmatrix} = \mathcal{H} \begin{bmatrix} x \\ -\mu \end{bmatrix}, \qquad x(t_0) = x^0, \qquad \lim_{t\to\infty} \mu(t) = 0, \tag{13}$$

with the *Hamiltonian matrix*

$$\mathcal{H} = \begin{bmatrix} F & G \\ H & -F^T \end{bmatrix} := \begin{bmatrix} A - BR^{-1}S^T & BR^{-1}B^T \\ Q - SR^{-1}S^T & -(A - BR^{-1}S^T)^T \end{bmatrix}.$$

These different mathematical representations for computing the optimal control exhibit different sensitivity.

The classical apporach to solving the boundary value problems (11) and (13) [2],[4], which is implemented in most design packages, is a two-step procedure. One computes first X, the positive semidefinite (stabilizing) solution of the associated algebraic Riccati equation

$$0 = H + XF + F^TX - XGX, \tag{14}$$

and then obtains the optimal stabilizing feedback as $u = -R^{-1}(S^T + B^TX)x$.

Another consideration is the deflating subspace approach of Van Dooren [34]. Suppose $(\mathcal{E}_c, \mathcal{A}_c)$ has an n-dimensional deflating subspace associated with eigenvalues in the left half plane. Let this subspace be spanned by the columns of a matrix

$$\mathcal{U} := \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix}. \tag{15}$$

Then, if U_1 is invertible, the optimal control is a linear feedback of the form $u = Kx = U_3U_1^{-1}x$. The solution of the associated Riccati equation (14) is then $X = U_2U_1^{-1}$, see [2] for details. In this case the solution of the Riccati equation is not needed to determine the feedback.

By analogy with the discussion of the pole placement problem, we first consider the distance to the nearest singular problem. The requirement that the closed-loop system be asymptotically stable leads to the requirement that the system (1) is *stabilizable*, i.e. $\operatorname{rank}[A-\lambda I_n, B] = n$, for all λ in the closed right half plane. The *distance to unstabilizability* is defined as the minimum of the quantity $\|[\delta A, \delta B]\|$, such that the pair $(A+\delta A, B+\delta B)$ is not stabilizable. This distance can be determined by studying the smallest perturbation so that the matrix pencil (12) ceases to have exactly n finite eigenvalues in the open left half complex plane, and hence we have to discuss the perturbation theory of eigenvalues and invariant subspaces of matrix pencils. Such analysis is beyond the scope of this paper; see [35],[36] for detaileds

It is clear that the three approaches to determining the feedback gain K may lead to different numerical results due to the different sensitivities of the subproblems. For example, we see that in order to use the representation (13), the invertibility of R is required, and thus it is clear that the sensitivity of the computation of $K = U_3U_1^{-1}$ is different from that of the procedure of first computing $X = U_2U_1^{-1}$ and then forming $K = -R^{-1}(S^T + B^TX)$. Consider the following example.

Example 11 [32] Let U be a randomly generated real orthogonal matrix, let $S = 0$, and let

$$A = U \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} U^T, \; B = U, \; R = \begin{bmatrix} 0.5 & 0 \\ 0 & \gamma \end{bmatrix}, \; Q = U \begin{bmatrix} 6 & 0 \\ 0 & 3\gamma \end{bmatrix} U^T,$$

where $\gamma > 0$. The stabilizing solution of the Riccati equation (14) and the associated feedback are given by

$$X = U \begin{bmatrix} 3 & 0 \\ 0 & 3\gamma \end{bmatrix} U^T, \; K = -\begin{bmatrix} 6 & 0 \\ 0 & 3 \end{bmatrix} U^T,$$

and the resulting closed loop spectrum is $\{-4, -2\}$. Since both K and the spectrum are independent of the value of γ and since U is orthogonal, we see that the spectral norm $\|K\|_2 = 6$ is small and hence we do not expect large perturbations in the solution X. The solution procedure based on the Riccati equation, however, depends on γ.

In Table 2 we compare the accuracy of the results obtained by the MATLAB function `care` from the MATLAB Control Toolbox [14], which is a solver for algebraic Riccati equations and those obtained by computing the deflating subspace by the MATLAB function `qz`. The Riccati solution is used to compute $K = -R^{-1}B^TX$ while, by using the deflating subspace (15) of $\alpha\mathcal{E}_c - \beta\mathcal{A}_c$, the feedback K is directly obtained as $U_3U_1^{-1}$. The relative error in X and K for the two methods as a function of different values of γ are listed in Table 2. We see that direct computation of the optimal control based on computation of the invariant subspace (using `qz`) yields smaller relative errors than the solution based on the Riccati equation (using `care`).

Table 2: Comparison of Riccati approach and subspace approach.

γ	Method	$\frac{\Vert\widehat{X}-X\Vert_2}{\Vert X\Vert_2}$	$\frac{\Vert\widehat{K}-K\Vert_2}{\Vert K\Vert_2}$
10^{-2}	`care`	7.0×10^{-16}	1.3×10^{-15}
	`qz`	2.4×10^{-16}	4.9×10^{-15}
10^{-6}	`care`	3.1×10^{-12}	3.2×10^{-9}
	`qz`	2.6×10^{-15}	4.7×10^{-11}
10^{-9}	`care`	2.1×10^{-8}	1.3×10^{-4}
	`qz`	1.6×10^{-15}	5.9×10^{-9}
10^{-13}	`care`	9.2×10^{-5}	3.9×10^{1}
	`qz`	1.7×10^{-15}	5.0×10^{-4}

As in the pole placement problem, we also have to ask what constitutes a solution to the problem. This solution could be the feedback gain $K = -R^{-1}B^TX = U_3U_1^{-1}$ or the closed loop matrix $A+BK$ or its spectrum. Examples 8 and 9 (which can be constructed to come from optimal control) show that these may have very different sensitivity.

The discussion demonstrates the importance of analyzing the sensitivity of the computational problem, and that a different modularization of the computational problem can lead to significantly different results. We see that the solution of the linear-quadratic control problem based on the solution of the algebraic Riccati equation presents a dangerous detour that may lead to poor results. However, this detour is not necessary, since the feedback and the closed-loop matrix may be computed from the deflating subspace. The situation is worse in the case of descriptor systems, see [2], [37], where the Riccati equation may be unrelated to the solution of the optimal control problem.

On the other hand, the Riccati equation approach is well analyzed, and efficient numerical software for the solution of algebraic Riccati equation is available, while the development of structure-preserving solution methods for the eigenvalue problem (12) has not yet matured, [37]. We need to be able to judge when the conditioning of the Riccati equation is worse than the conditioning of the optimization problem itself. Therefore, we now discuss the conditioning of the algebraic Riccati equation (14). We assume that there exists a non-negative-definite solution X such that $F - GX$ is stable.

Let the coefficient matrices F, G, H in (14) be subject to perturbations δF, δG, δH, respectively, so that, instead of the initial data, we have the matrices $\widetilde{F} = F + \delta F$, $\widetilde{G} = G + \delta G$, and $\widetilde{H} = H + \delta H$. The aim of perturbation analysis of (14) is to investigate the variation δX in the solution $\widetilde{X} = X + \delta X$ due to the perturbations δF, δG, δH. It is assumed that the data perturbations preserve the symmetric structure of the equation, i.e., the perturbations δG and δH are symmetric. If $\|\delta F\|$, $\|\delta G\|$ and $\|\delta H\|$ are sufficiently small, then the perturbed solution $\widetilde{X}$ is well defined [39]. The *condition number of the*

Riccati equation (14) is defined as (see [38])

$$K_R = \lim_{\alpha \to 0} \sup \left\{ \frac{\|\delta X\|}{\alpha \|X\|} : \|\delta F\| \leq \alpha \|F\|,\ \|\delta G\| \leq \alpha \|G\|,\ \|\delta H\| \leq \alpha \|H\| \right\}.$$

For sufficiently small α we have (to first order) $\|\delta X\|/\|X\| \leq K_R \alpha$. Let $\widehat{X}$ be the solution of the Riccati equation computed by a numerical method in finite arithmetic with rounding unit ε. If the method is backward stable, then we can bound the relative error in the solution by

$$\frac{\|\widehat{X} - X\|}{\|X\|} \leq p(n) K_R \varepsilon,$$

where $p(n)$ depends polynomially on n. This bound shows the importance of the condition number in the accuracy estimation of the computed solution.

The determination of the exact condition number K_R is a difficult task. To a first order approximation δX, can be represented as

$$\delta X = -\Omega^{-1}(\delta H) - \Theta(\delta F) + \Pi(\delta G), \tag{16}$$

where $\Omega(Z) := F_c^T Z + Z F_c$, $\Theta(Z) := \Omega^{-1}(Z^T X + XZ)$, and $\Pi(Z) := \Omega^{-1}(XZX)$ are linear operators in the space of $n \times n$ matrices, that determine the sensitivity of X with respect to the perturbations in F, G, H, respectively, and $F_c = F - GX$. Based on (16) it was suggested in [38] to use the approximate condition number

$$K_B := \frac{\|\Omega^{-1}\| \|H\| + \|\Theta\| \|F\| + \|\Pi\| \|G\|}{\|X\|}, \tag{17}$$

where $\|\Omega^{-1}\|$, $\|\Theta\|$, $\|\Pi\|$ are the corresponding induced operator norms. Note that [40]

$$\|\Omega^{-1}\|_F = \frac{1}{\operatorname{sep}(F_c^T, -F_c)},$$

where

$$\operatorname{sep}(F_c^T, -F_c) := \min_{Z \neq 0} \frac{\|F_c^T Z + Z F_c\|_F}{\|Z\|_F}.$$

Figures 4 and 5 show the relative variations $\|\delta X\|_F / \|X\|_F$ in the solutions of well-conditioned and ill-conditioned Riccati equations, respectively, for small relative perturbations in the matrices F and G. While in the case of well-conditioned Riccati equations the change in the solution is of the order of the perturbations in the data, we see that in the case of ill-conditioned Riccati equations the change in the solution is 10000 times larger than the perturbations in the data.

An important practical issue is how to inexpensively estimate the quantities in the condition number (17) and other condition numbers. This estimation is now a routine matter thanks to the development of efficient matrix norm estimators, and in particular the LAPACK norm estimator `xLACON`, [16],[41],[6, Chap. 15], that computes an estimate of the

1-norm $\|B\|_1$ given only the ability to evaluate matrix-vector products Bz and $B^T y$ for judiciously chosen z and y. The use of this estimator for condition estimation in nonsymmetric eigenproblems and matrix Sylvester equations was developed in [42] and [43], respectively. For Riccati equations it is possible to take advantage of the solution symmetry, thus significantly reducing the cost of the estimation.

For the Riccati equation we may use the condition estimator to obtain $\|\Omega^{-1}\|_F$ from the Lyapunov equation $F_c^T Z + ZF_c = C$. An estimate of $\|\Theta\|_1$ can be obtained in a similar manner by solving the Lyapunov equations

$$\begin{aligned} F_c^T Y + Y F_c &= V^T X + XV, \\ F_c Z + Z F_c^T &= V^T X + XV \end{aligned}$$

while $\|\Pi\|_1$ can be estimated by solving the equations

$$\begin{aligned} F_c^T Y + Y F_c &= XVX, \\ F_c Z + Z F_c^T &= XVX. \end{aligned}$$

As in the case of other condition estimators it is always possible to construct special examples where the value produced by `xLACON` underestimates the true value of the corresponding norm by an arbitrary factor. However, in practice severe underestimation happens only in rare circumstances. To demonstrate the performance of these estimators consider the following example.

Example 12 Consider a family of Riccati equations, constructed as $F = TF_0T^{-1}$, $G = T^{-T}G_0T^{-1}$, $H = TH_0T^T$, where $F_0 = \text{diag}(F_1, F_1)$, $G_0 = \text{diag}(G_1, G_1)$, $H_0 = \text{diag}(H_1, H_1)$ are diagonal matrices with $F_1 = \text{diag}(-1 \times 10^{-k}, -2, -3 \times 10^k)$, $H_1 = \text{diag}(3 \times 10^{-k}, 5, 7 \times 10^k)$, $G_1 = \text{diag}(10^{-k}, 1, 10^k)$ and T is a nonsingular transformation matrix. The solution of the Riccati equation is then given by $X = T^{-T}X_1T^{-1}$ where X_1 is a diagonal matrix whose entries are determined simply from the entries of F_1, G_1, H_1. To avoid large rounding errors in constructing and inverting T, this matrix is chosen as $T = T_2ST_1$, where T_1 and T_2 are elementary reflectors and S is the diagonal matrix given by

$$\begin{aligned} T_1 &= I_n - 2[1, 1, ..., 1]^T[1, 1, ..., 1]/n, \\ T_2 &= I_n - 2[1, -1, 1, ..., (-1)^{n-1}]^T[1, -1, 1, ..., (-1)^{n-1}]/n, \\ S &= \text{diag}(1, s, s^2, ..., s^{n-1}), \quad s > 1. \end{aligned}$$

By varying the scalar s it is possible to vary the condition number of T with respect to inversion, since $\text{cond}_2(T) = s^{n-1}$. The solution is obtained with $X_1 = \text{diag}(X_2, X_2)$, and $X_2 = \text{diag}(1, 1, 1)$.

In Figure 6 we show the ratio of the error in the solution to estimate (obtained by `xLACON`) as functions of k and s. We see that, for large k and s corresponding to ill-conditioned equations, the error estimate may become pessimistic. This conservatism is due to the fact that the error estimate is based on an analysis that is pessimistic, and thus a poor estimate of the solution error is usually due not to the estimator but rather to the estimated error bound. At the same time, the numerical experiments show that generally the condition number estimates are always of the same order as the true condition numbers.

As in the pole placement problem, where the choice of poles may represent extra freedom, we can use the freedom in the choice of the weighting matrices Q, S, R to optimize other performance criteria to solve an *optimized linear-quadratic control problem.*

Problem 3 [32] Given matrices $A \in \mathbb{R}^{n,n}$, $B \in \mathbb{R}^{n,m}$ and a set $\mathcal{P} \subset \mathbb{C}$, determine cost matrices Q, S, R such that the closed-loop system obtained via the solution of the associated linear quadratic control problem has eigenvalues that are contained in $\mathcal{P}$, and at the same time some robustness measure is optimized.

In this section we discussed the sensitivity of the linear quadratic optimal control problem and, in particular, the solution approach via the solution of algebraic Riccati equations. In the next section we discuss the optimal H_∞ control problem.

Suboptimal H_∞ control

For the third problem we consider the optimal H_∞ problem. Since, in general, it is difficult to compute the optimal controller, a *modified optimal H_∞ problem* is solved. Instead of looking for the minimum of the norm of the transfer function, one determines the infimum of the parameter γ for which $\|T_{zw}\|_\infty < \gamma$. The optimal H_∞ norm of the transfer function is thus less than or equal to the minimal γ in the modified problem.

The advantage of the modified problem, however, is that it is a one-parameter optimization problem. Furthermore, under some extra assumptions, it is easy to classify when, for a given parameter $\gamma > 0$, a controller exists such that $\|T_{zw}\|_\infty < \gamma$. The computation of such *an admissible controller* is usually called the *suboptimal H_∞ problem.*

Consider the following assumptions:

A1 The pair (A, B_2) is *stabilizable* and the pair (A, C_2) is *detectable*, that is (A^T, C_2^T) is stabilizable.

A2 $D_{22} = 0$ and both D_{12} and D_{21} have full rank.

A3 The matrix $\left[\begin{smallmatrix} A-j\omega I & B_2 \\ C_1 & D_{12} \end{smallmatrix}\right]$ has full column rank for all real ω.

A4 The matrix $\left[\begin{smallmatrix} A-j\omega I & B_1 \\ C_2 & D_{21} \end{smallmatrix}\right]$ has full row rank for all real ω.

One furthermore needs the symmetric matrices

$$\begin{aligned} R_H(\gamma) &:= \begin{bmatrix} D_{11}^T \\ D_{12}^T \end{bmatrix} \begin{bmatrix} D_{11} & D_{12} \end{bmatrix} - \begin{bmatrix} \gamma^2 I_{m_1} & 0 \\ 0 & 0 \end{bmatrix}, \\ R_J(\gamma) &:= \begin{bmatrix} D_{11} \\ D_{21} \end{bmatrix} \begin{bmatrix} D_{11}^T & D_{21}^T \end{bmatrix} - \begin{bmatrix} \gamma^2 I_{p_1} & 0 \\ 0 & 0 \end{bmatrix}. \end{aligned} \tag{18}$$

Let γ_0 be the largest value of γ for which $R_H(\gamma)$ or $R_J(\gamma)$ is singular. Then the solvability of the suboptimal problem is classified by the following theorem.

Theorem 2 [5]. *Consider system (3), with R_H, R_J as in (18). Under assumptions A1–A4, there exists an internally stabilizing controller such that the transfer function from w to z satisfies $\|T_{zw}\|_\infty < \gamma$ if and only if the following four conditions hold.*

1. *$\gamma > \gamma_0$.*
2. *There exists a positive semidefinite solution X_H of the algebraic Riccati equation associated with the Hamiltonian matrix*

$$\begin{aligned} H(\gamma) &= \begin{bmatrix} A_H(\gamma) & G_H(\gamma) \\ H_H(\gamma) & -A_H^T(\gamma) \end{bmatrix} \\ &= \begin{bmatrix} A & 0 \\ -C_1^T C_1 & -A^T \end{bmatrix} - \begin{bmatrix} B_1 & B_2 \\ -C_1^T D_{11} & -C_1^T D_{12} \end{bmatrix} R_H^{-1}(\gamma) \begin{bmatrix} D_{11}^T C_1 & B_1^T \\ D_{12}^T C_1 & B_2^T \end{bmatrix}. \end{aligned}$$

3. *There exists a positive semidefinite solution X_J of the algebraic Riccati equation associated with the Hamiltonian matrix*

$$\begin{aligned} J(\gamma) &= \begin{bmatrix} A_J(\gamma) & G_J(\gamma) \\ H_J(\gamma) & -A_J^T(\gamma) \end{bmatrix} \\ &= \begin{bmatrix} A^T & 0 \\ -B_1 B_1^T & -A \end{bmatrix} - \begin{bmatrix} C_1^T & C_2^T \\ -B_1 D_{11}^T & -B_1 D_{21}^T \end{bmatrix} R_J^{-1}(\gamma) \begin{bmatrix} D_{11} B_1^T & C_1 \\ D_{21}^T B_1^T & C_2 \end{bmatrix}. \end{aligned}$$

4. *$\gamma^2 > \rho(X_H X_J)$, where $\rho(\cdot)$ denotes the spectral radius.*

The optimal solution of the modified H_∞ control problem is obtained by finding the smallest admissible γ such that conditions 1–4 in Theorem 2 hold. This formulation of the problem allows to compute the suboptimal controllers as well.

As before, in order to assess the sensitivity, we must first decide which of the problems (and in which mathematical formulation) we wish to solve. Sensitivity analysis of the optimal H_∞ control problem is still an open problem, and in general it is not clear how to compute this minimum. Also, for the suboptimal H_∞ control problem the sensitivity is not completely understood, although progress has been made in recent years, [44],[45].

We will not repeat the discussion of the previous sections, but it should be clear by now that the sensitivity of different formulations may differ significantly. It is obvious that many factors contribute to the distance of this problem to the nearest problem that does not satisfy assumptions A1–A4, including the distance to the nearest unstabilizable problem. The current situation is even more complicated, since the method involves a nonlinear optimization procedure, and hence the problem of computing the suboptimal controller may be singular or close to singular for different values of γ.

The part of the sensitivity analysis that is most complete [46] is that of the suboptimal H_∞ control problem, where for given matrices A, B_1, B_2, C_1, C_2, D_{11}, D_{12}, D_{21}, $D_{22} = 0$ and for given $\gamma > \gamma_{\text{modopt}}$ the sensitivity of the resulting controller (4) under perturbations δA, $\delta B_1, \ldots, \delta D_{21}, \delta D_{22} = 0$, $\delta\gamma$ in the data is studied. These formulas are not presented here, but it should be obvious that the conditioning of the two Riccati equations for X_H

and X_J, as well as the distance to singularity of the matrices R_H, R_J, plays a major role. One of the major difficulties is the ill-conditioning of one or both Riccati equations near the suboptimal γ. In Figure 7 we show the conditioning of the Riccati equations involving X_H and X_J for a sixth-order system. With γ going to $\gamma_0 = 10.1806399112943$ the sensitivity of the second Riccati equation tends to infinity. As a consequence, most optimization methods will not be able to determine the optimal controller.

There are many more numerical difficulties in the computation of the optimal or suboptimal H_∞ controller. These difficulties and their solution is beyond the scope of this paper and is work in progress, [44].

Example 13 [44] Consider the system

$$\left[\begin{array}{c|c|c} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ \hline C_2 & D_{21} & 0 \end{array}\right] = \left[\begin{array}{cc|cc|c} -1 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 1 \\ \hline 1 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 & \frac{1}{2} & 1 \\ \hline 1 & 1 & 0 & 1 & 0 \end{array}\right].$$

Then (19) becomes

$$R_H(\gamma) = R_J(\gamma) = \begin{bmatrix} \frac{1}{4} - \gamma^2 & 0 & 0 \\ 0 & \frac{1}{4} - \gamma^2 & \frac{1}{2} \\ 0 & \frac{1}{2} & 1 \end{bmatrix}.$$

The positive semidefinite Riccati solution corresponding to $J(\gamma)$ is $X_J = 0$ and the positive semidefinite Riccati solution corresponding to $H(\gamma)$ is

$$X_H = \frac{3}{(1 - \frac{1}{4}\gamma^{-2})4} \times \begin{bmatrix} \frac{1}{2} + \frac{1}{3(1+\sqrt{5})} & \frac{1}{1+\sqrt{5}} - \frac{1}{2} \\ \frac{1}{1+\sqrt{5}} - \frac{1}{2} & \frac{1}{6} - \frac{1}{(1+\sqrt{5})(2+\sqrt{5})} \end{bmatrix}.$$

As γ approaches its minimal value, for the suboptimal H_∞ problem $\gamma_{\text{modopt}} = \frac{1}{2}$, the Riccati solution X_H tends to infinity, R_H and R_J become singular, and the Hamiltonian matrix $H(\gamma)$ becomes ill-defined. The fourth condition in Theorem 2 never fails, because $\rho(X_J X_H) = 0$ for all $\gamma > \gamma_{\text{modopt}}$.

This example demonstrates that the conditioning of the suboptimal H_∞ control problem can deteriorate near the optimum, and clearly in this case an iterative method that approaches γ_{modopt} will have to be terminated before the optimum is reached. Alternative formulations of the modified optimal H_∞ control problem, where these difficulties do not occur, are currently being investigated, [44]. In these formulations Riccati equations as well as the inversion of the matrices R_H, R_J is avoided.

Conclusion and challenges

We have discussed the sensitivity of certain problems of linear control theory, including pole assignment, full state-feedback linear-quadratic, and H_∞ control. We have demonstrated that the mathematical formulation and the splitting of the problem into subproblems are essential factors in the conditioning of these problems. We have shown that standard approaches implemented in numerical toolboxes, which present widely accepted approaches in numerical control, may face problems due to ill-conditioning. Some of these can be avoided by reformulating the problem, but several open problems remain. Another survey paper would be required to discuss all the recent developments in perturbation and error estimates, we refer the reader to the working notes of the SLICOT library [17] (see also `http://www.win.tue.nl/niconet/NIC2/slicot.html`) and the recent monograph [7]. Further analysis and software is needed, in particular, for the following important problems in control theory:

– solution of general quadratic and fractional-affine equations;

– solution of structured eigenvalue problems arising in control;

– computation of the matrices of the optimal and suboptimal controller for some H_∞ control problems;

– computation of the distance to uncontrollability (unobservability);

– computation or bounding of the distance to unstabilizability (undetectability);

– investigation and computation of the sensitivity of general classes of H_∞ control problems.

To assess the accuracy of calculations and to trust numerical results, such condition and accuracy estimates should accompany computational procedures and must be included in the corresponding computer codes. Users must be aware of possible difficulties accompanying the computational process and know how to avoid them. These issues should also become an essential part of the curriculum for scientists and engineers in learning how to use and develop modern computational software.

References

[1] T. Kailath, *Linear Systems.* Prentice-Hall, Englewood Cliffs, NJ, 1980.

[2] V. Mehrmann, *The Autonomous Linear Quadratic Control Problem, Theory and Numerical Solution.* Number 163 in Lecture Notes in Control and Information Sciences. Springer-Verlag, Heidelberg, 1991.

[3] P.Hr. Petkov, N.D. Christov, and M.M. Konstantinov, *Computational Methods for Linear Control Systems.* Prentice-Hall, Hemel Hempstead, 1991.

[4] V. Sima, *Algorithms for Linear Quadratic Optimization.* Marcel Dekker Inc., New York, 1996.

[5] K. Zhou, J.C. Doyle, and K. Glover, *Robust and Optimal Control.* Prentice-Hall, Englewood Cliffs, NJ, 1996.

[6] N.J. Higham, *Accuracy and Stability of Numerical Algorithms.* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, second edition, 2002.

[7] M. Konstantinov, D.-W. Gu, V. Mehrmann, and P. Petkov, *Perturbation Theory for Matrix Equations.* North-Holland, Amsterdam, 2003.

[8] *IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985.* Institute of Electrical and Electronics Engineers, New York, 1985. Reprinted in SIGPLAN Notices, 22(2):9–25, 1987.

[9] J.W. Demmel, "On condition numbers and the distance to the nearest ill-posed problem," *Numer. Math.*, 51:251–289, 1987.

[10] P. Fuhrmann, *A Polynomial Approach to Linear Algebra.* Springer-Verlag, Berlin, 1996.

[11] POLYX, *The Polynomial Toolbox, Version 2.5* Polyx Ltd, Prague, Czech Republic, 2002.

[12] J.H. Wilkinson, *The Algebraic Eigenvalue Problem.* Oxford University Press, Oxford, 1965.

[13] G.H. Golub and C.F. Van Loan, *Matrix Computations.* The Johns Hopkins University Press, Baltimore, third edition, 1996.

[14] The MathWorks, Inc., Cochituate Place, 24 Prime Park Way, Natick, Mass, 01760. *MATLAB Version 6.5.0.180913a (R13)*, 2002.

[15] M. Konstantinov, P. Petkov, and D.W. Gu, "Improved perturbation bounds for general quadratic matrix equations," *Numer. Func. Anal. Optim.*, 20:717–736, 1999.

[16] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, *LAPACK Users' Guide.* SIAM, Philadelphia, PA, third edition, 1999.

[17] P. Benner, V. Mehrmann, V. Sima, S. Van Huffel, and A. Varga. "SLICOT - a subroutine library in systems and control theory," *Appl. Comput. Contr., Sign., Circ.*, 1:499–532, 1999.

[18] J. Kautsky, N.K. Nichols, and P. Van Dooren, "Robust pole assignment in linear state feedback," *Internat. J. Control*, 41:1129–1155, 1985.

[19] G.S. Miminis and C.C. Paige, “A direct algorithm for pole assignment of linear time-invariant multi-input linear systems using state feedback,” *Automatica*, 24:343–356, 1988.

[20] P.Hr. Petkov, N.D. Christov, and M.M. Konstantinov, “A computational algorithm for pole assignment of linear multiinput systems,” *IEEE Trans. Automat. Control*, 31:1044–1047, 1986.

[21] A. Varga. “Robust pole assignment techniques via state feedback,” in *Proc. of IEEE Conference on Decision and Control CDC’2000*, pages 4655–4660, Sydney, Australia, 2000.

[22] C.L. Cox and W.F. Moss, “Backward error analysis for a pole assignment algorithm,” *SIAM J. Matrix Anal. Appl.*, 10:446–456, 1989.

[23] M. Arnold, *Algorithms and conditioning for eigenvalue assignment.* PhD thesis, Northern Illinois University, De Kalb, Illinois, USA, 1993.

[24] M.M. Konstantinov, P.Hr. Petkov, and N.D. Christov, “Sensitivity analysis of the feedback synthesis problem,” *IEEE Trans. Automat. Control*, 42:568–573, 1997.

[25] V. Mehrmann and H. Xu, “An analysis of the pole placement problem. I. The single-input case,” *Electr. Trans. Num. Anal.*, Vol 4:89–105, 1996.

[26] V. Mehrmann and H. Xu, “An analysis of the pole placement problem. II. The multi-input case,” *Electr. Trans. Num. Anal.*, Vol 5:77–97, 1997.

[27] R. Eising, “Between controllable and uncontrollable,” *Systems and Control Letters*, 4:263–264, 1984.

[28] P. Van Dooren, “The generalized eigenstructure problem in linear system theory,” *IEEE Trans. Automat. Control*, AC-26:111–129, 1981.

[29] V. Mehrmann and H. Xu, “Choosing poles so that the single-input pole placement problem is well-conditioned,” *SIAM J. Matrix Anal. Appl.*, 19:664–681, 1998.

[30] N.J. Higham, The Matrix Computation Toolbox. http://www.ma.man.ac.uk/˜higham/mctoolbox.

[31] P.Hr. Petkov, M.M. Konstantinov, D.W. Gu, and I. Postlethwaite, “Optimal eigenstructure assignment of linear systems,” In *Proc. 13 IFAC Congress*, volume C, pages 109–114, San Francisco, USA, 1996.

[32] V. Mehrmann and H. Xu, “Numerical methods in control,” *J. Comput. Appl. Math.*, 123:371–394, 2000.

[33] W.S. Haddad and D. Bernstein, “Controlled design with regional pole constraints,” *IEEE Trans. Automat. Control*, AC-37:54–69, 1992.

[34] P. Van Dooren, "A generalized eigenvalue approach for solving Riccati equations," *SIAM J. Sci. Statist. Comput.*, 2:121–135, 1981.

[35] M. Konstantinov, V. Mehrmann, and P. Petkov, "Perturbation analysis of Hamiltonian Schur and block-Schur forms," *SIAM J. Matrix Anal. Appl.*, 23:387–424, 2001.

[36] G.W. Stewart and J.-G. Sun, *Matrix Perturbation Theory.* Academic Press, New York, 1990.

[37] P. Benner, R. Byers, V. Mehrmann, and H. Xu, "Numerical computation of deflating subspaces of skew Hamiltonian/Hamiltonian pencils," *SIAM J. Matrix Anal. Appl.*, 24:165–190, 2002.

[38] R. Byers, "Numerical condition of the algebraic Riccati equation," *Contemp. Math.*, 47:35–49, 1985.

[39] C. Kenney and G. Hewer, "The sensitivity of the algebraic and differential Riccati equations," *SIAM J. Cont. Optim.*, 28:50–69, 1990.

[40] G. Hewer and C. Kenney, "The sensitivity of the stable Lyapunov equation," *SIAM J. Cont. Optim.*, 26:321–344, 1988.

[41] N.J. Higham, "FORTRAN codes for estimating the one-norm of a real or complex matrix, with applications to condition estimation (Algorithm 674)," *ACM Trans. Math. Software*, 14:381–396, 1988.

[42] Z. Bai, J. Demmel, and A. Mckenney, "On computing condition numbers for the nonsymmetric eigenproblem," *ACM Trans. Math. Software*, 19:202–223, 1993.

[43] N.J. Higham, "Perturbation theory and backward error for $AX - XB = C$," *BIT*, 33:124–136, 1993.

[44] P. Benner, R. Byers, V. Mehrmann, and H. Xu, "Robust methods for robust control," Technical report, Institut für Mathematik, TU Berlin, Str. des 17. Juni 136, D-10623 Berlin, FRG, 2003. in Preparation.

[45] P. Gahinet and A.J. Laub, "Numerically reliable computation of optimal performance in singular H_∞ control," *SIAM J. Cont. Optim.*, 35:1690–1710, 1997.

[46] M.M. Konstantinov, P.H. Petkov, and N.D. Christov, "Conditioning of the continuous-time H_∞ optimisation problem," In *Proc. Third European Control Conference ECC'95*, pages 613–618, Rome, Italy, September 1995.

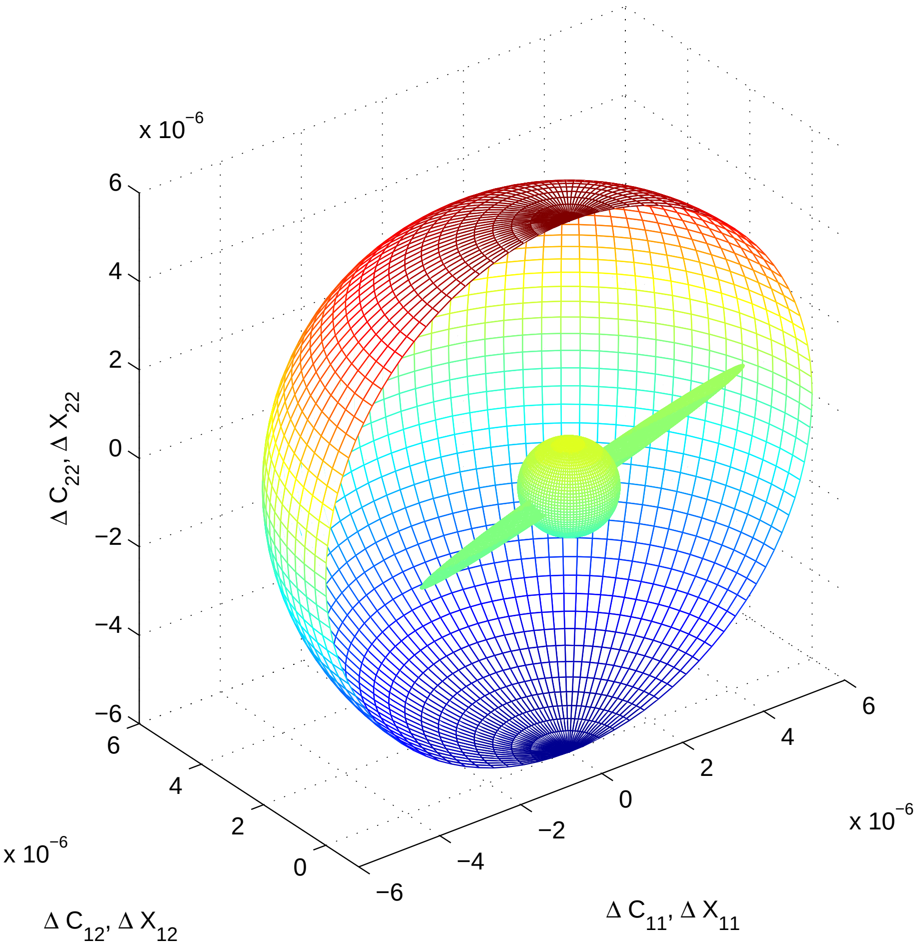

Figure 1: Perturbed Solutions of Lyapunov Equation and a Sensitivity Estimate. The small sphere shows the perturbations of the right hand side, the ellipsoid represents the corresponding variations in the solution, and the large sphere shows the norm-based sensitivity estimate. For some perturbations the sensitivity estimate is very pessimistic.

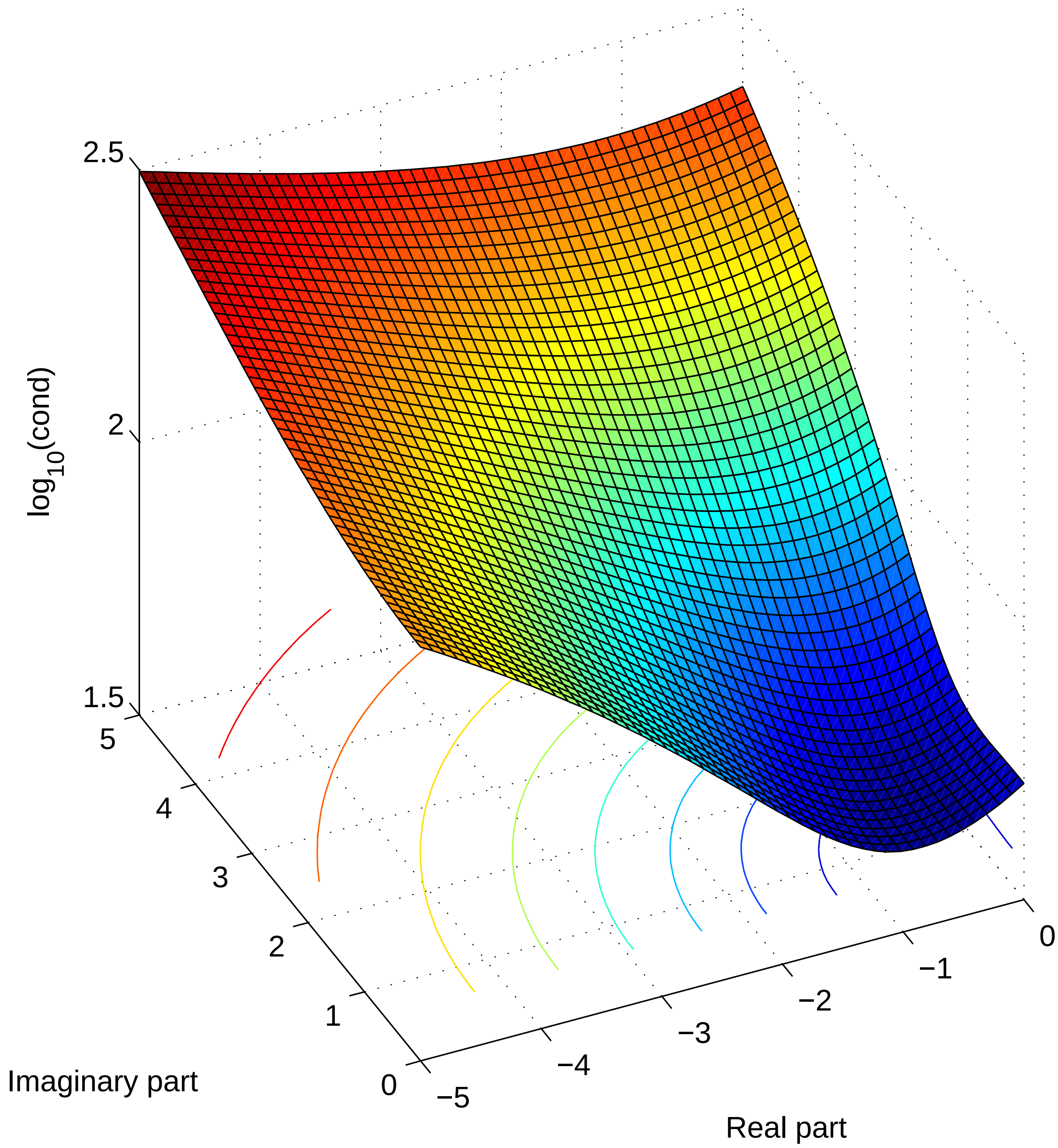

Figure 2: Pole Assignment Conditioning as a Function of Real and Imaginary Parts of λ_1, λ_2. The minimum of the condition number is achieved for λ_1, λ_2 near -1.

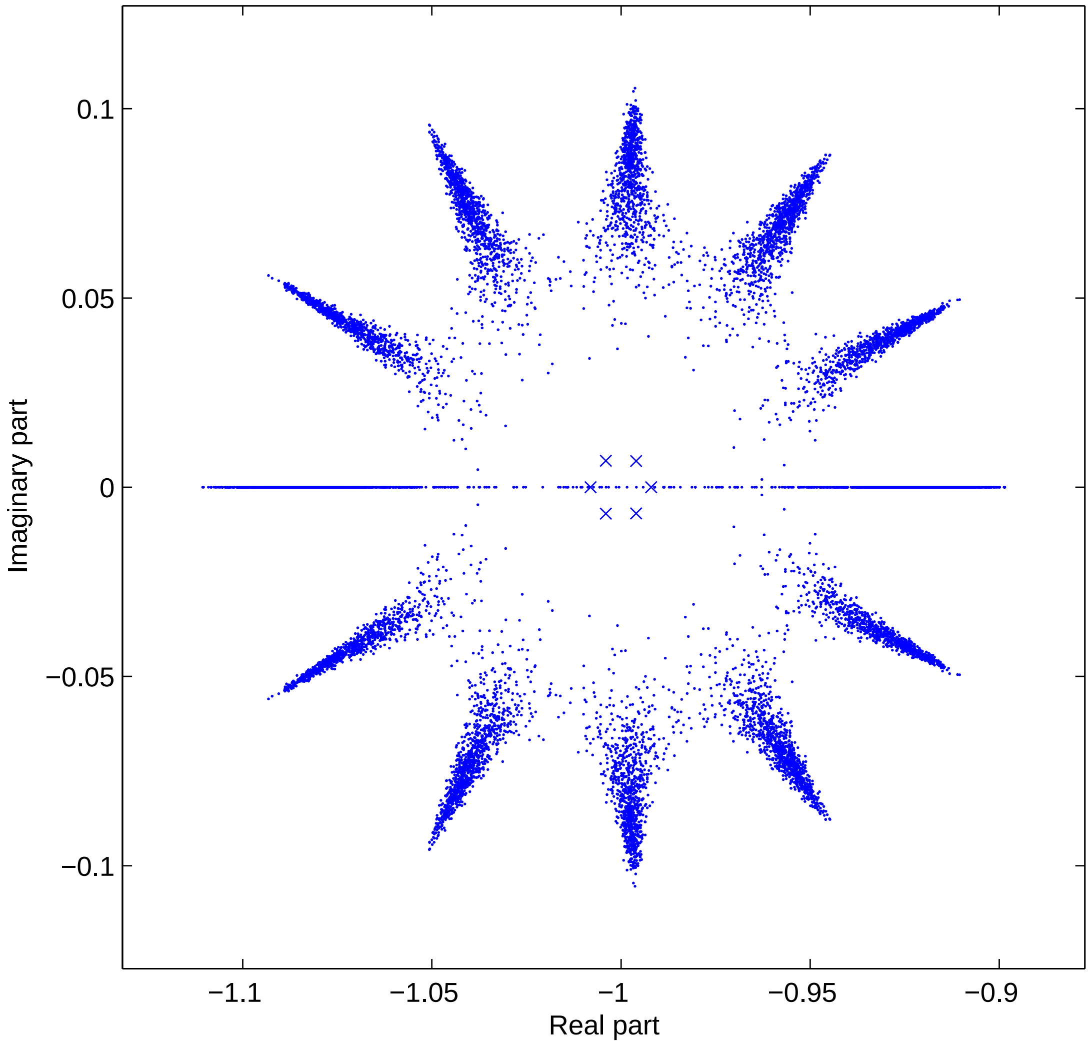

Figure 3: Sensitivity of Closed-Loop Poles. For this example, the poles are sensitive due to their multiplicity and the associated eigenstructure.

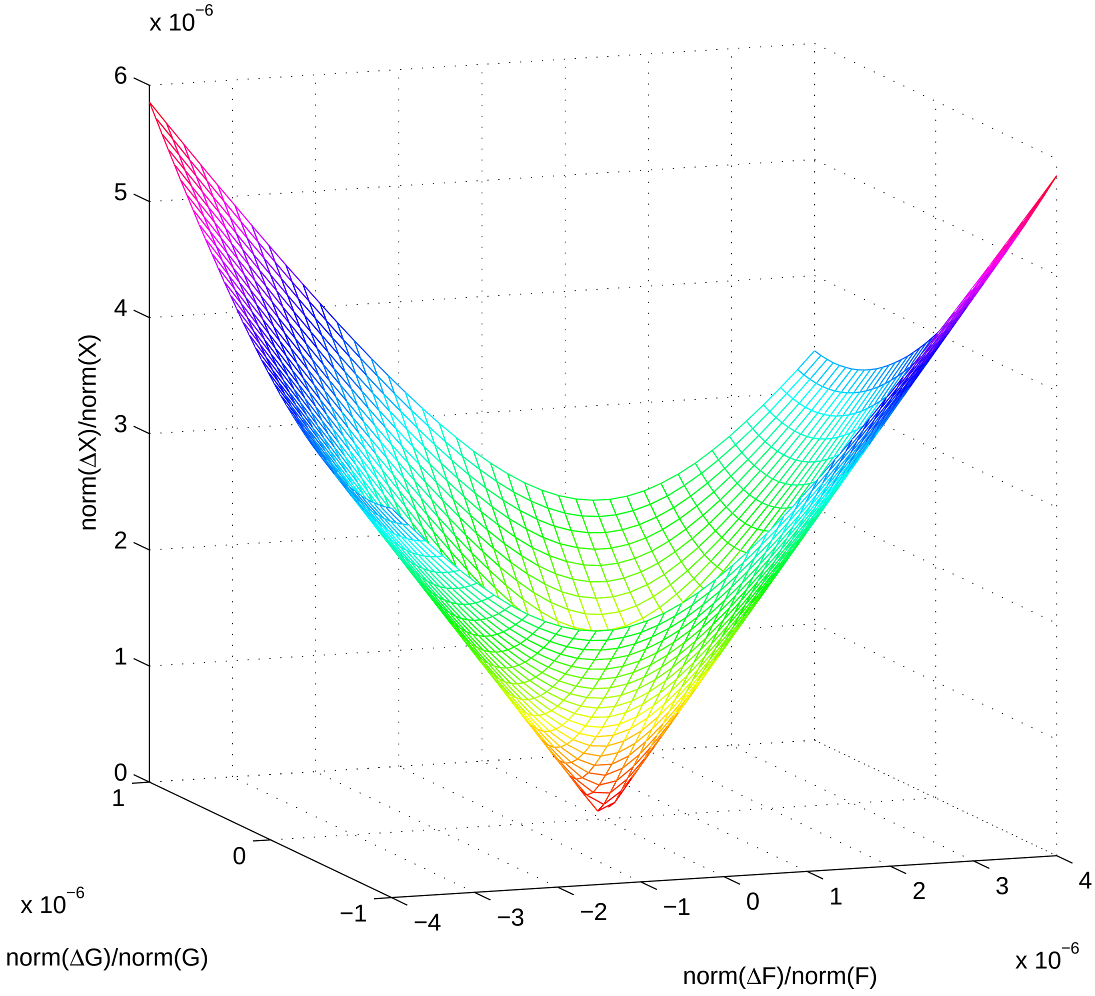

Figure 4: Perturbed Solutions of a Well-Conditioned Riccati Equation. The magnitude of variations in the solution is of the same order as the magnitude of perturbations in the data.

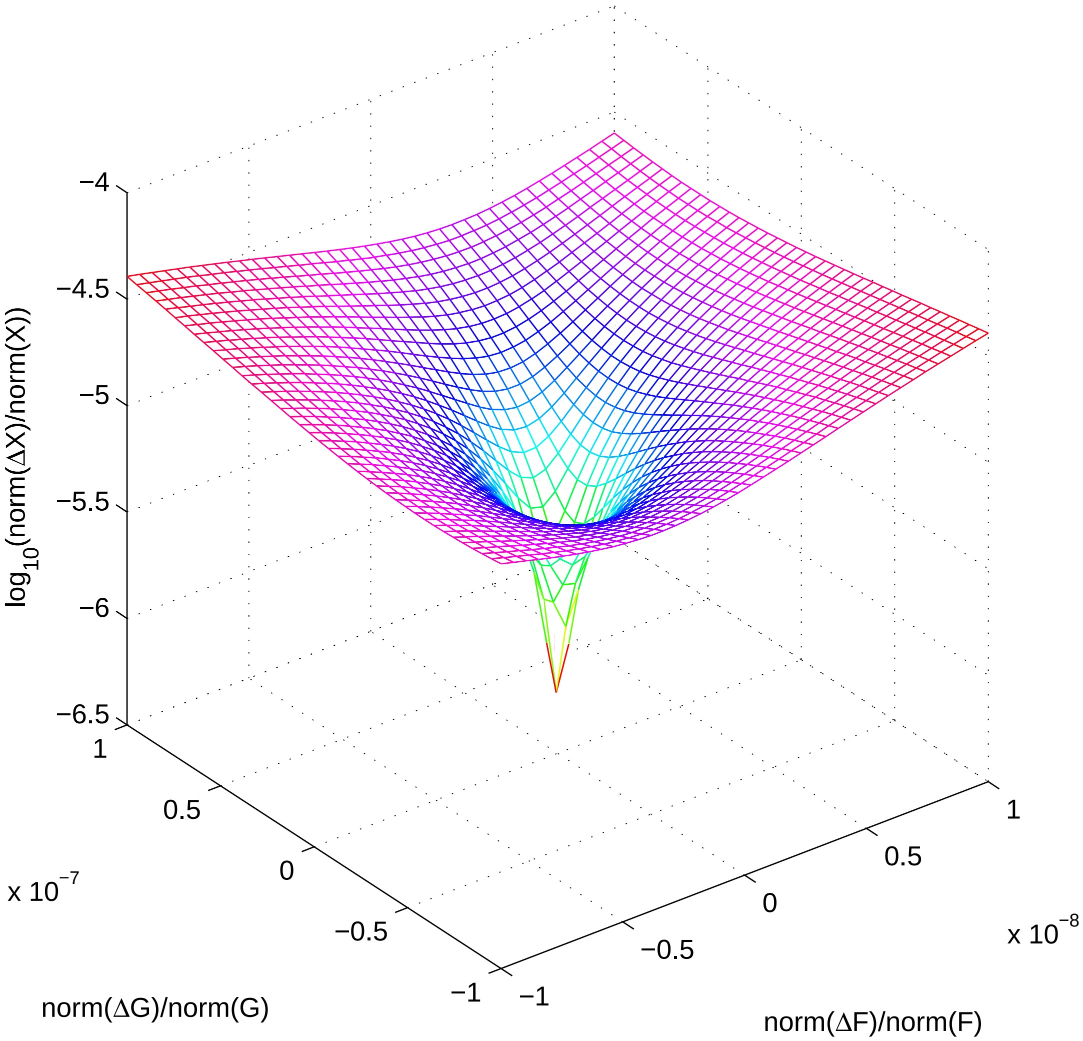

Figure 5: Perturbed Solutions of Ill-Conditioned Riccati Equation. The magnitude of variations in the solution is much larger than the magnitude of perturbations in the data.

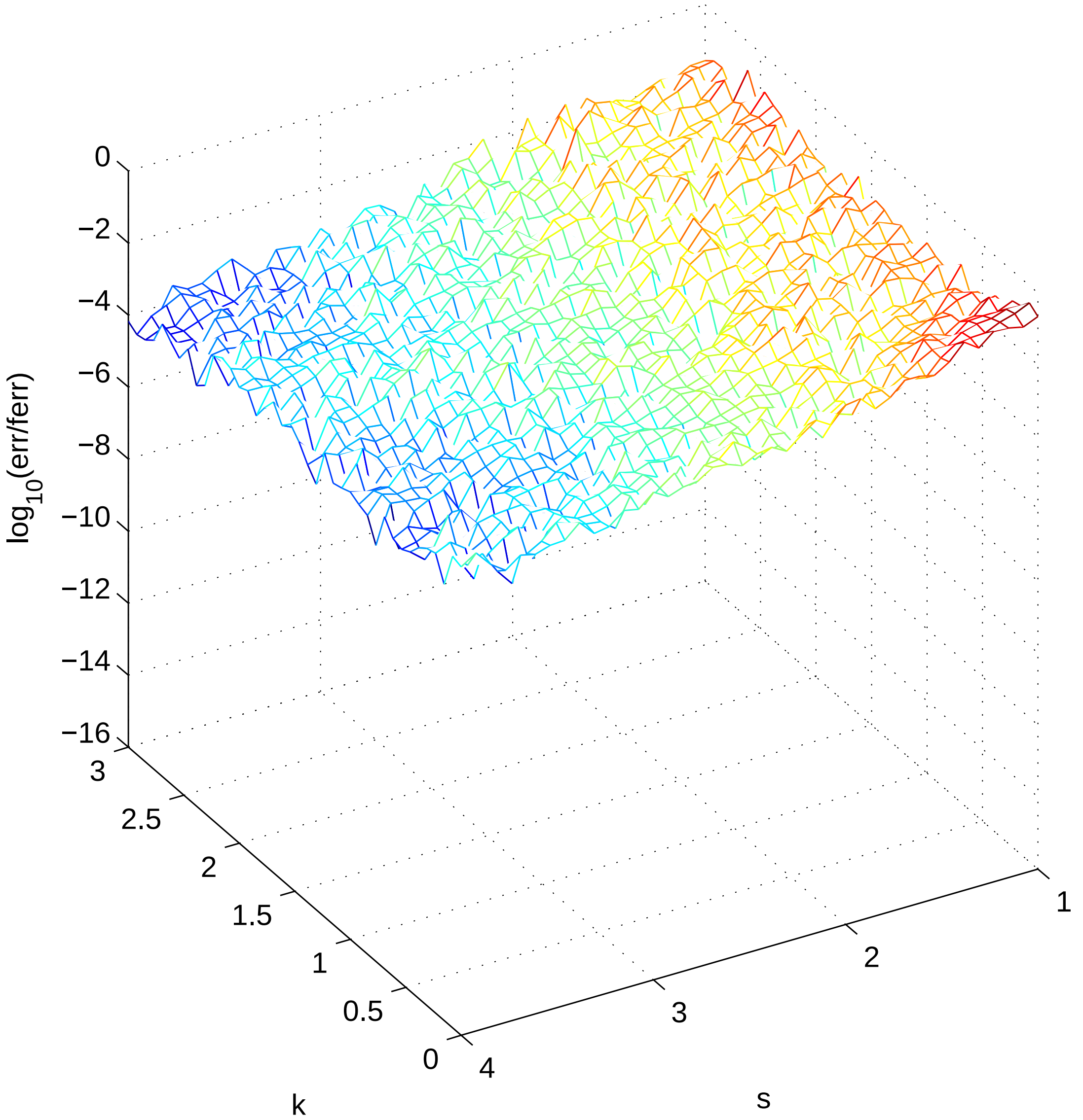

Figure 6: Accuracy of the Error Estimate for a Family of Riccati Equations. The accuracy is reduced for ill-conditioned equations.

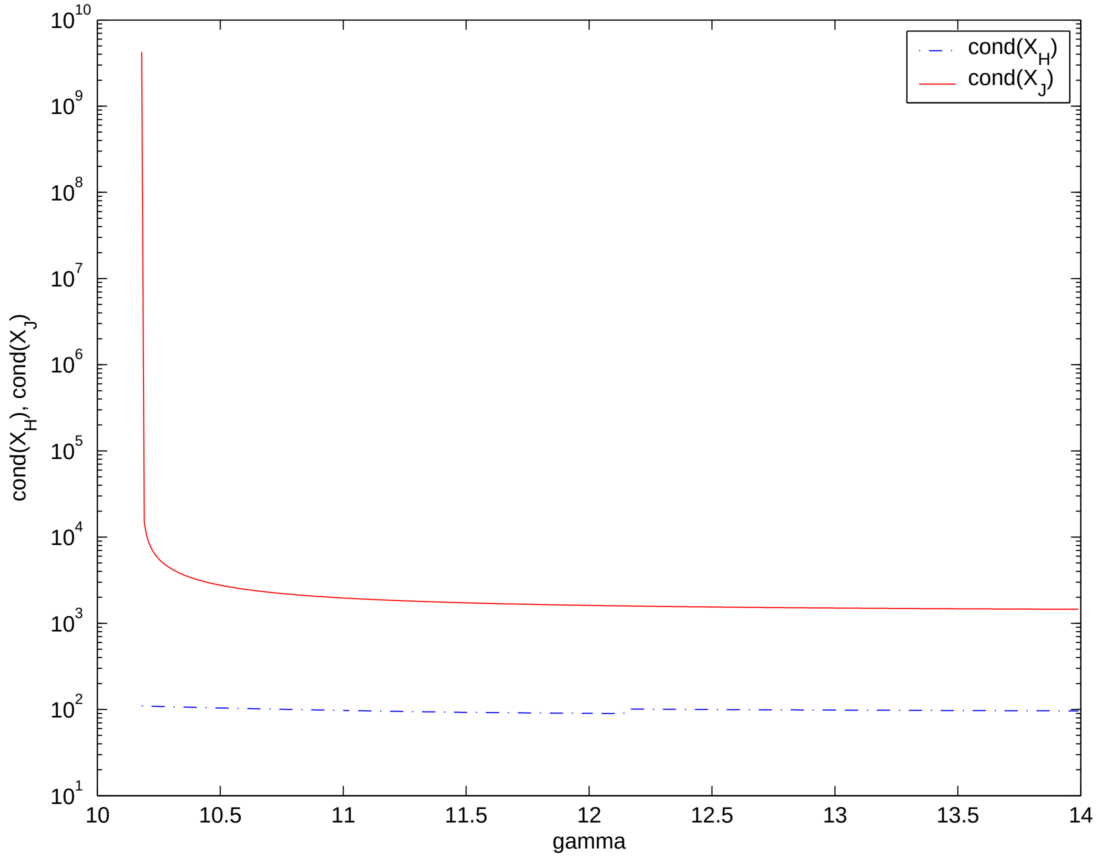

Figure 7: Conditioning of the Solutions of Riccati Equations as a Function of γ. The condition number of the second Riccati equation tends to infinity as γ approaches γ_0.