UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

New Methods for Estimating the Distance of Uncontrollability

Ming Gu

December 1996

CAM Report 96-50

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

New Methods for Estimating the Distance to Uncontrollability

Ming Gu*
Department of Mathematics
University of California
Los Angeles, CA 90095-1555
Tel: (310) 825-4201
Email: mgu@math.ucla.edu

December 3, 1996

Abstract

Controllability is a fundamental concept in control theory. Given a linear control system, we present new algorithms for estimating its distance to uncontrollability, i.e., the norm of the norm-wise smallest perturbation that makes the given system uncontrollable. Many algorithms have been previously proposed to estimate this distance. Our new algorithms are the first that correctly estimate this distance at a cost polynomial in dimension of the given system. We report results from some numerical experiments that demonstrate the reliability and effectiveness of these new algorithms.

*Part of this work was done when the author was with the Department of Mathematics, University of California at Berkeley and Lawrence Berkeley National Laboratory and was partially supported by Applied Mathematical Sciences Subprogram of the Office of Energy Research, U.S. Department of Energy under Contract DE-AC03-76SF00098.

1 Introduction

One of the most fundamental concepts in control theory is that of controllability. A matrix pair $(A, B) \in \mathbf{C}^{n\times n} \times \mathbf{C}^{n\times m}$ is controllable (see Kailath [20, pages 85-90]) if the state function $x = x(t)$ in the linear control system

$$\dot{x} = A\,x + B\,u \tag{1.1}$$

can be directed from any given state to a desired state in finite time by an input $u = u(t)$. Uncontrollability could signal fundamental trouble with the control model or the underlying physical system itself (Byers [11]).

A large number of algebraic and dynamic characterizations of controllability have been given (Laub [21], for example). But each and every one of these has difficulties when implemented in finite precision (Patel, Laub, and Van Dooren [27, page 15]). For instance, it is well known that (A, B) is controllable if and only if

$$\mathbf{rank}\left([A - \lambda I, B]\right) = n\ , \quad \text{for all } \lambda \in \mathcal{C}, \tag{1.2}$$

where $\mathcal{C}$ is the set of complex numbers. However, it is not clear how to numerically verify whether a system is controllable through (1.2). More critically, equation (1.2) does not provide any means to detect systems that are "nearly" uncontrollable, systems that could be equally troublesome. From these considerations, it became apparent (see Laub [21] and Paige [26]) that a more meaningful measure is the distance to uncontrollability, the norm distance of the pair (A, B) from the set of all uncontrollable pairs:

$$\rho(A, B) = \{\|[\Delta A, \Delta B]\|_F : (A + \Delta A, B + \Delta B) \quad \text{uncontrollable.}\} \tag{1.3}$$

It was later shown by Eising [15, 16] that

$$\rho(A, B) = \min_{\lambda\in\mathcal{C}} \sigma_n\left([A - \lambda I, B]\right)\ , \tag{1.4}$$

where $\sigma_n(G)$ denotes the n-th singular value of $G \in \mathbf{C}^{n\times(n+m)}$. Demmel [12] shows that $\rho(A, B)$ is closely related to the sensitivity of the pole-assignment problem.

Many algorithms have been designed to compute $\rho(A, B)$. However, the function to be minimized in (1.4) is not convex and may have as many as n or more local minima. It is not clear just how many local minima there are for any given problem (Byers [11]). Methods that search for a local minimum tend to be efficient but have no guarantee of finding $\rho(A, B)$ with any accuracy, since $\rho(A, B)$ is the global minimum (Boley [4, 6], Boley and Golub [5], Boley and Lu [7], Byers [11], Elsner and He [17], Miminis [24], and Wicks and DeCarlo [31]); and methods that search for the global minimum (Byers [11], Gao and Neumann [18], and He [19]) sometimes do have this guarantee, but require computing time that is inverse proportional to $\rho^2(A, B)$, prohibitively expensive for nearly uncontrollable systems, the kind of systems for which computing $\rho(A, B)$ is important. While the backward stable algorithms of Beelen and Van Dooren [2, 3, 30] and Demmel and Kågström [13, 14] are efficient and very useful for detecting uncontrollability, they often fail to detect near-uncontrollability.

In this paper, we propose new methods to correctly estimate $\rho(A, B)$ to within a factor of 2. They are based on the following bisection method:

Algorithm 1.1 Bisection Method.

```
Set $\delta := \sigma_{\min}([A, B])/2$.
while $\delta \geq \rho(A, B)$
    $\delta := \delta/2$.
endwhile
```

The bisection idea was used to compute the distance of a stable matrix to the unstable matrices (Byers [10]). It was then used to compute the $\mathbf{L}_\infty$ norm of a transfer matrix (Boyd, Balakrishnan and Kabamba [9]); a quadratically convergent version of this later method was developed by Boyd and Balakrishnan [8].

There were past attempts to use Algorithm 1.1 to estimate $\rho(A, B)$ as well [11, 18]; but they have resulted in potentially prohibitively expensive algorithms. The critical difference between our new approach and earlier attempts lies in how to numerically verify whether $\delta \geq \rho(A, B)$. Our new approach is based on a novel verifying scheme (see Section 3.2). Paralleling the development of Boyd and Balakrishnan [8], we have also developed a generally quadratically convergent version of Algorithm 1.1. With very little modification, our new methods can be used to detect the uncontrollable modes for any given tolerance. The knowledge of such modes is essential if one wishes to remove them from the system.

Complexity-wise, these new algorithms differ from previous algorithms in that they are the first algorithms that correctly estimate the distance at a cost polynomial in the matrix size. In fact, they require $O(n^6)$ floating pointing operations. The main cost of these new algorithms is the computation of some eigenvalues of certain sparse generalized eigenvalue problems of size $O(n^2)$.

In §2 we review methods of Byers and Gao and Neumann to minimize the function in (1.4) when λ is restricted to a straight line on the complex plane. In §3 we present our new methods to minimize the function in (1.4) over the entire complex plane. In §4 we present some numerical results. And In §5 we draw conclusions and discuss open questions.

2 Minimization Methods over a Straight Line

Define

$$g(\tau) \equiv \sigma_n \left(\left[A - \left(\lambda_0 + e^{i\theta}\tau \right) I, B \right] \right) , \tag{2.1}$$

where τ is a real variable. To motivate our new methods to minimize the function in (1.4) over the entire complex plane, in this section we present Algorithm 2.1 below to estimate the global minimum of $g(\tau)$ to within a factor of 2 for a given complex number λ_0 and a real angle θ. This algorithm is a variation of the bisection schemes of [11, 18], which can actually compute the global minimum.

Algorithm 2.1 Bisection Method over a Straight Line

```
Set $\delta := g(0)$.
while $\delta \geq g(\tau_*)$
    $\delta := \delta/2$.
endwhile
```

We will also discuss a quadratically convergent version of Algorithm 2.1 in §2.2.

2.1 The Bisection Method over a Straight Line

Let $g(\tau_*) = \min_\tau g(\tau)$. What is missing in Algorithm 2.1 is a scheme to numerically verify whether $\delta \geq g(\tau_*)$ for a given $\delta > 0$. We discuss such a scheme below. Versions of it were developed in Byers [11] and Gao and Neumann [18], and was based on earlier work of Byers [10].

We assume that $\delta \geq g(\tau_*)$. Since $g(\tau)$ is a continuous function with

$$\lim_{\tau \to +\infty} g(\tau + \tau_*) = \infty \quad \text{and} \quad \lim_{\tau \to -\infty} g(\tau + \tau_*) = \infty ,$$

it follows that there exist at least two solutions[1] to the equation $g(\tau) = \delta$. By the definition of singular values, this implies that there exist non-zero vectors $\begin{pmatrix} x \\ y \end{pmatrix}$ and z such that

$$\begin{pmatrix} A - \left(\lambda_0 + e^{i\theta}\tau\right) I & B \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \delta z \quad \text{and} \quad \begin{pmatrix} A^* - \left(\bar{\lambda}_0 + e^{-i\theta}\tau\right) I \\ B^* \end{pmatrix} z = \delta \begin{pmatrix} x \\ y \end{pmatrix} .$$

These equations can be rewritten as

$$\begin{pmatrix} -\delta I & A - \left(\lambda_0 + e^{i\theta}\tau\right) I & B \\ A^* - \left(\bar{\lambda}_0 + e^{-i\theta}\tau\right) I & -\delta I & 0 \\ B^* & 0 & -\delta I \end{pmatrix} \begin{pmatrix} z \\ x \\ y \end{pmatrix} = 0 . \tag{2.2}$$

To simplify (2.2), we QR-factorize

$$\begin{pmatrix} B \\ -\delta I \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \begin{pmatrix} R \\ 0 \end{pmatrix} \tag{2.3}$$

and define

$$\begin{pmatrix} z_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} Q_{11}^* & Q_{21}^* \\ Q_{12}^* & Q_{22}^* \end{pmatrix} \begin{pmatrix} z \\ y \end{pmatrix} .$$

These relations and equation (2.2) imply that $R^* z_1 = 0$. Since $\delta > 0$, R must be non-singular. It follows that $z_1 = 0$. Hence equation (2.2) is reduced to

$$\begin{pmatrix} A - \left(\lambda_0 + e^{i\theta}\tau\right) I & BQ_{22} - \delta Q_{12} \\ -\delta I & \left(A^* - \left(\bar{\lambda}_0 + e^{-i\theta}\tau\right)\right) Q_{12} \end{pmatrix} \begin{pmatrix} x \\ y_1 \end{pmatrix} = 0 ,$$

which can be further rewritten as

$$\begin{pmatrix} A - \lambda_0 I & BQ_{22} - \delta Q_{12} \\ -\delta I & \left(A^* - \bar{\lambda}_0 I\right) Q_{12} \end{pmatrix} \begin{pmatrix} x \\ y_1 \end{pmatrix} = \tau \begin{pmatrix} e^{i\theta} I & 0 \\ 0 & e^{-i\theta} Q_{12} \end{pmatrix} \begin{pmatrix} x \\ y_1 \end{pmatrix} . \tag{2.4}$$

Since Q_{12} is part of the Q factor in the QR factorization (2.3), it follows that

$$Q_{12}^* B - \delta Q_{22}^* = 0 \quad \text{and} \quad Q_{12}^* Q_{12} + Q_{22}^* Q_{22} = I ,$$

[1] If $\delta = g(\tau_*)$, then equation $g(\tau) = \delta$ has a double root at $\tau = \tau_*$.

which imply

$$Q_{12}^* \left(\delta^2 I + BB^*\right) Q_{12} = \delta^2 I \, .$$

Hence Q_{12} is non-singular and the pencil in (2.4) is regular. It is now easy to show that condition $\delta \geq g(\tau_*)$ holds if and only if the matrix pencil in (2.4) has a real eigenvalue τ.

To verify whether $\delta \geq g(\tau_*)$ in Algorithm 2.1, we compute the eigenvalues of the pencil in (2.4). If this pencil has real eigenvalues, then $\delta \geq g(\tau_*)$; otherwise, $\delta < g(\tau_*)$. Since Algorithm 2.1 guarantees that $2\delta \geq g(\tau_*)$ from the previous bisection step, the value of δ after Algorithm 2.1 exits from the **while** loop must satisfy

$$\frac{g(\tau_*)}{2} \leq \delta \leq g(\tau_*) \, . \tag{2.5}$$

We note that equation (2.2) was not reduced in [11], making it more time consuming to verify whether $\delta \geq g(\tau_*)$. It was reduced to a regular eigenvalue problem by solving for y in [18], but the reduction appears to be less numerically reliable than our reduction to (2.4).

2.2 A Quadratically Convergent Variation

Boyd and Balakrishnan [8] note that in the context of computing the $\mathbf{L}_\infty$-norm of a transfer matrix, the function to be minimized is in general approximately quadratic near the maximum, and they used this fact to design a quadratically convergent variation of a bisection method for computing the $\mathbf{L}_\infty$-norm.

Their idea applies equally well in minimizing (2.1). If $\delta - g(\tau_*)$ is small enough and if τ_1 and τ_2 are two roots of $g(\tau) = \delta$ closest to τ_*, then arguments similar to those of [8] show that $(\tau_1 + \tau_2)/2$ is in general a much better approximation to τ_*. We summarize this algorithm below.

Algorithm 2.2 Quadratically Convergent Variation of Algorithm 2.1

```
Set $\delta := g(0)$.
while $\delta \geq g(\tau_*)$
    Choose two real eigenvalues $\tau_1$ and $\tau_2$ of the pencil (2.4).
    $\delta := \min\left\{\delta, g\left(\frac{\tau_1+\tau_2}{2}\right)\right\}/2$.
endwhile
```

With arguments similar to those of [8], it is easy to show that if τ_1 and τ_2 are chosen correctly, then

$$g\left(\frac{\tau_1+\tau_2}{2}\right) - g(\tau_*) = O\left((\delta - g(\tau_*))^2\right) \, . \tag{2.6}$$

This estimate holds even if $g(\tau)$ is *not* approximately quadratic near τ_*. We caution that strictly speaking Algorithm 2.2 is not even asymptotically quadratically convergent, since it terminates as soon as it has found a δ that satisfies (2.5). Nevertheless, relation (2.6) does indicate rapid convergence of Algorithm 2.2 when $g(\tau_*)$ is tiny.

3 Minimization Methods over the Complex Plane

Now we discuss methods to minimize the function in (1.4) over the entire complex plane. Let

$$f(\alpha,\beta) \equiv \sigma_n\left([A-(\alpha+\beta i)\,I, B]\right) \quad \text{and} \quad \rho(A,B) = f\left(\alpha_*,\beta_*\right) = \min_{\alpha,\beta} f(\alpha,\beta)\,. \qquad (3.1)$$

One such method is Algorithm 1.1 discussed in §1. As in Algorithm 2.1, we need to develop a scheme to verify whether $\delta > \rho(A,B)$ in order to complete Algorithm 1.1. To do so, we first prove a fundamental theorem in §3.1; we then provide such a scheme in §3.2; and finally we develop a generally quadratically convergent version of Algorithm 1.1 in §3.3.

3.1 A Fundamental Theorem

Our scheme to verify whether $\delta > \rho(A,B)$ is based on Theorem 3.1 below.

Theorem 3.1 *Assume that $\delta > \rho(A,B)$. Then there are at least two pairs of real numbers α and β such that*

$$\sigma\left([A-(\alpha+\beta i)\,I, B]\right) = \delta \quad \textit{and} \quad \sigma\left([A-(\alpha+\eta+\beta i)\,I, B]\right) = \delta\,, \qquad (3.2)$$

where $0 < \eta \leq 2\,(\delta - \rho(A,B))$; and $\sigma(G)$ denotes a singular value of G.

Proof: From standard perturbation theory we have

$$|(\alpha-\alpha_*)+(\beta-\beta_*)i| - \rho(A,B) \leq f(\alpha,\beta) \leq |(\alpha-\alpha_*)+(\beta-\beta_*)i| + \rho(A,B)\,. \qquad (3.3)$$

Hence $f(\alpha,\beta)$ goes to infinity if $|\alpha+\beta i|$ does. It is well-known that $f(\alpha,\beta)$ is a continuous function of α and β. Consequently the fact that $\delta > \rho(A,B)$ immediately implies that there exists a pair of numbers (α_1,β_1) such that[2] $f(\alpha_1,\beta_1) = \delta$.

From the definition of singular values, α and β satisfy

$$\sigma\left([A-(\alpha+\beta i)\,I, B]\right) = \delta$$

if and only if they satisfy the algebraic equation

$$\det\Big((A-(\alpha+\beta i)\,I)\cdot(A^*-(\alpha-\beta i)\,I) + B\;B^* - \delta^2 I\Big) = 0\,.$$

It follows from $f(\alpha_1,\beta_1) = \delta$ that this algebraic equation has at least one solution; and it follows from (3.3) that all its solutions are finite. Consequently, these solutions form a finite number of closed (continuous) algebraic curves on the α-β plane.

Now we claim that the point (α_*,β_*) must be in the interior of one of these closed curves. In fact, if this is not the case, then there exists a continuous curve $\lambda(\tau) = (\lambda_1(\tau),\lambda_2(\tau))$ on the α-β plane that does not intersect with any of these algebraic curves but "connects" (α_*,β_*) and infinity:

$$\lambda(0) = (\alpha_*,\beta_*) \quad \text{and} \quad \lim_{\tau\to\infty} |\lambda(\tau)| = \infty\,.$$

[2] This fact has been shown in Byers [11] and Gao and Neumann [18].

In other words,

$$f(\lambda_1(0),\lambda_2(0)) = \rho(A,B) \quad \text{and} \quad \lim_{\tau\to\infty} f(\lambda_1(\tau),\lambda_2(\tau)) = \infty \,.$$

It follows from the continuity argument that there exists a τ_1 such that $f(\lambda_1(\tau_1),\lambda_2(\tau_1)) = \delta$. But this contradicts the assumption that the curve $\lambda(\tau)$ does not intersect with any of the algebraic curves. Consequently, the point (α_*,β_*) must be in the interior of one of these closed curves. Among all closed curves that have (α_*,β_*) in their interior, let $\mathcal{G}$ denote the one that covers the smallest area.

It follows from the same continuity argument that there exist two points

$$\mathcal{P}_1 = (\alpha_* - \eta_1, \beta_*) \quad \text{and} \quad \mathcal{P}_2 = (\alpha_* + \eta_2, \beta_*)$$

on $\mathcal{G}$ with $\eta_1 > 0$ and $\eta_2 > 0$. In other words,

$$\sigma([A - (\alpha_* - \eta_1, \beta_* i) I, B]) = \sigma([A - (\alpha_* + \eta_2, \beta_* i) I, B]) = \delta \,, \tag{3.4}$$

For simplicity, we assume that $\mathcal{P}_1$ and $\mathcal{P}_2$ are chosen so that η_1 and η_2 are the smallest positive numbers.

Since the point (α_*,β_*) is in the interior of $\mathcal{G}$ and also lies strictly inside the line segment between $\mathcal{P}_1$ and $\mathcal{P}_2$, it follows that any point that lies strictly inside this line segment is in the interior of curve $\mathcal{G}$. Combining (3.4) with relation (3.3), we get

$$\eta_1 \geq \delta - \rho(A,B) \quad \text{and} \quad \eta_2 \geq \delta - \rho(A,B) \,. \tag{3.5}$$

Now we shift all the points on $\mathcal{G}$ horizontally by the same amount $-\eta$ to get a closed curve

$$\widehat{\mathcal{G}} = \{(\alpha - \eta, \beta)\,|(\alpha,\beta) \text{is a point on } \mathcal{G}.\}$$

Since $\mathcal{P}_2$ is a point on $\mathcal{G}$, $\widehat{\mathcal{P}}_2 = (\alpha_* + \eta_2 - \eta, \beta_*)$ is a point on $\widehat{\mathcal{G}}$. Assume that $\eta < 2(\delta - \rho(A,B))$. Then relation (3.5) implies that $\widehat{\mathcal{P}}_2$ is a point that lies strictly inside the line segment between $\mathcal{P}_1$ and $\mathcal{P}_2$. Hence $\widehat{\mathcal{P}}_2$ is in the interior of curve $\mathcal{G}$. Let $\mathcal{P}_3 = (\alpha_3,\beta_3)$ be the leftmost point on $\mathcal{G}$. Then $\widehat{\mathcal{P}}_3 = (\alpha_3 - \eta, \beta_3)$ is the leftmost point on $\widehat{\mathcal{G}}$. Since $\eta > 0$, we have that $\widehat{\mathcal{P}}_3$ is in the exterior of $\mathcal{G}$.

In other words, we have found a point $\widehat{\mathcal{P}}_2$ that is on $\widehat{\mathcal{G}}$ and in the interior of $\mathcal{G}$, and another point $\widehat{\mathcal{P}}_3$ on $\widehat{\mathcal{G}}$ that is in the exterior of $\mathcal{G}$. Since $\mathcal{G}$ and $\widehat{\mathcal{G}}$ are continuous closed curves, we conclude that these two curves intersect.

Let (α_4,β_4) be any intersecting point. It follows that both (α_4,β_4) and $(\alpha_4 + \eta, \beta_4)$ must be points on $\mathcal{G}$. Hence α_4 and β_4 are a solution to (3.2). Therefore equations (3.2) have at least one solution.

In the following argument we assume that (α_4,β_4) is the only intersecting point of $\mathcal{G}$ and $\widehat{\mathcal{G}}$. Let $\widehat{\mathcal{G}}_1$ denote the set of points on $\widehat{\mathcal{G}}$ that are either on $\mathcal{G}$ or in the interior of $\mathcal{G}$ and let $\mathcal{G}_1$ denote the corresponding set of points on $\mathcal{G}$. If $\widehat{\mathcal{G}}_1$ is not a closed curve itself, then $\widehat{\mathcal{G}}_1$ must be an open curve with one end point on $\mathcal{G}$ and the other in the interior of $\mathcal{G}$. It follows that $\widehat{\mathcal{G}}_1$, and hence $\mathcal{G}_1$, must have positive arclength. Hence the portion of $\mathcal{G}$ without $\mathcal{G}_1$ is a closed curve. But this contradicts the way $\mathcal{G}$ is constructed. This contradiction implies that $\widehat{\mathcal{G}}_1$, and hence $\mathcal{G}_1$, must be closed curves themselves. Let $\widehat{\mathcal{G}}_2$ denote the set of points on $\widehat{\mathcal{G}}$ that are either on $\mathcal{G}$ or in the exterior of $\mathcal{G}$ and let $\mathcal{G}_2$ denote the corresponding set of points on $\mathcal{G}$. A similar argument shows that $\mathcal{G}_2$ must be a closed curve as well.

By construction, $\widehat{\mathcal{G}}_1$ and $\widehat{\mathcal{G}}_2$, and hence $\mathcal{G}_1$ and $\mathcal{G}_2$, do not share any common region with positive area. Hence (α_*, β_*) can only be in the interior of one of these closed curves, this implies that $\mathcal{G}$ is not a closed curve that has (α_*, β_*) in its interior and covers the smallest area, a contradiction to the way $\mathcal{G}$ was constructed. This contradiction is the result of the assumption that $\mathcal{G}$ and $\widehat{\mathcal{G}}$ intersect only once. Hence $\mathcal{G}$ and $\widehat{\mathcal{G}}$ must intersect at least twice, so equations (3.2) must have at least two real solutions. By a continuity argument, equations (3.2) have two, possibly identical, real solutions, even if $\eta = 2\,(\delta - \rho(A, B))$. ♣

3.2 A New Verifying Scheme

Let $\delta > 0$ and $\eta > 0$. In the following we consider how to numerically verify whether equations (3.2) have a real solution. By the definition of singular values, equations (3.2) imply that there exist non-zero vectors $\begin{pmatrix} x \\ y \end{pmatrix}$, z, $\begin{pmatrix} \widehat{x} \\ \widehat{y} \end{pmatrix}$, and $\widehat{z}$ such that

$$\begin{aligned} [A - (\alpha + \beta i)\,I \quad B] \begin{pmatrix} x \\ y \end{pmatrix} &= \delta z\,, \quad \begin{pmatrix} A^* - (\alpha - \beta i)\,I \\ B^* \end{pmatrix} z = \delta \begin{pmatrix} x \\ y \end{pmatrix} \\ [A - (\alpha + \eta + \beta i)\,I \quad B] \begin{pmatrix} \widehat{x} \\ \widehat{y} \end{pmatrix} &= \delta \widehat{z}\,, \quad \begin{pmatrix} A^* - (\alpha + \eta - \beta i)\,I \\ B^* \end{pmatrix} \widehat{z} = \delta \begin{pmatrix} \widehat{x} \\ \widehat{y} \end{pmatrix} . \end{aligned}$$

These equations can be rewritten as

$$\begin{pmatrix} -\delta I & A - \alpha I & B \\ A^* - \alpha I & -\delta I & 0 \\ B^* & 0 & -\delta I \end{pmatrix} \begin{pmatrix} z \\ x \\ y \end{pmatrix} = \beta i \begin{pmatrix} 0 & I & 0 \\ -I & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z \\ x \\ y \end{pmatrix} \tag{3.6}$$

and

$$\begin{pmatrix} -\delta I & A - (\alpha + \eta) I & B \\ A^* - (\alpha + \eta) I & -\delta I & 0 \\ B^* & 0 & -\delta I \end{pmatrix} \begin{pmatrix} \widehat{z} \\ \widehat{x} \\ \widehat{y} \end{pmatrix} = \beta i \begin{pmatrix} 0 & I & 0 \\ -I & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \widehat{z} \\ \widehat{x} \\ \widehat{y} \end{pmatrix} . \tag{3.7}$$

In the QR factorization (2.3), define

$$\begin{pmatrix} z_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} Q_{11}^* & Q_{21}^* \\ Q_{12}^* & Q_{22}^* \end{pmatrix} \begin{pmatrix} z \\ y \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} \widehat{z}_1 \\ \widehat{y}_1 \end{pmatrix} = \begin{pmatrix} Q_{11}^* & Q_{21}^* \\ Q_{12}^* & Q_{22}^* \end{pmatrix} \begin{pmatrix} \widehat{z} \\ \widehat{y} \end{pmatrix} .$$

These relations and equations (3.6) and (3.7) imply that $R^* z_1 = 0$ and $R^* \widehat{z}_1 = 0$. Since R is non-singular, it follows that $z_1 = 0$ and $\widehat{z}_1 = 0$. Hence equations (3.6) and (3.7) are reduced to

$$\begin{pmatrix} A - \alpha I & B Q_{22} - \delta Q_{12} \\ -\delta I & (A^* - \alpha I)\, Q_{12} \end{pmatrix} \begin{pmatrix} x \\ y_1 \end{pmatrix} = \beta i \begin{pmatrix} I & 0 \\ 0 & -Q_{12} \end{pmatrix} \begin{pmatrix} x \\ y_1 \end{pmatrix} \tag{3.8}$$

and

$$\begin{pmatrix} A - (\alpha + \eta) I & B Q_{22} - \delta Q_{12} \\ -\delta I & (A^* - (\alpha + \eta) I)\, Q_{12} \end{pmatrix} \begin{pmatrix} \widehat{x} \\ \widehat{y}_1 \end{pmatrix} = \beta i \begin{pmatrix} I & 0 \\ 0 & -Q_{12} \end{pmatrix} \begin{pmatrix} \widehat{x} \\ \widehat{y}_1 \end{pmatrix} . \tag{3.9}$$

As shown in §2.1, Q_{12} is always non-singular for $\delta > 0$. Hence the matrix on the right hand sides of both (3.8) and (3.9) is non-singular. In order for the two pencils defined in (3.8) and (3.9) to share a common pure imaginary eigenvalue βi, the following matrix equation for $X \in \mathbf{R}^{2n\times 2n}$

$$\begin{aligned} & \begin{pmatrix} A-\alpha I & BQ_{22}-\delta Q_{12} \\ -\delta I & (A^*-\alpha I)Q_{12} \end{pmatrix} X \begin{pmatrix} I & 0 \\ 0 & -Q_{12} \end{pmatrix}^* \\ - & \begin{pmatrix} I & 0 \\ 0 & -Q_{12} \end{pmatrix} X \begin{pmatrix} A-(\alpha+\eta)I & BQ_{22}-\delta Q_{12} \\ -\delta I & (A^*-(\alpha+\eta)I)Q_{12} \end{pmatrix}^* = 0 \end{aligned} \qquad (3.10)$$

must have a non-zero solution. Partition $X = \begin{pmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{pmatrix}$, this matrix equation becomes

$$\mathcal{H}u = 0 \quad \text{and} \quad \mathcal{A}u = 2\alpha\mathcal{B}u\ , \qquad (3.11)$$

with $\quad \mathcal{B} = \begin{pmatrix} 0 & 0 & -Q_{12}\otimes I & 0 \\ 0 & 0 & 0 & I\otimes Q_{12} \end{pmatrix}, \quad \hat{B} = BQ_{22}-\delta Q_{12}\ , \quad u = \begin{pmatrix} \mathbf{vec}\,(X_{11}) \\ \mathbf{vec}\,(X_{22}) \\ \mathbf{vec}\,(X_{12}) \\ \mathbf{vec}\,(X_{21}) \end{pmatrix};$

$$\begin{aligned} \mathcal{A} &= \begin{pmatrix} \delta I & Q_{12}\otimes\hat{B} & Q_{12}\otimes A-((A-\eta)Q_{12})\otimes I & 0 \\ -\delta I & \hat{B}^*\otimes Q_{12} & 0 & (A-\eta)\otimes Q_{12}-I\otimes(A^*Q_{12}) \end{pmatrix}; \\ \mathcal{H} &= \begin{pmatrix} I\otimes A-(A-\eta I)\otimes I & 0 & -\hat{B}^*\otimes I & I\otimes\hat{B} \\ 0 & ((A-\eta)Q_{12})\otimes Q_{12}-Q_{12}\otimes(A^*Q_{12}) & \delta Q_{12}\otimes I & -\delta I\otimes Q_{12} \end{pmatrix}. \end{aligned}$$

In these equations, $\otimes$ is the Kronecker product and $\mathbf{vec}(G)$ is a vector formed by stacking the column vectors of G.

To reduce (3.11) to a standard generalized eigenvalue problem, let

$$\mathcal{H} = (\mathcal{R}\quad 0)\begin{pmatrix} \mathcal{Q}_{11} & \mathcal{Q}_{12} \\ \mathcal{Q}_{21} & \mathcal{Q}_{22} \end{pmatrix} \qquad (3.12)$$

be the RQ factorization of $\mathcal{H}$, where $\mathcal{Q}_{ij} \in \mathbf{C}^{2n^2\times 2n^2}$; and define

$$\begin{pmatrix} w \\ v \end{pmatrix} = \begin{pmatrix} \mathcal{Q}_{11} & \mathcal{Q}_{12} \\ \mathcal{Q}_{21} & \mathcal{Q}_{22} \end{pmatrix} u\ .$$

Then the first equation in (3.11) reduces to $\mathcal{R}w = 0$. By setting $w = 0$, the second equation in (3.11) becomes

$$\mathcal{A}_1 v = \alpha\mathcal{B}_1 v\ , \qquad (3.13)$$

where

$$\mathcal{A}_1 = \mathcal{A}\begin{pmatrix} \mathcal{Q}_{21}^* \\ \mathcal{Q}_{22}^* \end{pmatrix} \quad \text{and} \quad \mathcal{B}_1 = \begin{pmatrix} -Q_{12}\otimes I & 0 \\ 0 & I\otimes Q_{12} \end{pmatrix}\mathcal{Q}_{22}^*\ .$$

Equation (3.13) is now a $2n^2$-by-$2n^2$ generalized eigenvalue problem. Hence we have reduced the problem of finding a non-zero solution to (3.10) to the generalized eigenvalue problem (3.13).

To summarize, we have shown that in order for (3.2) to have at least one real solution (α, β), both matrix pencils in (3.8) and (3.9) must share a common pure imaginary eigenvalue βi. This requires that the matrix equation (3.10) must have a non-zero solution, which, in turn, is equivalent to requiring that the generalized eigenvalue problem (3.13) have a real eigenvalue α.

In order to verify whether $\delta > \rho(A,B)$ in any bisection step of Algorithm 1.1, we set $\eta = \delta$ in (3.2) and check whether the generalized eigenvalue problem (3.13) has any real eigenvalues α. If it does, we then check for each real α whether the two matrix pencils in (3.8) and (3.9) share a common pure imaginary eigenvalue βi. If they do for at least one α, then we have found a pair of α and β such that

$$\sigma\left([A-(\alpha,\beta i)\,I,B]\right)=\delta \quad \text{and hence} \quad \delta \geq \rho(A,B)\,.$$

On the other hand, if (3.13) does not have a real eigenvalue, or if the matrix pencils in (3.8) and (3.9) do not share a common pure imaginary eigenvalue for any real eigenvalue of (3.13), then we conclude by Theorem 3.1 that

$$\delta = \eta > 2\left(\delta - \rho(A,B)\right)\,.$$

On the other hand, Algorithm 1.1 guarantees that $2\delta \geq \rho(A,B)$ from the previous bisection step. Thus the value of δ after Algorithm 1.1 exits from the **while** loop must satisfy

$$\frac{\rho(A,B)}{2} \leq \delta \leq 2\,\rho(A,B)\,. \tag{3.14}$$

3.3 A Generally Quadratically Convergent Variation

Algorithm 1.1 converges linearly. Following Boyd and Balakrishnan [8] (see §2.2), we develop a generally quadratically convergent version of Algorithm 1.1 in this section. In the following development, we assume that $f(\alpha,\beta)$ is analytic in both α and β in a small neighborhood of (α_*,β_*) so that $f(\alpha,\beta)$ permits the following expansions

$$\begin{aligned} f(\alpha+\eta,\beta) &= f(\alpha,\beta)+\eta\frac{\partial}{\partial\alpha}f(\alpha+\frac{\eta}{2},\beta)+O(\eta^3) && (3.15)\\ f(\alpha,\beta) &= f(\alpha_*,\beta_*)+\gamma\,(\alpha-\alpha_*)^2+2\nu\,(\alpha-\alpha_*)\,(\beta-\beta_*)+\xi\,(\beta-\beta_*)^2 \\ &\quad +O\left((|\alpha-\alpha_*|+|\beta-\beta_*|)^3\right)\,, && (3.16)\end{aligned}$$

where γ, ν, and ξ are the relevent second-order partial derivatives at (α_*,β_*). We further assume that the matrix $\Gamma \equiv \begin{pmatrix} \gamma & \nu \\ \nu & \xi \end{pmatrix}$ is positive definite. These expansions with a positive definite Γ imply that (α_*,β_*) is at least a local minimum.

Now assume that α and β are such that $f(\alpha+\eta,\beta)=f(\alpha,\beta)$. It follows from (3.15) that

$$\frac{\partial}{\partial\alpha}f(\alpha+\frac{\eta}{2},\beta)+O(\eta^2)=0\,.$$

Expanding the partial derivative at (α_*,β_*) to get

$$\gamma\left(\alpha+\frac{\eta}{2}-\alpha_*\right)+\nu\,(\beta-\beta_*)=O\left(\eta^2+(\alpha-\alpha_*)^2+(\beta-\beta_*)^2\right)\,, \tag{3.17}$$

where we have used the fact that $\frac{\partial}{\partial\alpha}f(\alpha_*,\beta_*)=0$.

Let $\mu = \delta - f(\alpha_*, \beta_*) = \delta - \rho(A, B)$; and let (α_1, β_1) and (α_2, β_2) be the two solutions to (3.2) that are near (α_*, β_*) (see Theorem 3.1). It follows that α_i and β_i satisfy both (3.17) and (3.16) for $i = 1$ and 2. Now let $\zeta_{1,i}$ and $\zeta_{2,i}$ denote the error terms in (3.17) and (3.16) for (α_i, β_i). Consequently,

$$\begin{aligned} \gamma\left(\alpha_i + \frac{\eta}{2} - \alpha_*\right) + \nu(\beta_i - \beta_*) &= \zeta_{1,i} \\ \gamma(\alpha_i - \alpha_*)^2 + 2\nu(\alpha_i - \alpha_*)(\beta_i - \beta_*) + \xi(\beta_i - \beta_*)^2 &= \mu + \zeta_{2,i} \,. \end{aligned}$$

It follows from the first equation that

$$(\alpha_i - \alpha_*) = -\frac{\eta}{2} - \frac{\nu}{\gamma}(\beta_i - \beta_*) + \frac{\zeta_{1,i}}{\gamma} \,. \tag{3.18}$$

Plugging this into the second equation and simplifying:

$$(\beta_i - \beta_*)^2 = \frac{\mu - \gamma\eta^2/4}{\xi - \dfrac{\nu^2}{\gamma}} + \zeta_{3,i} \,,$$

where

$$\zeta_{3,i} = \frac{\zeta_{2,i} + 2\zeta_{1,i}\left(\dfrac{\eta}{2} + \dfrac{\nu}{\gamma}(\beta_i - \beta_*)\right) - \dfrac{\zeta_{1,i}^2}{\gamma} - 2\zeta_{1,i}(\beta_i - \beta_*)\dfrac{\nu}{\gamma}}{\xi - \dfrac{\nu^2}{\gamma}} = O\left((\eta + |\alpha_i - \alpha_*| + |\beta_i - \beta_*|)^3\right) .$$

On the other hand, since Γ is assumed to be positive definite, equation (3.16) implies that $|\alpha_i - \alpha_*|^2 + |\beta_i - \beta_*|^2 = \mathcal{O}(\mu)$. Furthermore, the choice of $\eta = \delta$ in Algorithm 1.1 ensures that $\eta = \mathcal{O}(\mu)$ (see §3.2). Hence,

$$\begin{aligned} (\beta_i - \beta_*) &= \pm\sqrt{\frac{\mu - \gamma\eta^2/4}{\xi - \dfrac{\nu^2}{\gamma}} + \zeta_{3,i}} = \pm\sqrt{\frac{\mu - \gamma\eta^2/4}{\xi - \dfrac{\nu^2}{\gamma}}} + O\left(\frac{\zeta_{3,i}}{\sqrt{\mu}}\right) \\ &= \pm\sqrt{\frac{\mu - \gamma\eta^2/4}{\xi - \dfrac{\nu^2}{\gamma}}} + O(\mu) \,. \end{aligned}$$

The result can be rewritten as[3]

$$(\beta_1 - \beta_*) = \sqrt{\frac{\mu - \gamma\eta^2/4}{\xi - \frac{\nu^2}{\gamma}}} + O(\mu) \quad \text{and} \quad (\beta_2 - \beta_*) = -\sqrt{\frac{\mu - \gamma\eta^2/4}{\xi - \frac{\nu^2}{\gamma}}} + O(\mu) \,.$$

Combining these equations,

$$\beta_* = \frac{\beta_1 + \beta_2}{2} + O(\mu) \,.$$

[3] These relations hold as long as $\gamma\eta^2/4 \ll \mu$. It is likely that under certain conditions, Theorem 3.1 holds for much larger values of η as well.

Plugging this relation into (3.18) and combining the equations for $i = 1, 2$, we obtain

$$\alpha_* = \frac{\alpha_1 + \alpha_2 + \eta}{2} + O(\mu) \, .$$

Define

$$\alpha_{\mathbf{new}} = \frac{\alpha_1 + \alpha_2 + \eta}{2} \quad \text{and} \quad \beta_{\mathbf{new}} = \frac{\beta_1 + \beta_2}{2} \, .$$

It follows that

$$\alpha_* = \alpha_{\mathbf{new}} + O(\mu) \quad \text{and} \quad \beta_* = \beta_{\mathbf{new}} + O(\mu) \, ,$$

and that

$$f(\alpha_{\mathbf{new}}, \beta_{\mathbf{new}}) - f(\alpha_*, \beta_*) = \mathcal{O}\left(|\alpha_{\mathbf{new}} - \alpha_*|^2 + |\beta_{\mathbf{new}} - \beta_*|^2\right) = \left(\mathcal{O}(\mu)\right)^2 \, . \tag{3.19}$$

We note that this relation is very similar to (2.6). Now we modify Algorithm 1.1 to get

Algorithm 3.1 Quadratically Convergent Variation of Algorithm 2.1

```
  Set $\delta := g(0)$.
  while $\delta \geq \rho(A, B)$
      Choose two real solutions $(\alpha_1, \beta_1)$ and $(\alpha_2, \beta_2)$ of (3.2).
      $\delta := \min\left\{\delta, f\left(\frac{\alpha_1 + \alpha_2 + \eta}{2}, \frac{\beta_1 + \beta_2}{2}\right)\right\} /2.$
  endwhile
```

In our implementation, we computed $f\left(\frac{\alpha_1 + \alpha_2 + \eta}{2}, \frac{\beta_1 + \beta_2}{2}\right)$ among adjacent pairs of real solutions and chose the pair with smallest f value. We note that in both Algorithms 1.1 and 3.1, we can compute a better initial guess δ by using Algorithm 2.2 with some values of λ_0 and θ, such as $\lambda_0 = 0$ and $\theta = 0$.

Algorithm 3.1 was derived under the assumptions at the beginning of §3.3, which need not hold for all linear control systems of the form (1.1). Hence estimate (3.19) may not hold for some linear control systems. However, it is clear that Algorithm 3.1 converges at least linearly to arrive at an estimate δ that satisfies (3.14). Similar to Algorithm 2.2, Algorithm 3.1 is not strictly speaking quadratically convergent since it terminates as soon as it has found a δ that satisfies (3.14). Nevertheless, Algorithm 3.1 does converge much more rapidly than Algorithm 1.1 when $\rho(A, B)$ is tiny. We discuss this point further in §4.

3.4 Further Considerations

Sometimes it may be more important to find the uncontrollable modes of (1.1) for a given tolerance ε. In this case, we solve equations (3.2) with $\delta = \eta = \varepsilon$. If there are no solutions to (3.2), then the system (1.1) is controllable; otherwise, each solution to (3.2) corresponds to an uncontrollable mode. Conversely, it is easy to see from the proof of Theorem 3.1 that any uncontrollable mode will result in at least two solutions to (3.2). Hence the set of all solutions to (3.2) provide approximations to the uncontrollable modes of (3.2). The formulas for $\alpha_{\mathbf{new}}$ and $\beta_{\mathbf{new}}$ provide more accurate approximations to these modes.

If $\rho(A, B)$ is very small, then $\delta = \eta$ will also become very small during the execution of Algorithms 1.1 and 3.1. In fact, for small enough η, the two *different points* $\alpha + \beta i$ and $\alpha + \eta + \beta i$ will look identical. Hence the solutions to (3.2) are potentially ill-conditioned. See §4 for more details.

Like many other algorithms in engineering computations, such as those for semi-definite programming [1, 25, 29], both Algorithms 1.1 and 3.1 are expensive for large problems, since both the reduction to and the solution of the pencil (3.13) require $O(n^6)$ floating point operations. However, the eigenvalue problem (3.11) is highly sparse as a $4n^2 \times 4n^2$ problem. It is likely that sparse matrix computation technologies, such as the implicitly restarted Arnoldi iteration [22, 23, 28], can be used to compute the real eigenvalues of (3.13) quickly. The effectiveness of this approach is currently under thorough investigation.

4 Numerical Experiments

We have done some elementary numerical experiments with Algorithms 1.1 and 3.1. In this section we report some of the results obtained from these experiments. The experiments were done in `matlab` in double precision.

Matrices in Examples 2 through 5 were taken from Gao and Neumann [18]. These are systems with small $\rho(A, B)$. Global optimization methods (Byers [11], Gao and Neumann [18], and He [19]) sometimes require prohibitively expensive computation time to correctly estimate $\rho(A, B)$ in these cases. On the other hand, both Algorithms 1.1 and 3.1 worked well on them, with Algorithm 3.1 converging much faster than Algorithm 1.1 as expected.

Example 1. In this example we took $A \in \mathbf{R}^{5\times 5}$ and $B \in \mathbf{R}^5$ to be random matrices. This is a matrix pair with fairly large $\rho(A, B)$. Both Algorithms 1.1 and 3.1 took 2 iterations to terminate. This example illustrates that for linear systems (1.1) that are far away from the set of uncontrollable systems, both Algorithms 1.1 and 3.1 take very few iterations.

Example 2. In this example we took

$$A = \begin{pmatrix} 1 & 1 & 2 & 3 \\ -1 & 1 & 4 & 5 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & -2 & 1 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

This pair is uncontrollable since the smallest singular value of $[A - (1 \pm 2i)I \;\; B]$ is zero. Algorithms 1.1 and 3.1 took 42 and 5 iterations, respectively, to find $\rho(A, B) \leq 10^{-15}$.

Example 3. In this example we took

$$A = \begin{pmatrix} -0.32616458 & -0.09430266 & 0.05207847 & -.08481401 & 0.05829280 \\ 0.01158922 & -.39787419 & -.14901699 & -.01394125 & -.10626942 \\ 0.05623810 & -.03153954 & -.50160557 & -.05748511 & -.00552321 \\ 0.07474298 & .01205259 & -.00249719 & -.26871590 & .23107147 \\ .07829650 & -.19442017 & -.05631780 & .17761876 & -.45847313 \end{pmatrix}.$$

and $B = (-.13705019, -.18309206, -.13735654, .80258791, .53355446)^T$. Algorithm 1.1 took 20 iterations to return $\delta = 4.49639424 \times 10^{-7}$. For $\delta = 8.99278848 \times 10^{-7}$, it returned six distinct solutions to (3.2): $(\alpha, \beta) = \left(-6.5094391 \times 10^{-1}, \pm 5.02193288 \times 10^{-7}\right)$ and

$$\left(-2.60379928 \times 10^{-1}, \pm 5.20865745 \times 10^{-7}\right), \left(-5.20755942 \times 10^{-1}, \pm 9.06572930 \times 10^{-7}\right) .$$

On the other hand, Algorithm 3.1 took 4 iterations to return $\delta = 3.81345287 \times 10^{-7}$. For $\delta = 7.62690574 \times 10^{-7}$, it returned one distinct solution $(\alpha, \beta) = \left(-5.207554634 \times 10^{-1}, 0\right)$.

Example 4. In this example we took

$$A = \begin{pmatrix} -.22907968 & 0.08886286 & -.18085425 & -.03469234 & -.32819211 \\ .11868229 & -.43816868 & -.27812914 & -.04200964 & -.07784618 \\ -.02507663 & .30736050 & -.24819024 & .21852948 & -.06260819 \\ .16055050 & -.00818190 & -.19591208 & .08940924 & .22683641 \\ -.19138555 & .13088864 & -.22839105 & -.23175762 & .12274100 \end{pmatrix} .$$

and $B = (-.73491186, -.35694241, .04637973, .52703303, .22930713)^T$. Algorithm 1.1 took 7 iterations to return $\delta = 3.64238211 \times 10^{-5}$. For $\delta = 7.28476422 \times 10^{-5}$, it returned four distinct solutions

$$(\alpha, \beta) = \left(-9.48126863 \times 10^{-2}, \pm 3.91642858 \times 10^{-2}\right), \left(-9.47712590 \times 10^{-2}, \pm 4.63687903 \times 10^{-2}\right).$$

On the other hand, Algorithm 3.1 took 3 iterations to return $\delta = 3.40238900 \times 10^{-5}$. For $\delta = 6.80477800 \times 10^{-5}$, it returned two distinct solutions $(\alpha, \beta) = (-9.67874713, \pm 4.031215939 \times 10^{-2})$.

Example 5. In this example we took

$$A = \begin{pmatrix} -.27422658 & -.21968089 & -.21065336 & -.22134064 & 0.19235875 \\ -.07210867 & .18848014 & -.29068998 & .28936270 & 0.10007703 \\ -.03547166 & .17931676 & .14590007 & .00556579 & .38838791 \\ .00029995 & .14755893 & -.25420697 & -.12193382 & -.14071387 \\ -.07780546 & -.29477373 & .01366200 & .32749991 & -.0131683 \end{pmatrix} .$$

and $B = (.81475593, -.30523653, -.34286610, -.05815542, .34937688)^T$. Algorithm 1.1 took 4 iterations to return $\delta = 1.60740324 \times 10^{-7}$. For $\delta = 3.21480648 \times 10^{-7}$, it returned 14 distinct solutions to (3.2). On the other hand, Algorithm 3.1 took 2 iterations to return $\delta = 1.08714072 \times 10^{-7}$. And for $\delta = 2.17428144 \times 10^{-7}$, it returned one distinct solution $(\alpha, \beta) = (8.37424478 \times 10^{-3}, 0)$.

Example 6. In this example we took the matrix pair (A, B) from Example 2, and set

$$A := Q\ A\ Q^T \quad \text{and} \quad B := Q\ B\ ,$$

where Q is a random orthogonal matrix. This new matrix pair is still uncontrollable. But Algorithm 1.1 took 28 iterations to return $\delta = 4.06303199 \times 10^{-9}$, while Algorithm 3.1 took 6 iterations to return $\delta = 2.33160388 \times 10^{-10}$. This example illustrates that both algorithms can have numerical difficulties in correctly estimating $\rho(A, B)$ if it is very tiny.

5 Conclusions and Extensions

In this paper, we have presented the first algorithms that require a cost polynomial in the matrix size to correctly estimate the controllability distance $\rho(A, B)$ for a given linear control system. And we have demonstrated their effectiveness and reliability through some numerical experiments.

The biggest open question is how to further reduce the cost. At the core of these algorithms is the computation of all real eigenvalues of a sparse $4n^2 \times 4n^2$ eigenvalue problem. Currently, we find these eigenvalues by treating the eigenvalue problem as a dense one, resulting in algorithms that are too expensive for large problems. In the future, we plan to exploit the possibility of finding these real eigenvalues via sparse matrix computation technologies, such as the implicitly restarted Arnoldi iteration [22, 23, 28], to significantly reduce the computation cost;

Another open question is to better understand the effects of finite precision arithmetic on the estimated distance $\rho(A,B)$. As we observed in §4, if $\rho(A,B)$ is very tiny, then the distance estimated by the new algorithms in finite precision could be much larger than the exact distance.

Finally, the perturbation $[\Delta A, \Delta B]$ in (1.3) can be complex even if both A and B are real. It is known (Byers [11]) that the norm-wise smallest real perturbation can be much larger than $\rho(A,B)$. Whether our new algorithms shed new light on the computation of the norm-wise smallest real perturbation remains to be seen.

References

[1] F. Alizadeh. Interior-point methods in semidefinite programming with applications to combinatorial optimization. *SIAM J. Opt.*, 5:13–51, 1995.

[2] T. Beelen and P. Van Dooren. An improved algorithm for the computation of Kronecker's canonical form of a singular pencil. *Lin. Alg. Appl.*, 105:9–65, 1988.

[3] T. Beelen, P. Van Dooren, and M. Verhaegen. A class of staircase algorithms for generalized state space systems. In *Proceedings of the American Control Conference*, pages 425–426, Seattle, Wash., 1986.

[4] D. Boley. *Computing the controllability/observability decomposition of a linear time-invariant dynamic system: a numerical approach.* PhD thesis, Stanford University, 1981. Computer Science Dept.

[5] D. Boley and G. Golub. The Lanczos-Arnoldi algorithm and controllability. *Systems Control Lett.*, 4:317–324, 1984.

[6] D. L. Boley. Computing rank-deficiency of rectangular matrix pencils. *Systems Control Lett.*, 9:207–214, 1987.

[7] D. L. Boley and W.-S. Lu. Measuring how far a controllable system is from uncontrollable one. *IEEE Trans. Automat. Contr.*, AC-31:249–251, 1986.

[8] S. Boyd and V. Balakrishnan. A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its $\mathbf{L}_\infty$-norm. *Systems Control Lett.*, 15:1–7, 1990.

[9] S. Boyd, V. Balakrishnan, and P. Kabamba. A bisection method for computing the $\mathbf{H}_\infty$ norm of a transfer matrix and related problems. *Mathematics of Control, Signals, and Systems*, 2(3):207–219, 1989.

[10] R. Byers. A bisection method for measuring the distance of a stable matrix to the unstable matrices. *SIAM J. Sci. Stat. Comp.*, 9(5):875–881, 1988.

[11] R. Byers. Detecting nearly uncontrollable pairs. In M. A. Kaashoek, J. H. van Schuppen, and A. C. M. Ran, editors, *Numerical methods proceedings of the international symposium MTNS-89*, volume III, pages 447–457. Springer-Verlag, 1990.

[12] J. Demmel. On condition numbers and the distance to the nearest ill-posed problem. *Numer. Math.*, 51(3):251–289, July 1987.

[13] J. Demmel and B. Kågström. Accurate solutions of ill-posed problems in control theory. *SIAM J. Mat. Anal. Appl.*, 9(1):126–145, January 1988.

[14] J. Demmel and B. Kågström. The generalized Schur decomposition of an arbitrary pencil $A - \lambda B$: Robust software with error bounds and applications. Parts I and II. *ACM Trans. Math. Soft.*, 19(2), June 1993.

[15] R. Eising. The distance between a system and the set of uncontrollable systems. Memo COSOR 82-19, Eindhoven Univ. Technol., Eindhoven, The Netherlands, 1982.

[16] R. Eising. Between controllable and uncontrollable. *Syst. Contr. Lett.*, 4(5):263–264, 1984.

[17] L. Elsner and C. He. An algorithm for computing the distance to uncontrollability. *Syst. Contr. Lett.*, 17:453–464, 1991.

[18] M. Gao and M. Neumann. A global minimum search algorithm for estimating the distance to uncontrollability. *Lin. Alg. Appl.*, 188-189:305–350, 1993.

[19] C. He. Estimating the distance to uncontrollability: A fast method and a slow one. Unpublished manuscript, 1995.

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

[21] A. J. Laub. Survey of computational methods in control theory. In A. M. Erisman, K. W. Neves, and M. H. Dwarakanath, editors, *Electric Power Problems: The Mathematical Challenge*, pages 231–260, Philadelpha, PA, 1980. SIAM.

[22] R. Lehoucq. *Analysis and Implementation of an Implicitly Restarted Arnoldi Iteration.* PhD thesis, Rice University, Houston, TX, 1995.

[23] R. B. Lehoucq and D. C. Sorensen. Deflation techniques for an implicitly restarted Arnoldi iteration. *SIAM J. Mat. Anal. Appl.*, 17(6), 1996.

[24] G. Miminis. Numerical algorithms for controllability and eigenvalue location, 1981. Master Thesis, School of Computer Science, McGill University.

[25] R. D. C. Monteiro. Primal-dual path following algorithms for semidefinite programming. Unpublished manuscript, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 1995.

[26] C. C. Paige. Error analysis of some techniques for updating orthogonal decompositions. *Math. Comp.*, 34:465–471, 1980.

[27] R. V. Patel, A. J. Laub, and P. M. Van Dooren. *Numerical Linear Algebra Techniques For Systems and Control.* IEEE Press, New York, 1994.

[28] D. Sorensen. Implicit application of polynomial filters in a k-step Arnoldi method. *SIAM J. Mat. Anal. Appl.*, 13(1):357–385, 1992.

[29] M. J. Todd, K. C. Toh, and R. H. Tütüncü. On the Nesterov-Todd direction in semidefinite programming. Technical Report 1154, School of Operations Research nad Industrial Engineering, Cornell University, 1996.

[30] P. Van Dooren. The computation of Kronecker's canonical form of a singular pencil. *Lin. Alg. Appl.*, 27:103–141, 1979.

[31] M. Wicks and R. A. DeCarlo. Computing the distance to an uncontrollable system. *IEEE Trans. Auto. Contr.*, AC-36(1):39–49, 1991.