Computing Nearby Non-trivial Smith Forms

Mark Giesbrecht, Joseph Haraldson, George Labahn

Cheriton School of Computer Science, University of Waterloo, Waterloo, Canada

Abstract

We consider the problem of computing the nearest matrix polynomial with a non-trivial Smith Normal Form. We show that computing the Smith form of a matrix polynomial is amenable to numeric computation as an optimization problem. Furthermore, we describe an effective optimization technique to find a nearby matrix polynomial with a non-trivial Smith form. The results are then generalized to include the computation of a matrix polynomial having a maximum specified number of ones in the Smith Form (i.e., with a maximum specified McCoy rank).

We discuss the geometry and existence of solutions and how our results can be used for an error analysis. We develop an optimization-based approach and demonstrate an iterative numerical method for computing a nearby matrix polynomial with the desired spectral properties. We also describe an implementation of our algorithms and demonstrate the robustness with examples in `Maple`.

1. Introduction

For a given square matrix polynomial $\mathcal{A} \in \mathbb{R}[t]^{n\times n}$, one can find unimodular matrices $\mathcal{U}, \mathcal{V} \in \mathbb{R}[t]^{n\times n}$ such that $\mathcal{U}\mathcal{A}\mathcal{V}$ is a diagonal matrix $\mathcal{S}$. Unimodular means that there is a polynomial inverse matrix, or equivalently, that the determinant is a nonzero constant from $\mathbb{R}$. The unimodular matrices $\mathcal{U}, \mathcal{V}$ encapsulate the polynomial row and column operations, respectively, needed for such a diagonalization. The best known diagonalization is the Smith Normal Form (SNF, or simply Smith form) of a matrix polynomial. Here

$$\mathcal{S} = \begin{pmatrix} s_1 & & & \\ & s_2 & & \\ & & \ddots & \\ & & & s_n \end{pmatrix} \in \mathbb{R}[t]^{n\times n},$$

where $s_1, \ldots, s_n \in \mathbb{R}[t]$ are monic and $s_i \mid s_{i+1}$ for $1 \le i < n$. The Smith form always exists and is unique though the associated unimodular matrices $\mathcal{U}, \mathcal{V}$ are not unique (see, e.g., (Kailath, 1980; Gohberg et al., 2009)). The diagonal entries $s_1, \ldots, s_n$ are referred to as the *invariant factors* of $\mathcal{A}$.

Email addresses: `mwg@uwaterloo.ca` (Mark Giesbrecht), `mwg@uwaterloo.ca` (Mark Giesbrecht), `jharalds@uwaterloo.ca` (Joseph Haraldson), `glabahn@uwaterloo.ca` (George Labahn)

URL: `https://cs.uwaterloo.ca/~mwg` (Mark Giesbrecht), `https://cs.uwaterloo.ca/~jharalds` (Joseph Haraldson), `https://cs.uwaterloo.ca/~glabahn` (George Labahn)

Matrix polynomials and their Smith forms are used in many areas of computational algebra, control systems theory, differential equations and mechanics. The Smith form is important as it effectively reveals the structure of the polynomial lattice of rows and columns, as well as the effects of localizing at individual eigenvalues. That is, it characterizes how the rank decreases as the variable t is set to different values (especially eigenvalues, where the rank drops). The Smith form is closely related to the more general *Smith-McMillan form* for matrices of rational functions, a form that reveals the structure of the eigenvalue at infinity, or the infinite spectral structure. The eigenvalue at infinity is non-trivial if the leading coefficient matrix is rank deficient or equivalently, the determinant does not achieve the generic degree.

The algebra of matrix polynomials is typically described assuming that the coefficients are fixed and come from an exact arithmetic domain, usually the field of real or complex numbers. In this exact setting, computing the Smith form has been well studied, and very efficient procedures are available (see (Kaltofen and Storjohann, 2015) and the references therein). However, in some applications, particularly control theory and mechanics, the coefficients can come from measured data or contain some amount of uncertainty. Compounding this, for efficiency reasons such computations are usually performed using floating point to approximate computations in $\mathbb{R}$, introducing roundoff error. As such, algorithms must accommodate numerical inaccuracies and are prone to numerical instability.

Numerical methods to compute the Smith form of a matrix polynomial typically rely on linearization and orthogonal transformations (Van Dooren and Dewilde, 1983; Beelen and Van Dooren, 1988; Demmel and Kågström, 1993a,b; Demmel and Edelman, 1995) to infer the Smith form of a nearby matrix polynomial via the Jordan blocks in the Kronecker canonical form (see (Kailath, 1980)). These linearization techniques are numerically backwards stable, and for many problems this is sufficient to ensure that the computed solutions are computationally useful when a problem is continuous.

Unfortunately, the eigenvalues of a matrix polynomial are not necessarily continuous functions of the coefficients of the matrix polynomial, and backwards stability is not always sufficient to ensure computed solutions are useful in the presence of discontinuities. Previous methods are also unstructured in the sense that the computed non-trivial Smith form may not be the Smith form of a matrix polynomial with a prescribed coefficient structure, for example, maintaining the degree of entries or not introducing additional non-zero entries or coefficients. In extreme instances, the unstructured backwards error can be arbitrarily small, while the structured distance to an interesting Smith form is relatively large. Finally, existing numerical methods can also fail to compute meaningful results on some problems due to uncertainties. Examples of such problems include nearly rank deficient matrix polynomials, repeated eigenvalues or eigenvalues that are close together and other ill-posed instances.

In this paper we consider the problem of computing a nearby matrix polynomial with a prescribed spectral structure, broadly speaking, the degrees and multiplicities of the invariant factors in the Smith form, or equivalently the structure and multiplicity of the eigenvalues of the matrix polynomial. The results presented in this paper extend those in the conference paper (Giesbrecht, Haraldson, and Labahn, 2018). This work is not so much about computing the Smith normal form of a matrix polynomial using floating point arithmetic, but rather our focus is on the computation of a nearby matrix polynomial with "an interesting" or non-generic Smith normal form. The emphasis in this work is on the finite spectral structure of a matrix polynomial, since the techniques described are easily generalized to handle the instance of the infinite spectral structure as a special case. made some changes here...

The Smith form of a matrix polynomial is not continuous with respect to the usual topology

of the coefficients and the resulting limitations of backward stability is not the only issue that needs to be addressed when finding nearest objects in an approximate arithmetic environment. A second issue can be illustrated by recalling the well-known representation of the invariant factors $s_1, \ldots, s_n$ of a matrix $\mathcal{A} \in \mathbb{R}[t]^{n\times n}$ as ratios $s_i = \delta_i/\delta_{i-1}$ of the *determinantal divisors* $\delta_0, \delta_1, \ldots, \delta_n \in \mathbb{R}[t]$, where

$$\delta_0 = 1, \quad \delta_i = \mathrm{GCD}\Big\{\text{all } i \times i \text{ minors of } \mathcal{A}\Big\} \in \mathbb{R}[t].$$

In the case of 2×2 matrix polynomials, computing the nearest non-trivial Smith form is thus equivalent to finding the nearest matrix polynomial whose polynomial entries have a non-trivial GCD. Recall that approximate GCD problems can have infima that are *unattainable*. That is, there are co-prime polynomials with nearby polynomials with a non-trivial GCD at distances arbitrarily approaching an infimum, while at the infimum itself the GCD is trivial (see, e.g., (Giesbrecht, Haraldson, and Kaltofen, 2017a)).

The issue of unattainable infima extends to the Smith normal form. As an example, consider

$$\mathcal{A} = \begin{pmatrix} t^2 - 2t + 1 & \\ & t^2 + 2t + 2 \end{pmatrix} = \mathrm{diag}(f, g) \in \mathbb{R}[t]^{2\times 2}.$$

If we look for nearby polynomials $\widetilde{f}, \widetilde{g} \in \mathbb{R}[t]$ of degree at most 2 such that $\gcd(\widetilde{f}, \widetilde{g}) = \gamma t + 1$ (i.e. a nontrivial gcd) at minimal distance $\|f - \widetilde{f}\|_2^2 + \|g - \widetilde{g}\|_2^2$ for some $\gamma \in \mathbb{R}$, then it is shown in (Haraldson, 2015, Example 3.3.6) that this distance is $(5\gamma^4 - 4\gamma^3 + 14\gamma^2 + 2)/(\gamma^4 + \gamma^2 + 1)$. This distance has an infimum of 2 at $\gamma = 0$. However at $\gamma = 0$ we have $\gcd(\widetilde{f}, \widetilde{g}) = 1$ even though $\deg(\gcd(\widetilde{f}, \widetilde{g})) > 0$ for all $\gamma \neq 0$. For $\mathcal{A}$ to have a non-trivial Smith form we must perturb f, g such that they have a non-trivial GCD, and thus any such perturbation must be at a distance of at least 2. However, the perturbation of distance precisely 2 has a trivial Smith form. There is no merit to perturbing the off-diagonal entries of $\mathcal{A}$.

Our work indirectly involves measuring the sensitivity to the eigenvalues of $\mathcal{A}$ and the determinant of $\mathcal{A}$. Thus we differ from most sensitivity and perturbation analysis (e.g., (Stewart, 1994; Ahmad and Alam, 2009)), since we also study how perturbations affect the invariant factors, instead of the roots of the determinant. Additionally our theory is able to support the instance of $\mathcal{A}$ being rank deficient and having degree exceeding one. One may also approach the problem geometrically in the context of manifolds (Edelman et al., 1997, 1999). We do not consider the manifold approach directly since it does not yield numerical algorithms.

Determining what it means for a matrix polynomial to have a non-trivial Smith form numerically and finding the distance from one matrix polynomial to another matrix polynomial having an interesting or non-trivial Smith form are formulated as finding solutions to continuous optimization problems. The main contributions of this paper are deciding when $\mathcal{A}$ has an interesting Smith form, providing bounds on a "radius of triviality" around $\mathcal{A}$ and a structured stability analysis on iterative methods to compute a structured matrix polynomial with desired spectral properties.

The remainder of the paper is organized as follows. In Section 2 we give the notation and terminology along with some needed background used in our work. Section 3 discusses the approximate Smith form computation as an optimization problem and provides some new bounds on the distance to non-triviality. We present an optimization algorithm in Section 4 with local stability properties and rapid local convergence to compute a nearby matrix polynomial with a non-trivial Smith form and discuss implementation details. A method to compute a matrix

polynomial with a prescribed lower bound on the number of ones in the Smith form is discussed in Section 5. We discuss our implementation and include some examples in Section 6. The paper ends with a conclusion along with topics for future research.

A preliminary version of this paper appears in the Proceedings of the ISSAC 2018 conference (Giesbrecht, Haraldson, and Labahn, 2018), and has been submitted to this special issue. For the benefit of the referees, a list of changes appears in the Appendix Section 8.

2. Preliminaries

In this section we give the notation and formal definitions needed to precisely describe the problems summarized above. We also present some existing results used as building blocks for our work. In addition, we provide a basic description of matrix functions and their first-order derivatives (Jacobian matrices) which will be needed for the optimization work central to our results.

2.1. Notation and Terminology

We make extensive use of the following terminology and definitions. A matrix polynomial $\mathcal{A} \in \mathbb{R}[t]^{n\times n}$ is an $n \times n$ matrix whose entries are polynomials. Typically we also work with matrices whose entries have degree bound d and use the notation $\mathbb{R}_{\le d}[t]^{n\times n}$ to describe this set. Alternatively, we may express matrix polynomials as $\mathcal{A} = \sum_{1\le j\le d} A_j t^j$ where $A_j \in \mathbb{R}^{n\times n}$. The *degree* of a matrix polynomial d is defined to be the degree of the highest-order non-zero entry of $\mathcal{A}$, or the largest index j such that $A_j \neq 0$. We say that $\mathcal{A}$ has *full rank* or is *regular* if $\det(\mathcal{A}) \not\equiv 0$. As noted in the introduction, $\mathcal{A}$ is said to be *unimodular* if $\det(\mathcal{A}) \in \mathbb{R}\backslash\{0\}$. The *(finite) eigenvalues* are the roots of $\det(A) \in \mathbb{R}[t]$. The norm of a polynomial $a \in \mathbb{R}[t]$ is defined as $\|a\| = \|a\|_2 = \|(a_0, a_1, \ldots, a_d, 0, \ldots, 0)\|_2$. For matrix polynomials we define $\|\mathcal{A}\| = \|\mathcal{A}\|_F = \sqrt{\sum_{i,j} \|\mathcal{A}_{i,j}\|_2^2}$. Our choice of norm is a distributed coefficient norm, sometimes known as the *Frobenius norm*.

Definition 2.1 (SVD – Golub and Van Loan 2012)**.** *The Singular Value Decomposition (SVD) of $A \in \mathbb{R}^{n\times n}$ is given by $U^T\Sigma V$, where $U, V \in \mathbb{R}^{n\times n}$ satisfy $U^TU = I$, $V^TV = I$ and $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_n)$ is a diagonal matrix with non-negative real entries in descending order of magnitude, the* singular values *of A. The distance to the nearest (unstructured) matrix of rank $m < n$ is $\sigma_{m+1}(A)$.*

For scalar matrices we frequently write $\|\cdot\|_2$ for the largest singular value, and $\sigma_{\min}(\cdot)$ for the smallest singular value.

In this paper we are mainly concerned with coefficient structures that preserve the zero coefficient structure of a matrix polynomial, that is, we generally do not change zero coefficients to non-zero, or increase the degrees of matrix entries.

Definition 2.2 (Affine/Linear Structure)**.** *A non-zero matrix polynomial $\mathcal{A} \in \mathsf{R}[t]^{n\times n}$ of degree at most d has a* linear structure *from a set $\mathcal{K}$ if $\mathcal{A} \in \operatorname{span}(\mathcal{K})$ as a vector space over R, where*

$$\mathcal{K} = \{C_{0,0}, \ldots, C_{0,k}, tC_{1,0}, \ldots, tC_{1,k}, \ldots, t^dC_{d,0}, \ldots, t^dC_{d,k}\},$$

where $C_{l,j} \in \mathsf{R}^{n\times n}$ for $0 \le j \le k$, where $k > 0$ is a finite index variable. If $\mathcal{A} = \mathcal{C}_0 + \mathcal{C}_1$, where $\mathcal{C}_0 \in \mathsf{R}[t]^{n\times n}$ is fixed and $\mathcal{C}_1 \in \operatorname{span}(\mathcal{K})$, then $\mathcal{A}$ is said to have an affine *structure from the set $\mathcal{K}$.*

I changed this. Removed LI and said k was an index variable. Linearly and affine linearly structured matrices are best thought of as imposing linear equality constraints on the entries. Examples of matrices with a linear structure include matrices with prescribed zero entries or coefficients, Toeplitz/Hankel matrices, Sylvester matrices, resultant-like matrices, Ruppert matrices and several other matrices appearing in symbolic-numeric computation. Matrices with an affine structure include all matrices with a linear structure and, in addition, matrices having prescribed non-zero constant entries/coefficients, for example monic matrix polynomials.

Recall that the *rank* of a matrix polynomial is the maximum number of linearly independent rows or columns as a vector space over $\mathbb{R}(t)$. This is the same as the rank of the matrix $\mathcal{A}(\omega)$ for any $\omega \in \mathbb{C}$ that is not an eigenvalue of $\mathcal{A}(t)$. If we allow evaluation at eigenvalues, then the McCoy rank is the lowest rank when $\mathcal{A}$ is evaluated at an eigenvalue.

Definition 2.3 (McCoy Rank and Non-Trivial SNF)**.** *The McCoy rank of $\mathcal{A}$ is* $\min_{\omega\in\mathbb{C}}\{\operatorname{rank}\mathcal{A}(\omega)\}$*, the lowest rank possible when $\mathcal{A}$ is evaluated at any $\omega \in \mathbb{C}$. Note that the rank of $\mathcal{A}$ only drops at all if it is evaluated at an eigenvalue $\omega \in \mathbb{C}$. The McCoy rank is also the number of ones in the Smith form. Equivalently, if $\mathcal{A}$ has r non-trivial invariant factors, then the McCoy rank of $\mathcal{A}$ is $n - r$. The matrix polynomial $\mathcal{A}$ is said to have a* non-trivial or interesting Smith normal form *if the McCoy rank is at most $n - 2$, or equivalently, if it has two or more invariant factors of non-zero degree.*

I changed this to allow zero as an invariant factor. It was wrong earlier.

Problem 2.4 (Approximate SNF Problem)**.** *Given a matrix polynomial $\mathcal{A} \in \mathbb{R}[t]^{n\times n}$, find the distance to a non-trivial SNF. Find a matrix polynomial $\widehat{\mathcal{A}} \in \mathbb{R}[t]^{n\times n}$ of prescribed coefficient structure that has a prescribed McCoy rank of at most $n - r$ for $r \geq 2$ such that $\|\mathcal{A} - \widehat{\mathcal{A}}\|$ is minimized under $\|\cdot\|$, if such an $\widehat{\mathcal{A}}$ exists.*

We will consider $\|\cdot\| = \|\cdot\|_F$ to be the Frobenius norm. Computing the nearest (if it exists) McCoy rank $n - 2$ matrix is the *approximate SNF*.

Problem 2.5 (Lower McCoy Rank Approximation Problem)**.** *Computing the nearest (if it exists) McCoy rank $n - r$ matrix for $r \geq 3$ is a lower McCoy rank approximation.*

In a generic sense, the nearest matrix polynomial with an interesting SNF will have McCoy rank $n-2$ with probability one, but many matrices arising from applications are expected to have more interesting (i.e. the invariant factors have a richer or non-generic multiplicity structure) Smith forms nearby. made a change here

As described in the introduction, it is possible that the distance to a non-trivial SNF is not attainable. That is, there is a solution that is approached asymptotically, but where the Smith form is trivial at the infimum. Fortunately, in most instances of interest, solutions will generally be attainable. We will also later discuss how to identify and compute unattainable solutions. Problem 2.4 and Problem 2.5 admit the nearest rank $n - 1$ or rank $n - 2$ matrix polynomial as a special case. However, the computational challenges are fundamentally different for non-trivial instances.

2.2. Embedding into Scalar Domains

In our study of nearest non-trivial Smith forms we often make use of the representation of the diagonal elements as ratios of GCDs of sub-determinants. When the coefficients are polynomials

with numeric coefficients it is helpful to embed the arithmetic operations of polynomial multiplication and polynomial GCD into a matrix problem having numeric coefficients (i.e., from $\mathbb{R}$). Such an embedding allows us to employ a number of tools, including condition numbers and perturbations of matrix functions.

We start with some basic notation for mapping matrices and polynomials to vectors.

Definition 2.6 (Vec Operator)**.** *We define the operator* vec : $\mathbb{R}[t] \to \mathbb{R}^{(d+1)\times 1}$ *as follows:*

$$p = \sum_{j=0}^{d} p_j t^j \in \mathbb{R}[t] \quad \mapsto \quad \text{vec}(p) = (p_0, p_1, \ldots, p_d)^T \in \mathbb{R}^{(d+1)\times 1}$$

The vec *operator* vec$(\cdot)$ *is extended to map a matrix* $\mathbb{R}[t]^{m\times n}$ *to a single vector in* $\mathbb{R}^{mn(d+1)\times 1}$ *by stacking columns of (padded) coefficient vectors on top of each other.*

$$\mathcal{A} \in \mathbb{R}[t]^{m\times n} \mapsto \text{vec}(\mathcal{A}) = \begin{pmatrix} \text{vec}(\mathcal{A}_{11}) \\ \vdots \\ \text{vec}(\mathcal{A}_{mn}) \end{pmatrix} \in \mathbb{R}^{mn(d+1)\times 1}.$$

It is sometimes useful to reduce matrix polynomials to vectors of polynomials in $\mathbb{R}[t]$ rather than vectors over $\mathbb{R}$.

Definition 2.7 (Polynomial Vec Operator)**.** *The* pvec *operator maps* $\mathbb{R}[t]^{m\times n}$ *to a vector* $\mathbb{R}[t]^{nm\times 1}$ *as*

$$\mathcal{A} \in \mathbb{R}[t]^{m\times n} \mapsto \text{pvec}(\mathcal{A}) = \begin{pmatrix} \mathcal{A}_{11} \\ \vdots \\ \mathcal{A}_{mn} \end{pmatrix} \in \mathbb{R}[t]^{mn\times 1}.$$

We define the vectorization of matrix polynomials in this somewhat non-standard way so that we can later facilitate the computation of derivatives of matrix polynomial valued functions.

To describe polynomial multiplication in terms of linear maps over scalars we have:

Definition 2.8 (Convolution Matrix)**.** *Polynomial multiplication between polynomials* $a, b \in \mathbb{R}[t]$*, of degrees* d_1 *and* d_2*, respectively may be expressed as a Toeplitz-matrix-vector product. We define*

$$\phi_{d_2}(a) = \begin{pmatrix} a_0 & & \\ \vdots & \ddots & \\ a_{d_1} & & a_0 \\ & \ddots & \vdots \\ & & a_{d_1} \end{pmatrix} \in \mathbb{R}^{(d_1+d_2+1)\times(d_2+1)}. \qquad \textit{It follows that} \quad \text{vec}(ab) = \phi_{d_2}(a)\text{vec}(b).$$

When a is non-zero, we can also define division through pseudo-inversion or linear least squares. In a similar manner, we can define the product of matrix polynomials through a Toeplitz-block matrix.

Definition 2.9 (Block Convolution Matrix)**.** *We can express multiplication of a matrix and vector of polynomials, $\mathcal{A} \in \mathbb{R}[t]^{m\times n}$ and $\mathcal{b} \in \mathbb{R}[t]^{n\times 1}$, of degrees at most d_1 and d_2 respectively, as a scalar linear system*

$$\operatorname{vec}(\mathcal{A}\mathcal{b}) = \Phi_{d_2}(\mathcal{A})\operatorname{vec}(\mathcal{b}),$$

where

$$\Phi_{d_2}(\mathcal{A}) = \begin{pmatrix} \phi_{d_2}(\mathcal{A}_{11}) & \cdots & \phi_{d_2}(\mathcal{A}_{1n}) \\ \vdots & & \vdots \\ \phi_{d_2}(\mathcal{A}_{m1}) & \cdots & \phi_{d_2}(\mathcal{A}_{mn}) \end{pmatrix} \in \mathbb{R}^{m(d_1+d_2+1)\times n(d_2+1)}.$$

The block convolution matrix is sometimes referred to as a "Sylvester matrix" associated with $\mathcal{A}$. However, we reserve the term "Sylvester matrix" for the more standard linear system appearing from the GCD of two (or more) polynomials. The block convolution matrix is a scalar matrix whose entries have a linear (Toeplitz-block) structure.

Definition 2.10 (Kronecker Product)**.** *The Kronecker product of $\mathcal{A} \in \mathbb{R}[t]^{m\times n}$ and $\mathcal{B} \in \mathbb{R}[t]^{k\times \ell}$ denoted as $\mathcal{A} \otimes \mathcal{B}$ is the $mk \times n\ell$ matrix over $\mathbb{R}[t]$ defined as*

$$\mathcal{A} \otimes \mathcal{B} = \begin{pmatrix} \mathcal{A}_{11}\mathcal{B} & \cdots & \mathcal{A}_{1n}\mathcal{B} \\ \vdots & & \vdots \\ \mathcal{A}_{m1}\mathcal{B} & \cdots & \mathcal{A}_{mn}\mathcal{B} \end{pmatrix} \in \mathbb{R}[t]^{mk\times n\ell}.$$

This definition of Kronecker product, sometimes referred to as the "outer product", also holds for scalar matrices (and vectors).

Lemma 2.11. *For scalar matrices of compatible dimension A, X and B over $\mathbb{R}$, we can*

$$\operatorname{vec}(AXB) = (B^T \otimes A)\operatorname{vec}(X).$$

Likewise, for matrix polynomials $\mathcal{A}, \mathcal{X}$ and $\mathcal{B}$ of compatible dimension over $\mathbb{R}[t]$, we have

$$\operatorname{pvec}(\mathcal{A}\mathcal{X}\mathcal{B}) = (\mathcal{B}^T \otimes \mathcal{A})\operatorname{pvec}(\mathcal{X}).$$

The Kronecker product can also be used to re-write matrix equations of the form $AX = B$, for matrices A, B and X of compatible dimensions, to

$$\operatorname{vec}(AX) = (X^T \otimes I)\operatorname{vec}(A) = (I \otimes A)\operatorname{vec}(X) = \operatorname{vec}(B).$$

2.3. Derivatives of Matrix Polynomial Valued Functions

In this paper we will need to compute derivatives of some important matrix polynomial valued functions, namely the determinant and adjoint. This problem is approached in the context of computing the Jacobian matrix of a vector valued function. The analysis in this section will be useful for showing that Lagrange multipliers typically exist in the optimization problems encountered. The quantities computed can also be used to derive first-order perturbation bounds for these matrix polynomial valued functions with respect to $\|\cdot\|_F$.

Recall that the adjoint of a matrix polynomial $\mathcal{A} \in \mathbb{R}[t]^{n\times n}$, denoted by $\operatorname{Adj}(\mathcal{A}) \in \mathbb{R}[t]^{n\times n}$, is the transpose of the cofactor matrix. Thus $\operatorname{Adj}(\mathcal{A})_{ij} = (-1)^{i+j}\det(\mathcal{A}[j|i])$ where $\mathcal{A}[j|i]$ is the (j,i) cofactor of $\mathcal{A}$, that is, the matrix formed by removing row j and column i from $\mathcal{A}$. When $\mathcal{A}$ has full rank, $\mathcal{A}$ satisfies $\mathcal{A}\operatorname{Adj}(\mathcal{A}) = \det(\mathcal{A})I$.

The determinant of an $n \times n$ matrix polynomial having entries of degree at most d can be viewed as a mapping from $\mathbb{R}^{n^2(d+1)} \to \mathbb{R}^{nd+1}$, since the determinant has degree at most nd. With this same viewpoint, we can view the adjoint of a matrix polynomial as a mapping from $\mathbb{R}^{n^2(d+1)} \to \mathbb{R}^{n^2((n-1)d+1)}$, since the degree of the entries of the adjoint are at most $(n-1)d$. Our notation for computing derivatives of vector valued functions follows that of (Magnus and Neudecker, 1988).

It is not surprising that the determinant of a matrix polynomial has a similar identity (Magnus and Neudecker, 1988) to the well-known scalar identity $\nabla \det(A) = \text{vec}((\text{Adj}(A)^T)^T)$.

Theorem 2.12. *Let $\mathcal{A} \in \mathbb{R}[t]^{n \times n}$ have degree at most d, then*

$$J_{\det} = \frac{\partial \text{vec}(\det(\mathcal{A}))}{\partial \text{vec}(\mathcal{A})} = \Phi_d(\text{pvec}(\text{Adj}(\mathcal{A})^T)^T) \in \mathbb{R}^{(nd+1) \times n^2(d+1)}.$$

Proof. We note that from generalizing the scalar identity $\nabla \det(\cdot) = \text{vec}(\text{Adj}(\cdot)^T)^T$, we can write a first-order expansion of the determinant as

$$\det(\mathcal{A} + \Delta\mathcal{A}) = \det(\mathcal{A}) + \text{pvec}(\text{Adj}(\mathcal{A})^T)^T \text{pvec}(\Delta\mathcal{A}) + O(\|\Delta\mathcal{A}\|_F^2),$$

and ignoring higher-order terms we obtain the scalar expression

$$\text{vec}(\det(\mathcal{A} + \Delta\mathcal{A})) \approx \text{vec}(\det(\mathcal{A})) + \text{vec}(\text{pvec}(\text{Adj}(\mathcal{A})^T)^T \text{pvec}(\Delta\mathcal{A})).$$

The Jacobian can be extracted by (padding with zero coefficient entries as necessary) writing $\text{vec}(\text{pvec}(\text{Adj}(\mathcal{A})^T)^T \text{pvec}(\Delta\mathcal{A})) = J_{\det}\text{vec}(\Delta\mathcal{A})$ as a matrix-vector product. Thus, using block-convolution matrices we have

$$\frac{\partial \text{vec}(\det(\mathcal{A}))}{\partial \text{vec}(\mathcal{A})} = \nabla(\det(\mathcal{A})) = \Phi_d(\text{pvec}(\text{Adj}(\mathcal{A})^T)^T). \qquad \square$$

Now that we have a closed-form expression for the derivative of the determinant, it is useful to derive a closed-form expression for the adjoint matrix. The closed-form expression reveals rank information, and is independently useful for optimization algorithms requiring derivatives. The rank information is useful to obtain insights about the existence of Lagrange multipliers. If $J_{\det}$ has full or locally constant (row) rank then constraint qualifications will hold for several constrained optimization problems involving the determinant. If $\text{Adj}(\mathcal{A})$ is non-zero then one can often infer the existence of Lagrange multipliers for other problems as well.

Theorem 2.13. *Let $\mathcal{A} \in \mathbb{R}[t]^{n \times n}$ have degree at most d and rank n. The Jacobian of $\text{Adj}(\mathcal{A})$ is $J_{\text{Adj}} \in \mathbb{R}^{(n^2((n-1)d+1)) \times n^2(d+1)}$ with*

$$J_{\text{Adj}} = \left[\Phi_{(n-1)d}(I \otimes \mathcal{A})\right]^+ \left[\Phi_d(\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T) - \Phi_d(\text{Adj}(\mathcal{A})^T \otimes I)\right],$$

where I is understood to be the $n \times n$ identity matrix and for a scalar matrix A of full rank, A^+ is the Moore-Penrose pseudo-inverse arising from the SVD.

Proof. First recall that if $\mathcal{A}$ has full rank, then $\mathcal{A}\,\text{Adj}(\mathcal{A}) = \text{Adj}(\mathcal{A})\mathcal{A} = \det(\mathcal{A})I$. This expression defines the adjoint matrix when $\mathcal{A}$ has full rank. We can write

$$\text{pvec}(\mathcal{A}\,\text{Adj}(\mathcal{A})) = (\text{Adj}(\mathcal{A})^T \otimes I)\text{pvec}(\mathcal{A}) = (I \otimes \mathcal{A})\text{pvec}(\text{Adj}(\mathcal{A})),$$

thus converting to a linear system over $\mathbb{R}$ produces

$$\text{vec}(\mathcal{A}\ \text{Adj}(\mathcal{A})) = \Phi_{(n-1)d}(I \otimes \mathcal{A})\text{vec}(\text{Adj}(\mathcal{A})) = \Phi_d(\text{Adj}(\mathcal{A})^T \otimes I)\text{vec}(\mathcal{A}).$$

Applying the product rule yields

$$\partial\text{vec}(\mathcal{A}\ \text{Adj}(\mathcal{A})) = (\partial\Phi_{(n-1)d}(I \otimes \mathcal{A}))\text{vec}(\text{Adj}(\mathcal{A})) + \Phi_{(n-1)d}(I \otimes \mathcal{A})\partial\text{vec}(\text{Adj}(\mathcal{A})). \quad (1)$$

Next we observe that (1) has the same coefficients as the expression

$$\text{vec}((\partial\mathcal{A})\ \text{Adj}(\mathcal{A}) + \mathcal{A}(\partial\ \text{Adj}(\mathcal{A})))$$

which is equivalent to

$$\text{vec}((\text{Adj}(\mathcal{A})^T \otimes I)\text{pvec}(\partial\mathcal{A}) + (I \otimes \mathcal{A})\text{pvec}(\partial\ \text{Adj}(\mathcal{A}))),$$

which reduces to

$$\Phi_d((\text{Adj}(\mathcal{A})^T \otimes I))\text{vec}(\partial\mathcal{A}) + \Phi_{(n-1)d}(I \otimes A)\text{vec}(\partial\ \text{Adj}(\mathcal{A})). \quad (2)$$

We now have the derivative of the left hand side the expression $\mathcal{A}\ \text{Adj}(\mathcal{A}) = \det(\mathcal{A})I$. Differentiation of the right hand side yields

$$\partial\text{vec}(\det(\mathcal{A})I) = \text{vec}(\partial\text{pvec}(\det(\mathcal{A})I)),$$

which is equivalent to the expression

$$\text{vec}(\partial\text{pvec}(\det(\mathcal{A})I)) = \text{vec}(\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T\text{pvec}(\partial\mathcal{A})). \quad (3)$$

Converting (3) into a linear system over $\mathbb{R}$ leads to

$$\text{vec}(\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T)\text{pvec}(\partial\mathcal{A}) = \Phi_d(\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T)\text{vec}(\partial\mathcal{A}), \quad (4)$$

which is the derivative of the right-hand side.

Combining (2) and (4) we have

$$\Phi_{(n-1)d}(I \otimes \mathcal{A})\frac{\partial\text{vec}(\text{Adj}(\mathcal{A}))}{\partial\text{vec}(\mathcal{A})} = \Phi_d(\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T) - \Phi_d(\text{Adj}(\mathcal{A})^T \otimes I).$$

Assuming that $\mathcal{A}$ has full rank so $\Phi_{(n-1)d}(\text{pvec}(I \otimes \mathcal{A}))$ is pseudo-invertible, we can write

$$J_{\text{Adj}} = [\Phi_{(n-1)d}(I \otimes \mathcal{A})]^+ \left[\Phi_d(\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T) - \Phi_d(\text{Adj}(\mathcal{A})^T \otimes I)\right],$$

which completes the proof. □

An observation that is important later is that the derivative of the adjoint has a Toeplitz-block structure. More importantly, the bandwidth is $O(d)$, and we only need to compute $O(n^2)$ columns instead of $O(n^2d)$. We also note that J_{Adj} may be padded with zeros, since $\mathcal{A}$ may not have generic degrees.

Corollary 2.14. *If $\mathcal{A}$ has full rank then J_{Adj} has full rank.*

Proof. The matrix $\Phi_{(n-1)d}(I \otimes \mathcal{A})$ has full rank since $I \otimes \mathcal{A}$ has full rank. The matrix

$$\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T - \text{Adj}(\mathcal{A})^T \otimes I = -\left(-\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T + \text{Adj}(\mathcal{A})^T \otimes I\right) \quad (5)$$

is a rank one update to a matrix polynomial. By evaluating (5) at a complex number ω that is not an eigenvalue of $\mathcal{A}$ we can show that (5) has full rank. Let $A = \mathcal{A}(\omega)$, so $A \in \mathbb{R}^{n\times n}$ has full rank.

Using the Sherman-Morrison formula (Higham, 2002, pg. 487) for rank 1 updates to a matrix, we need to verify that

$$1 - \text{vec}(\text{Adj}(A)^T)^T \left[\left(\text{Adj}(A)^T\right)^{-1} \otimes I\right] \text{vec}(I) \neq 0,$$

in order to ensure that (5) has full rank. We have that

$$\begin{aligned}\text{vec}(\text{Adj}(A)^T)^T \left[\left(\text{Adj}(A)^T\right)^{-1} \otimes I\right] \text{vec}(I) &= \text{vec}(\text{Adj}(A)^T)^T \text{vec}\left(\text{Adj}(A)^T)^{-1}\right) \\ &= \text{Tr}\left(\text{Adj}(A)^T \left(\text{Adj}(A)^T\right)^{-1}\right) \\ &= n,\end{aligned}$$

thus (5) has full rank. Note we used the identities for matrices X, Y and Z of appropriate dimension, that $\text{vec}(XYZ) = (Z^T \otimes X)\text{vec}(Y)$ and $\text{vec}(X^T)^T\text{vec}(Y) = \text{Tr}(XY)$. Again, we have that

$$\Phi_d(\text{pvec}(I)\text{pvec}(\text{Adj}(\mathcal{A})^T)^T) - \Phi_d(\text{Adj}(\mathcal{A})^T \otimes I)$$

has full rank, thus J_{Adj} is a product of two matrices of full rank, so J_{Adj} must also have full rank. □

Corollary 2.14 implies that Lagrange multipliers will exist to several optimization problems involving the adjoint matrix as a constraint, since the Jacobian matrix of the adjoint has full rank. The linear independent constraint qualification or the constant rank constraint qualification will hold for several optimization problems of the form

$$\min \|\Delta\mathcal{A}\| \text{ subject to } \text{Adj}(\mathcal{A} + \Delta\mathcal{A}) = \mathcal{F},$$

for some reasonably prescribed $\mathcal{F} \in \mathbb{R}[t]^{n\times n}$.

Remark 2.15. *If $\mathcal{A}$ is rank deficient, then the derivative is still defined, but not necessarily by Theorem 2.13. If $\text{rank}(\mathcal{A}) \leq n-3$ then $J_{\text{Adj}} = 0$, since all $(n-3)\times(n-3)$ minors vanish (J_{Adj} consists of the coefficients of these minors). If $\text{rank}(\mathcal{A}) = n-1$ or $\text{rank}(\mathcal{A}) = n-2$ then J_{Adj} is still defined and in both cases $J_{\text{Adj}} \neq 0$. However J_{Adj} is not necessarily described by Theorem 2.13.*

For several affine or linear perturbation structures (such as ones that preserve the degree of entries or the support of entries), Theorem 2.13 and the associated Corollary 2.14 will hold (after deleting some extraneous rows or columns).

3. When Does a Numerical Matrix Polynomial have a trivial SNF?

In this section we consider the question of determining if a matrix polynomial has a non-trivial SNF, or rather how much do the coefficients need to be perturbed to have a non-trivial SNF. We provide a lower bound on this distance by analyzing the distance to a reduced-rank generalized Sylvester matrix.

3.1. Embeddings into generalized Sylvester matrices and approximate GCDs

In the introduction we demonstrated that some nearby non-trivial Smith Forms are unattainable. In this subsection we investigate why these unattainable values occur. We first review some basic results needed to analyze the topology of the approximate Smith form problem.

For a matrix $\mathcal{A} \in \mathbb{R}[x]^{n\times n}$, we know that $s_n = \delta_n/\delta_{n-1}$, the quotient of the determinant and the GCD of all $(n-1)\times(n-1)$ minors. Since these minors are precisely the entries of the adjoint matrix, it follows that $\mathcal{A}$ has a non-trivial Smith form if and only if the GCD of all entries of the adjoint is non-trivial, that is, $\deg(\gcd(\{\mathrm{Adj}(\mathcal{A})_{ij}\})) \geq 1$. In order to obtain bounds on the distance to a matrix having a non-trivial Smith form, we consider an approximate GCD problem of the form

$$\min\left\{\|\Delta\mathcal{A}\| \text{ subject to } \deg\left(\gcd\left\{\mathrm{Adj}\left(\mathcal{A}+\Delta\mathcal{A}\right)_{ij}\right\}\right) \neq 1\right\}.$$

I changed this. If this was a classical approximate GCD problem, then the use of Sylvester-like matrices would be sufficient. However, in our problem the degrees of the entries of the adjoint may change under perturbations. In order to perform an analysis, we need to study a family of generalized Sylvester matrices that allow higher-degree zero coefficients to be perturbed.

The computation of the GCD of many polynomials is typically embedded into a scalar matrix problem using the classical Sylvester matrix. However, in our case we want to look at GCDs of nearby polynomials but with the added wrinkle that the degrees of the entries of the individual polynomials may change under perturbations. In order to perform such an analysis, we need to study a family of generalized Sylvester matrices that allow higher-degree zero coefficients to be perturbed.

Let $\mathbf{f} = (f_1, \ldots, f_k) \in \mathbb{R}[t]^k$ be a vector of polynomials with degrees $\mathbf{d} = (d_1, \ldots, d_k)$ ordered as $d_j \geq d_{j+1}$ fixed the order. We wanted the mto be decreasing. oops. for $1 \leq j \leq k-1$. Set $d = d_1$ and $\ell = \max(d_2, \ldots, d_k)$ and suppose that for each $i \in \{2, \ldots, k\}$ we have $f_i = \sum_{1\leq j\leq \ell} f_{ij}t^j$.

Definition 3.1 (Generalized Sylvester Matrix)**.** *The* generalized Sylvester matrix *of* $\mathbf{f}$ *is defined as*

$$\mathrm{Syl}(\mathbf{f}) = \mathrm{Syl}_{\mathbf{d}}(\mathbf{f}) = \begin{pmatrix} \phi_{\ell-1}(f_1)^T \\ \phi_{d-1}(f_2)^T \\ \vdots \\ \phi_{d-1}(f_k)^T \end{pmatrix} \in \mathbb{R}^{(\ell+(k-1)d)\times(\ell+d)}.$$

Some authors, e.g., (Fatouros and Karcanias, 2003; Vardulakis and Stoyle, 1978), refer to such a matrix as an expanded Sylvester matrix or generalized resultant matrix. The generalized Sylvester matrix has many useful properties pertaining to the Bézout coefficients. However, we are only concerned with the well known result that $\gcd(\mathbf{f}) = \gcd(f_1, \ldots, f_k) = 1$ if and only if $\mathrm{Syl}_{\mathbf{d}}(\mathbf{f})$ has full rank (Vardulakis and Stoyle, 1978).

Sometimes treatreating a polynomial of degree d as one of larger degree is useful. This can be accomplished by constructing a similar matrix and padding rows and columns with zero entries. The generalized Sylvester matrix of degree at most $\mathbf{d}' \geq \mathbf{d}$ (component-wise) of $\mathbf{f}$ is defined analogously as $\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f})$, taking d to be the largest degree entry and ℓ to be the largest degree of the remaining entries of $\mathbf{d}'$. Note that $\ell = d$ is possible and typical. If the entries of $\mathbf{f}$ have a non-trivial GCD (that is possibly unattainable) under a perturbation structure $\Delta\mathbf{f}$, then it is necessary that $\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f})$ is rank deficient, and often this will be sufficient.

If we view the entries of $\mathbf{f}$ as polynomials of degree $\mathbf{d}'$ and $d_i' > d_i$ for all i, then the entries of $\mathbf{f}$ have an unattainable GCD of distance zero, typically of the form $1+\varepsilon t \sim t+\varepsilon^{-1}$. In other words,

the underlying approximate GCD problem is ill-posed in a sense that the solution is unattainable. In order to study the theory of unattainable GCD's, sometimes referred to as GCD's at infinity, we need to study the notion of a degree reversed polynomial.

Lemma 3.2. *If* $\max(\mathbf{d}) = \max(\mathbf{d}')$ *then* $\mathrm{Syl}_{\mathbf{d}}(\mathbf{f})$ *has full rank if and only if and* $\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f})$ *has full rank.*

Proof. Let d and ℓ be the largest and second largest entries of $\mathbf{d}$ and ℓ' be the second largest entry of $\mathbf{d}'$. The result follows from the main theorem of Vardulakis and Stoyle (1978) by considering the case of $\ell' = d$. □

This lemma characterizes the (generic) case when elements of maximal degree of $\mathbf{f}$ do not change under perturbations, in which case the generalized Sylvester matrix still meaningfully encodes GCD information. However, it is possible that $\mathrm{Syl}_{\mathbf{d}}(\mathbf{f})$ has full rank and $\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f})$ is rank deficient but the distance to a non-trivial GCD is not zero. This can occur when $d_j = d'_j$ for some j and $\mathbf{d}' \geq \mathbf{d}$. To understand the most general case, we need to look at generalized Sylvester matrices of the reversal of several polynomials.

Definition 3.3. *The* degree d reversal *of* $f \in \mathbb{R}[t]$ *of degree at most* d *is defined as* $\mathrm{rev}_d(f) = t^d f(t^{-1})$. *For a vector of polynomials* $\mathbf{f} \in \mathbb{R}[t]^k$ *of degrees at most* $\mathbf{d} = (d_1, \ldots, d_k)$ *the* degree $\mathbf{d}$ reversal *of* $\mathbf{f}$ *is the vector* $\mathrm{rev}_{\mathbf{d}}(\mathbf{f}) = (\mathrm{rev}_{d_1}(f_1), \ldots, \mathrm{rev}_{d_k}(f_k))$.

The following theorem enables us to determine if unattainable solutions are occurring in an approximate GCD problem with an arbitrary (possibly non-linear) structure on the coefficients.

Theorem 3.4. *Let* $\mathbf{f}$ *be a vector of non-zero polynomials of degree at most* d. *Suppose that* $\mathrm{Syl}_{\mathbf{d}}(\mathbf{f})$ *has full rank and* $\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f})$ *is rank deficient, where the perturbations* $\Delta\mathbf{f}$ *have degrees at most* $\mathbf{d}'$ *and the entries of* $\mathbf{f}$ *have degrees* $\mathbf{d}$. *Then* $\mathbf{f}$ *has an* unattainable *non-trivial GCD of distance zero under the perturbation structure* $\Delta\mathbf{f}$ *if and only if* $\mathrm{Syl}(\mathrm{rev}_{\mathbf{d}'}(\mathbf{f}))$ *is rank deficient.*

Proof. Suppose that $\mathrm{Syl}(\mathrm{rev}_{\mathbf{d}'}(\mathbf{f}))$ has full rank. Then $\gcd(\mathrm{rev}_{\mathbf{d}'}(\mathbf{f})) = 1$, hence $\mathbf{f}$ does not have an unattainable non-trivial GCD, since $\gcd(\mathbf{f}) = 1$. Conversely, suppose that $\mathrm{Syl}(\mathrm{rev}_{\mathbf{d}'}(\mathbf{f}))$ is rank deficient. Then, t is a factor of $\gcd(\mathrm{rev}_{\mathbf{d}'}(\mathbf{f}))$ but t is not a factor of $\gcd(\mathrm{rev}_{\mathbf{d}}(\mathbf{f}))$. Accordingly, all entries of $\mathbf{f} + \Delta\mathbf{f}$ may increase in degree and so the distance of $\mathbf{f}$ having a non-trivial GCD is zero, and so is unattainable. □

If the generalized Sylvester matrix of $\mathbf{f}$ has full rank, but the generalized Sylvester matrix that encodes the perturbations $\mathbf{f} + \Delta\mathbf{f}$ is rank deficient, then either there is an unattainable GCD, or the generalized Sylvester matrix is rank deficient due to over-padding with zeros. Theorem 3.4 provides a reliable way to detect this over-padding.

We need to say how we handle structured perturbations

Definition 3.5. *We say that* $\mathcal{A}$ *has an* unattainable non-trivial Smith form *if* $\gcd(\mathrm{Adj}(\mathcal{A})) = 1$ *and* $\gcd(\mathrm{Adj}(\mathcal{A} + \widetilde{\Delta}\mathcal{A})) \neq 1$ *for an arbitrarily small perturbation* $\widetilde{\Delta}\mathcal{A} = \widetilde{\Delta}(\Delta\mathcal{A})$ *of some prescribed affine structure.*

Note that $\widetilde{\Delta}\mathcal{A}$ just means that perturbations to $\mathcal{A}$ are structured as an affine function of $\Delta\mathcal{A}$. I changed this shit or something. It is important to carefully consider structured perturbations, because some matrix polynomials have an unattainable non-trivial SNF under unstructured perturbations, but have an attainable non-trivial SNF under structured perturbations (perturbations

that preserve the degree of entries or support of entries are structured). Solutions that cannot be attained correspond to an eigenvalue at infinity of $\mathcal{A}$ with a non-trivial spectral structure. Such examples are easily constructed when $\det(\mathcal{A})$ or $\mathrm{Adj}(\mathcal{A})$ have non-generic degrees.

Example 3.6. *Let*

$$\mathcal{A} = \begin{pmatrix} t & t-1 \\ t+1 & t \end{pmatrix} \in \mathbb{R}[t]^{2\times 2} \text{ and } \mathcal{C} = \begin{pmatrix} \mathcal{A} & \\ & \mathcal{A} \end{pmatrix} \in \mathbb{R}[t]^{4\times 4}.$$

Then $\mathcal{C}$ has an unattainable non-trivial Smith form if all perturbations to $\mathcal{A}$ are support or degree preserving (i.e. they preserve zero entries or do not increase the degree of each entry), both linear structures. Note that $\mathcal{A}$ and $\mathcal{C}$ are both unimodular. However small perturbations to the non-zero coefficients of $\mathcal{A}$ make $\mathcal{A} + \Delta\mathcal{A}$ non-unimodular.

The Smith form of $\mathrm{rev}(\mathcal{C}) = t\mathcal{C}|_{t=t^{-1}}$ *is*

$$\mathrm{SNF}(\mathrm{rev}(\mathcal{C})) = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & t^2 & \\ & & & t^2 \end{pmatrix},$$

which implies that the eigenvalue at infinity of $\mathcal{A}$ has a non-trivial spectral structure. The eigenvalue at infinity having a non-trivial spectral structure implies that the SNF of $\mathcal{C}$ is unattainable. Note that this is equivalent to saying that $\mathcal{C}$ has a non-trivial Smith-McMillan form.

These examples are non-generic. Generically, the degree of all entries in the adjoint will be $(n-1)d$ and will remain unchanged locally under perturbations to the coefficients. Computing the distance to the nearest matrix polynomial with a non-trivial Smith form under a prescribed perturbation structure can be formulated as finding the nearest rank deficient (structured) generalized Sylvester matrix of the adjoint or the $\mathbf{d}'$ reversal of the adjoint.

3.2. Nearest Rank Deficient Structured Generalized Sylvester Matrix

Suppose that $\mathcal{A} \in \mathbb{R}[t]^{n\times n}$ of degree at most d has a trivial Smith form and does not have an unattainable non-trivial Smith form. Then one method to compute a lower bound on the distance the entries of $\mathcal{A}$ need to be perturbed to have an attainable or unattainable non-trivial Smith form is to solve

$$\inf \|\Delta\mathcal{A}\| \text{ subject to } \begin{cases} \mathrm{rank}(\mathrm{Syl}_{\mathbf{d}'}(\mathrm{Adj}(\mathcal{A} + \widetilde{\Delta}\mathcal{A}))) < e, \\ e = \mathrm{rank}(\mathrm{Syl}_{\mathbf{d}'}(\mathrm{Adj}(\mathcal{A}))). \end{cases} \tag{6}$$

Here $\mathbf{d}'$ is the vector of the largest possible degrees of each entry of $\mathrm{Adj}(\mathcal{A} + \widetilde{\Delta}\mathcal{A})$, and $\widetilde{\Delta}\mathcal{A}$ is a prescribed linear or affine perturbation structure. I made a change here... typo was fixed as well.

It is sufficient to compute $\max(\mathbf{d}')$, a quantity which will generically be $(n-1)d$. For non-generic instances we require the computation of $\mathbf{d}'$. This optimization problem is non-convex, but multi-linear in each coefficient of $\Delta\mathcal{A}$.

We do not attempt to solve this problem directly via numerical techniques, since it enforces a necessary condition that is often sufficient. Instead we use it to develop a theory of solutions which can be exploited by faster and more robust numerical methods.

Lemma 3.7. *Let* $\mathbf{f}$ *be a vector of polynomials with degrees* $\mathbf{d}$ *and admissible perturbations* $\Delta\mathbf{f}$ *of degrees* $\mathbf{d}'$ *where* $\max(\mathbf{d}) \leq \max(\mathbf{d}')$. *Then the family of generalized Sylvester matrices* $\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f})$ *of rank at least e form an open set under the perturbations* $\Delta\mathbf{f}$.

Proof. By the degree assumption on $\Delta\mathbf{f}$ we have that for an infinitesimal $\Delta\mathbf{f}$ that $\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f})$ and $\mathrm{Syl}_{\mathbf{d}'}(\Delta\mathbf{f})$ have the same dimension. Accordingly, let us suppose that $\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f})$ has rank at least e. Then the Sylvester matrix in question must have rank at least e in an open-neighborhood around it. In particular, when $\|\mathrm{Syl}_{\mathbf{d}'}(\Delta\mathbf{f})\|_2 < \sigma_e(\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f}))$ then $\mathrm{rank}(\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f}+\Delta\mathbf{f})) \geq \mathrm{rank}(\mathrm{Syl}_{\mathbf{d}'}(\mathbf{f}))$ and the result follows. □

Theorem 3.8. *The optimization problem (6) has an attainable global minimum under linear perturbation structures.*

Proof. Let $\mathcal{S}$ be the set of all rank at most $e-1$ generalized Sylvester matrices of prescribed shape by $\mathbf{d}'$ and $\mathrm{Adj}(\mathcal{A})$. Lemma 3.7 implies that $\mathcal{S}$ is topologically closed.

Let $\mathcal{R} = \{\mathrm{Syl}_{\mathbf{d}'}(\mathrm{Adj}(\mathcal{C})) \text{ subject to } \|\mathcal{C}\| \leq \|\mathcal{A}\|\}$, where the generalized Sylvester matrices are padded with zeros to have the appropriate dimension if required. Since $\Delta\mathcal{A}$ has a linear perturbation structure, a feasible point is always $C = -\mathcal{A}$. By inspection $\mathcal{R}$ is seen to be a non-empty set that is bounded and closed.

The functional $\|\cdot\|$ is continuous over the non-empty closed and bounded set $\mathcal{S}\cap\mathcal{R}$. Let $\mathcal{B} \in \mathcal{S}\cap\mathcal{R}$. By Weierstrass's theorem $\|\mathcal{A}-\mathcal{B}\|$ has an attainable global minimum over $\mathcal{S}\cap\mathcal{R}$. □

Note that if a feasible point exists under an affine perturbation structure, then a solution to the optimization problem exists as well. What this result says is that computing the distance to non-triviality is generally a well-posed problem, even though computing a matrix polynomial of minimum distance may be ill-posed (the solution is unattainable). The same results also hold when working over the $\mathbf{d}'$ reversed coefficients. A similar argument is employed by (Kaltofen et al., 2007, Theorem 2.1).

3.3. Bounds on the Distance to non-triviality

Suppose that $\mathcal{A} \in \mathbb{R}[t]^{n\times n}$, of degree at most d, has a trivial Smith form and does not have an unattainable non-trivial Smith form. This section provides some basic bounds on the distance coefficients of $\mathcal{A}$ need to be perturbed to have a non-trivial Smith form. The bounds we derive are unstructured, although they can be generalized to several perturbation structures (such as ones that preserve the degree or support of entries) in a straight forward manner.

If we consider the mapping $\mathrm{Adj}(\cdot)$ as a vector-valued function from $\mathbb{R}^{n^2(d+1)} \to \mathbb{R}^{n^2((n-1)d+1)}$ (with some coordinates possibly fixed to zero), then we note that the mapping is locally Lipschitz. More precisely, there exists $c > 0$ such that for a sufficiently small $\Delta\mathcal{A}$,

$$\|\mathrm{Adj}(\mathcal{A}) - \mathrm{Adj}(\mathcal{A}+\Delta\mathcal{A})\| \leq c\|\Delta\mathcal{A}\|.$$

The quantity c can be approximately bounded above by the (scalar) Jacobian matrix $\nabla\,\mathrm{Adj}(\cdot)$ evaluated at $\mathcal{A}$. A local upper bound for c is approximately $\|\nabla\,\mathrm{Adj}(\mathcal{A})\|_2$. We can invoke Theorem 2.13 if $\mathcal{A}$ has full rank. By considering $\hat{c} = \left\|[\Phi_{(n-1)d}(I\otimes\mathcal{A})]^+\right\|_2$, we obtain the (absolute) first-order *approximate* perturbation bound

$$\|\mathrm{Adj}(\mathcal{A}) - \mathrm{Adj}(\mathcal{A}+\Delta\mathcal{A})\|_F \lesssim \hat{c}(n+\sqrt{n})(d+1)\|\mathrm{Adj}(\mathcal{A})\|_F\|\Delta\mathcal{A}\|_F.$$

The entries of $\nabla\,\mathrm{Adj}(\mathcal{A})$ consist of the coefficients of the $(n-2)\times(n-2)$ minors of $\mathcal{A}$. This follows because $\mathrm{Adj}(\cdot)$ is a multi-linear vector mapping and the derivative of each entry is a coefficient of the leading coefficient with respect to the variable of differentiation. The size of

each minor can be bounded above (albeit poorly) by Hadamard's inequality (Goldstein-Graham variant, see (Lossers, 1974)). As such, we have the sequence of bounds

$$\|\nabla \operatorname{Adj}(\mathcal{A})\|_2 \;\le\; n\sqrt{d+1}\|\nabla \operatorname{Adj}(\mathcal{A})\|_\infty \;\le\; n^3(d+1)^{5/2}\|\mathcal{A}\|_\infty^{n-2}(d+1)^{n-2}n^{(n-2)/2},$$

where $\|\mathcal{A}\|_\infty$ is understood to be a vector norm and $\|\nabla \operatorname{Adj}(\mathcal{A})\|_\infty$ is understood to be a matrix norm. The bound in question can be used in conjunction with the SVD to obtain a lower bound on the distance to a matrix polynomial with a non-trivial Smith form.

Theorem 3.9. *Suppose that* $\mathbf{d}' = (\gamma, \gamma \ldots, \gamma)$ *and* $\operatorname{Syl}_{\mathbf{d}'}(\operatorname{Adj}(\mathcal{A}))$ *has rank e. Then an approximate lower bound on the distance to non-triviality is*

$$\frac{1}{\gamma\|\nabla \operatorname{Adj}(\mathcal{A})\|_F}\sigma_e(\operatorname{Syl}_{\mathbf{d}'}(\operatorname{Adj}(\mathcal{A}))).$$

Proof. We note that for polynomials $\mathbf{f}$ with degrees $\mathbf{d}'$ that $\|\operatorname{Syl}_{\mathbf{d}'}(\mathbf{f})\|_F = \gamma\|\mathbf{f}\|_F$. Accordingly, if $\Delta\mathcal{A}$ is a minimal perturbation to non-triviality, then

$$\begin{aligned}\frac{1}{\gamma}\sigma_e(\operatorname{Syl}_{\mathbf{d}'}(\operatorname{Adj}(\mathcal{A}))) &\le \|\operatorname{Adj}(\mathcal{A}) - \operatorname{Adj}(\mathcal{A}+\Delta\mathcal{A})\|_F \\ &\lesssim \|\nabla \operatorname{Adj}(\mathcal{A})\|_F\|\Delta\mathcal{A}\|_F,\end{aligned}$$

and the theorem follows by a simple rearrangement. Note that $\|\cdot\|_2 \le \|\cdot\|_F$. □

If $\mathbf{d}'$ has different entries, then $\ell\|\mathbf{f}\|_F \le \|\operatorname{Syl}_{\mathbf{d}'}(\mathbf{f})\|_F \le \gamma\|\mathbf{f}\|_F$, where γ and ℓ are the largest and second-largest entries of $\mathbf{d}'$. The lower bound provided can also be improved using the Karmarkar-Lakshman distance (Karmarkar and Lakshman, 1996) in lieu of the smallest singular value of the generalized Sylvester matrix, the $\mathbf{d}'$ reversal of the adjoint or other approximate GCD lower bounds (e.g., (Beckermann and Labahn, 1998)).

4. Approximate SNF via Optimization

In this section we formulate the approximate Smith form problem as the solution to a continuous constrained optimization problem. We assume that the solutions in question are attainable and develop a method with rapid local convergence. As the problem is non-convex, our convergence analysis will be local.

4.1. Constrained Optimization Formulation

An equivalent statement to $\mathcal{A}$ having a non-trivial attainable Smith form is that $\operatorname{Adj}(\mathcal{A}) = \mathcal{F}^* h$ where $\mathcal{F}^*$ is a vector (or matrix) of scalar polynomials and h is a divisor of $\gcd(\operatorname{Adj}(\mathcal{A}))$. This directly leads to the following optimization problem:

$$\min \|\Delta\mathcal{A}\|_F^2 \;\text{ subject to }\; \begin{cases}\operatorname{Adj}(\mathcal{A}+\Delta\mathcal{A}) = \mathcal{F}^* h, & \mathcal{F}^* \in \mathbb{R}[t]^{n\times n}, h \in \mathbb{R}[t], \\ \mathcal{N}_h \operatorname{vec}(h) = 1, & \mathcal{N}_h \in \mathbb{R}^{1\times(\deg(h)+1)}.\end{cases} \tag{7}$$

This is a multi-linearly structured approximate GCD problem which is a non-convex optimization problem. Instead of finding a rank deficient Sylvester matrix, we directly enforce that the entries of $\operatorname{Adj}(\mathcal{A})$ have a non-trivial GCD. The normalization requirement that $\mathcal{N}_h \operatorname{vec}(h) = 1$ is chosen

to force h to have a non-zero degree, so that h is not a scalar. One useful normalization is to define η_h such that lcoeff$(h) = 1$ (that is lcoeff$(\cdot)$ is the leading coefficient of a polynomial). Explicitly, we assume the degree of the approximate GCD is known and make it monic. Of course, other valid normalizations also exist.

Since we are working over $\mathbb{R}[t]$, there will always be a quadratic, linear or zero factor of attainable solutions. If $h = 0$ then the approximate SNF of $\mathcal{A}$ is rank deficient and computing approximate SNF reduces to the nearest rank at-most $n-1$ or $n-2$ matrix polynomial problems, both of which are well-understood (Giesbrecht, Haraldson, and Labahn, 2017b,c). Assuming that we are now working in the nonzero case, we can assume generically that $\deg(h) = 1$ or $\deg(h) = 2$.

4.2. Lagrange Multipliers and Optimality Conditions

In order to solve our problem we will employ the method of Lagrange multipliers. The Lagrangian is defined as

$$L = \|\Delta\mathcal{A}\|_F^2 + \lambda^T \begin{pmatrix} \text{vec}(\text{Adj}(\mathcal{A} + \Delta\mathcal{A}) - \mathcal{F}^* h) \\ \eta_h \text{vec}(h) - 1 \end{pmatrix},$$

where λ is a vector of Lagrange multipliers.

A necessary first-order condition (KKT condition, e.g. (Bertsekas, 1999)) for a tuple $z^\star = z^\star(\Delta\mathcal{A}, \mathcal{F}^*, h, \lambda)$ to be a regular (attainable) minimizer is that the gradient of L vanishes, that is,

$$\nabla L(z^\star) = 0. \tag{8}$$

Let J be the Jacobian matrix of the constraints defined as

$$J = \nabla_{\Delta\mathcal{A}, \mathcal{F}^*, h} \left(\text{vec}(\text{Adj}(\mathcal{A} + \Delta\mathcal{A}) - \mathcal{F}^* h) \right).$$

The second-order sufficiency condition for optimality at a local minimizer $z^\star$ is that

$$\ker(J(z^\star))^T \nabla^2_{xx} L(z^\star) \ker(J(z^\star)) \succ 0, \tag{9}$$

that is, the Hessian with respect to $x = x(\Delta\mathcal{A}, \mathcal{F}^*, h)$ is positive definite over the kernel of the Jacobian of the constraints. The vector x corresponds to the variables in the affine structure of $\Delta\mathcal{A}$,$\mathcal{F}^*$, and h. If (8) and (9) both hold, then $z^\star$ is necessarily a local minimizer of (7). Of course, it is also necessary that $\ker(J(z^\star))^T \nabla^2_{xx} L(z^\star) \ker(J(z^\star)) \succeq 0$ at a minimizer, which is the second-order necessary condition. Our strategy for computing a local solution is to solve $\nabla L = 0$ using a Newton-like method.

4.3. An Implementation with Local Quadratic Convergence

A problem with Newton-like methods is that when the Hessian is rank deficient or ill-conditioned, then the Newton step becomes ill-defined or the rate of convergence degrades. The proposed formulation of our problem can encounter a rank deficient Hessian (this is due to over padding some vectors with zero entries or redundant constraints). Despite this we are still able to obtain a method with rapid local convergence under a very weak normalization assumption.

In order to obtain rapid convergence we make use of the Levenberg-Marquart (LM) algorithm. If $H = \nabla^2 L$, then the LM iteration is defined as repeatedly solving for $z^{(k+1)} = z^{(k)} + \Delta z^{(k)}$ by

$$(H^T H + \nu_k I)\Delta z^{(k)} = -H^T \nabla L(z^{(k)}) \text{ where } z = \begin{pmatrix} x \\ \lambda \end{pmatrix} \in \mathbb{R}^\ell,$$

for some $\ell > 0$ while using $\|\nabla L\|_2$ as a merit function. The speed of convergence depends on the choice of $\nu_k > 0$. Note that since LM is essentially a regularized Gauss-Newton method, when the Hessian is rank deficient then we may converge to a stationary point of the merit function. If convergence to a stationary point of the merit function is detected, then the method of Wright (2005) can be used to replace LM in several instances.

Yamashita and Fukushima (2001) show that, under a local-error bound condition, a system of non-linear equations $g(z) = 0$ approximated by LM will converge quadratically to a solution with a suitable initial guess. Essentially, what this says is that to obtain rapid convergence it is sufficient for regularity (J having full rank) to hold or second-order sufficiency, but it is not necessary to satisfy both. Note that we assume Lagrange multipliers exist. However, unlike the case when J has full rank, the multipliers need not be unique. The advantage of LM over other Newton-like methods is that this method is globalized[1] in exchange for an extra matrix multiplication, as $H^T H + \nu_k I$ is always positive definite, and hence always a descent direction for the merit function. We make the choice of $\nu_k \approx \|g(z)\|_2$ based on the results of Fan and Yuan (2005).

Definition 4.1 (Local Error Bound)**.** *Let $Z^\star$ be the set of all solutions to $g(z) = 0$ and X be a subset of $\mathbb{R}^\ell$ such that $X \cap Z^\star \neq \emptyset$. We say that $\|g(z)\|$ provides a local error bound on $g(z) = 0$ if there exists a positive constant c such that $c \cdot \operatorname{dist}(z, Z^\star) \leq \|g(z)\|$ for all $z \in X$, where $\operatorname{dist}(\cdot)$ is the distance between a point and a set.*

In this section it is useful to consider $g(z) = \nabla L(z)$, as we need the local error bounds to estimate $\nabla L(z) = 0$.

Theorem 4.2. *If the second-order sufficiency condition (9) holds at an attainable solution to (7), then the local error-bound property holds.*

Proof. This result follows immediately from Section 3 of Wright (2005) and the references therein. □

The bounds of Wright can be used to infer when quadratic convergence occurs for Newton-like methods. In this problem, perturbations to x are important in understanding how the problem behaves locally.

Remark 4.3. *Let $z = z(x, \lambda)$ where x is a vector of variables and λ is a vector of Lagrange multipliers, and define $g(z) = \nabla L(z)$. First suppose that both the second-order sufficiency condition (9) and first-order necessary condition (8) hold at the point $z^\star$. We can write the first-order expansion*

$$g(z^\star + \Delta z) = H(z^\star)(\Delta z) + O(\|\Delta z\|_2^2) \approx H(z^\star)(\Delta z),$$

noting that $g(z^\star) = 0$. It is useful to observe that

$$H(z^\star) = \begin{pmatrix} H_{xx}(z^\star) & J^T(z^\star) \\ J(z^\star) & 0 \end{pmatrix}.$$

[1] Here "globalized" means that the method will converge to a stationary point of the merit function, not a local extremum of the problem.

If $\Delta x = 0$ then the error-bound from Hoffman (1952) (main theorem) applies and we have that there exists $c_{hof} > 0$ such that $c_{hof}\|\Delta\lambda\| \leq \|g(x, \lambda + \Delta\lambda)\|$. If $\Delta x \neq 0$ then $\left\|\begin{pmatrix} H_{xx}(z^\star) \\ J(z^\star) \end{pmatrix}\Delta x\right\| \approx \|g(x + \Delta x, \lambda)\|$ and (9) implies that $H(z^\star)(\Delta z) = 0 \implies \Delta x = 0$, so

$$\sigma_{\min}\begin{pmatrix} H_{xx}(z^\star) \\ J(z^\star) \end{pmatrix}\|\Delta x\| \lesssim \|g(x + \Delta x, \lambda)\|,$$

so there exists $c_{\sigma_{\min}} > 0$ when $\|\Delta x\|$ is sufficiently small such that $c_{\sigma_{\min}}\|\Delta x\| \leq \|g(x + \Delta x, \lambda)\|$. Note that $c_{\sigma_{\min}} \approx \sigma_{\min}\begin{pmatrix} H_{xx}(z^\star) \\ J(z^\star) \end{pmatrix}$.

The first-order approximation implies that when $\|\Delta z\|$ is sufficiently small that

$$g(z^\star + \Delta z) \approx H(z^\star)(\Delta z) = H(z^\star)\begin{pmatrix} \Delta x \\ 0 \end{pmatrix} + H(z^\star)\begin{pmatrix} 0 \\ \Delta\lambda \end{pmatrix} \approx g(x + \Delta x, \lambda) + g(x, \lambda + \Delta\lambda).$$

The key idea is to separate the problem into the cases of $\Delta x = 0$ and $\Delta x \neq 0$, and then derive error bounds for each case. The important part of the discussion is that if one can estimate $c_{\sigma_{\min}}$ then one can often infer when quadratic convergence occurs.

The second-order sufficiency assumption is not necessary to derive error bounds bounds. It is straightforward to show the local error bound property holds if $J(z^\star)$ has full rank, as the Lagrange multipliers will be (locally) unique, hence the solution is (locally) unique. Alternatively, if J had constant rank in a non-trivial open neighborhood around a solution, then a similar argument could be made about the local error-bound property.

Theorem 4.4. *The second-order sufficiency condition holds at minimal solutions with Lagrange multipliers of minimal norm if h is of maximal degree and monic and the minimal structured perturbation $\|\Delta\mathcal{A}^\star\|$ is sufficiently small.*

Proof. The Hessian of L with respect to $x = x(\Delta A, \mathcal{F}^*, h)$ is

$$\nabla^2_{xx}L = H_{xx} = \begin{pmatrix} F + 2I & & \\ & & E \\ & E^T & \end{pmatrix},$$

where F is a square matrix with zero diagonal whose entries are a multi-linear polynomial in λ and $\Delta\mathcal{A}$ and E^T is a matrix whose entries are homogeneous linear functions in λ.

If $\Delta\mathcal{A}^\star = 0$ then $\lambda^\star = 0$. Hence both $E = 0$ and $F = 0$ and so, if $y \in \ker(H_{xx}) \cap \ker(J)$ then $y = \begin{pmatrix} 0 & y_2 & y_3 \end{pmatrix}^T$. The Jacobian of the constraints may be written (up to permutation) as

$$J = \begin{pmatrix} * & C_h & C_{\mathcal{F}^*} \\ & & \mathcal{N}_h \end{pmatrix},$$

where $*$ are blocks corresponding to differentiating with respect to variables in $\Delta\mathcal{A}$ and the blocks $C_{\mathcal{F}^*}$ and C_h consist of block convolution and convolution matrices that correspond to multiplication by $\mathcal{F}^*$ and h, respectively. The block $\mathcal{N}_h$ contains a normalization vector to ensure that h has the appropriate degree. $Jy = 0$ implies that there exists a vector of polynomials v and a polynomial u with the same degrees as $\mathcal{F}^*$ and h such that $\mathcal{F}^*u + vh = 0$ and $\mathcal{N}_h\text{vec}(u) = 0$.

We have that h is a factor of both $\mathcal{F}^*u$ and vh. Since $\gcd(\mathcal{F}^*, h) = 1$ it must be that h is a factor of u. It follows that $\deg(u) = \deg(h)$, so there exists some $\alpha \neq 0$ such that $\alpha u = h$. Since h

is monic, we have that $\mathcal{N}_h \text{vec}(h) = 1$ but $\mathcal{N}_h \text{vec}(u) = 0$, which implies that $\alpha = 0$, and so $u = 0$. We have that $vh = 0$ and this implies $v = 0$. Hence $\ker(J) \cap \ker(H_{xx}) = 0$ and second-order sufficiency holds when $\|\Delta\mathcal{A}^*\| = 0$.

If $\|\Delta\mathcal{A}^*\|$ is sufficiently small, then $\|F\|$ will be sufficiently small so that $F + 2I$ has full rank. Accordingly, we have that

$$\ker \begin{pmatrix} F + 2I & & \\ & 0 & E \\ & E^T & 0 \end{pmatrix} \subseteq \ker \begin{pmatrix} 2I & & \\ & 0 & \\ & & 0 \end{pmatrix}. \qquad \square$$

We remark that the techniques in the proof are very similar to those of Zeng and Dayton (2004) and Giesbrecht, Haraldson, and Kaltofen (2017a) to show that a Jacobian matrix appearing in approximate GCD computations of two (or more) polynomials has full rank. If we over-estimated the degrees of $\mathcal{F}^*$ then H_{xx} would have some columns and rows consisting of zero (the block-convolution matrices would be padded with extra zero entries).

In the proof of Theorem 4.4 we note that

$$\nabla^2_{xx} L = \nabla^2_{xx} \|\Delta\mathcal{A}\|_F^2 + \nabla_x \lambda^T J.$$

The matrix $F = \nabla_{\Delta\mathcal{A}} \lambda^T J_{\text{Adj}}(\mathcal{A} + \Delta\mathcal{A})$ will consist of coefficients of the $(n-3) \times (n-3)$ minors of $\mathcal{A} + \Delta\mathcal{A}$ scaled by entries of λ. Accordingly, F will generally not have -2 as an eigenvalue.

Remark 4.5. *Thus far we have assumed that Lagrange multipliers exist at the current solutions of interest, which are attainable solutions that have full rank. Corollary 2.14 and the proof of Theorem 4.4 imply that Lagrange multipliers generally exist under these assumptions for several perturbation structures, since we need to solve*

$$\begin{pmatrix} 2\text{vec}(\Delta\mathcal{A})^T & 0 \end{pmatrix} = -\lambda^T J,$$

of which J generally has constant or full rank. Of course if the solution was unattainable then the GCD constraints would break down as there is a "solution at infinity" in a sense that $\|h\| \to \infty$ as $\Delta\mathcal{A} \to \Delta\mathcal{A}^\star$.

The implication of the local-error bound property holding is that one can reasonably approximate when quadratic convergence occurs by estimating $\sigma_{\min}\left(\begin{bmatrix} H_{xx} \mid J^T \end{bmatrix}\right)$ and c_{hof}. In particular, these quantities act as a structured condition number on the system. A structured backwards-error analysis of existing techniques can be performed using these quantities. Additionally, it is somewhat generic that $F + 2I$ has full rank, hence the local error-bound will hold for most instances of the approximate SNF problem with an attainable solution. It is also important to note that we did not explicitly use the adjoint matrix. Indeed the result remains valid if we replace the adjoint with minors of prescribed dimension. Likewise, if $\mathcal{A}$ is an ill-posed instance of lower McCoy rank or approximate SNF without an attainable global minimum, then optimizing over a reversal of each entry of $\text{Adj}(\mathcal{A} + \Delta\mathcal{A})$ would yield a non-trivial answer and the same stability properties would hold. Thus, poorly posed problems also remain poorly posed if slightly perturbed.

Corollary 4.6. *The LM algorithm for solving $\nabla L = 0$ has quadratic convergence under the assumptions of Theorem 4.2 and using $\nu_k = \|\nabla(L(z^k))\|_2$.*

Proof. The quantity ∇L is a multivariate polynomial, hence it is locally Lipschitz. Second-order sufficiency holds, thus we have the local error bound property is satisfied. The method converges rapidly with a suitable initial guess. $\square$

Note that for several perturbation structures if the adjoint has generic degrees, then the Jacobian of the constraints will have full rank, and a standard Newton iteration is also well-defined, and will converge quadratically as well.

In the next section we discuss a technique that possibly forgoes rapid local convergence, but has a polynomial per iteration cost to compute a low McCoy rank approximation.

4.4. Computational Challenges and Initial Guesses

The most glaring problem in deriving a fast iterative algorithm for the approximate Smith form problem is that the matrix $\mathrm{Adj}(\mathcal{A} + \Delta\mathcal{A})$ has exponentially many coefficients as a multivariate polynomial in $\Delta\mathcal{A}$. This means computing the adjoint matrix symbolically as an ansatz is not feasible. In order to solve (8) we instead approximate the derivatives of the coefficients of the adjoint numerically.

To compute an initial guess, we can use $\Delta\mathcal{A}_{init} = 0$ and take $\mathcal{F}^*$ and h to be a reasonable approximation to an approximate GCD of $\mathrm{Adj}(\mathcal{A})$, which will often be valid as per Theorem 4.2. To make sure the point is feasible, one can use a variant of Newton's method to project to a feasible point. Corollary 2.14 implies that with a suitable initial guess, reasonable variants of Newton's method (such as LM) will converge quadratically to a feasible point, assuming one exists.

Another technique is to take two rows or columns of $\mathcal{A}$ and perturb them so that the $2n$ entries have a non-trivial GCD. To find the best guess with this technique, $O(n^2)$ approximate GCD computations on $O(n)$ polynomials of degree d need to be performed. In the next section we will discuss more sophisticated techniques.

4.5. Attaining Unattainable Solutions

If a solution is unattainable then the degrees of all the entries of the adjoint matrix may change in an open neighborhood around a solution. If $\Delta\mathcal{A}^\star$ is an unattainable solution (of full rank) to (7) then $h(t) = t$ is clearly not a solution since $h(t) = t$ being a solution implies that such a solution would be attainable. Let d_{Adj} be the generic degree of $\mathrm{Adj}(\mathcal{A} + \Delta\mathcal{A})$, then t is a factor of $\gcd(\mathrm{rev}_{d_{\mathrm{Adj}}}(\mathrm{Adj}(\mathcal{A} + \Delta\mathcal{A}^\star)))$. The reversed adjoint has no GCD at infinity by assumption, as such a GCD at infinity would be an attainable solution to the original problem. Accordingly, we note that Theorem 4.4 applies after some straightforward modifications, since

$$\nabla\mathrm{vec}(\mathrm{Adj}(\mathcal{A} + \Delta\mathcal{A})) \quad \text{and} \quad \nabla\mathrm{vec}(\mathrm{rev}_{d_{\mathrm{Adj}}}(\mathrm{Adj}(\mathcal{A} + \Delta\mathcal{A})))$$

are essentially (block) permutations of each other.

Since $\mathrm{rev}_{d_{\mathrm{Adj}}}(\mathrm{Adj}(\mathcal{A} + \Delta\mathcal{A}))$ achieves the generic degree, Lagrange multipliers should exist as we can apply Corollary 2.14 on $\nabla\mathrm{vec}(\mathrm{rev}_{d_{\mathrm{Adj}}}(\mathrm{Adj}(\mathcal{A} + \Delta\mathcal{A})))$ by permuting entries, and the underlying approximate GCD problem is well-posed. Thus the problem will also typically admit Lagrange multipliers.

The essential ingredient in Theorem 4.4 is the normalization of the underlying approximate GCD problem. This means that "backwards stable" algorithms will compute the exact SNF of a nearby matrix polynomial that has no meaning in the context of computation. This generally occurs because the radius of uncertainty, usually proportional to unit rounding errors, contains infinitely many matrix polynomials with a non-trivial SNF. The backwards stability is not meaningful in this context, because the instance of the problem is not continuous. In such instances, computing the SNF is most likely the wrong how do I fix this? problem to be considering. Instead, computing the spectral structure of eigenvalues at infinity is most likely the appropriate

problem. However there exist instances where both problems could be simultaneously poorly conditioned.

If the reversed problem has a radius of stability with respect to Theorem 4.4, then the original problem has a radius of instability, meaning that the iterates will converge to a point where $\|h\|$ is excessively large. In other words, if an instance of a problem is ill-posed, then it cannot be regularized — the finite and infinite eigenvalues and their spectral structure is indistinguishable in floating point arithmetic — in the context of the QZ decomposition, GUPTRI (Demmel and Kågström, 1993a,b) or similar algorithms. There are some instances where attempting to compute the SNF numerically is not possible and should not be attempted. In the context of an optimization problem, we can of course regularize the problem as we have just described. Van Dooren (1979) suggests that ill-posed problems should be formulated as an optimization problem as a means of regularization to overcome some of the numerical difficulties.

5. Lower McCoy Rank Approximation

In this section we describe how to find a nearby matrix polynomial of lower McCoy. Another way to formulate $\mathcal{A}$ having a non-trivial SNF is to solve the minimization problem

$$\min \|\Delta\mathcal{A}\|_F^2 \quad \text{subject to} \quad \begin{array}{l}(\mathcal{A}(\omega)+\Delta\mathcal{A}(\omega))B = 0 \ \text{ and } \ B^*B = I_2,\\ \text{for some } \omega \in \mathbb{C} \text{ and } B \in \mathbb{C}^{n\times 2},\end{array} \tag{10}$$

where $\Delta\mathcal{A}$ must have the appropriate structure. Essentially this finds the smallest perturbation of $\mathcal{A}$ with an eigenvalue that lowers the rank by at least 2. The auxiliary variables ω and B are used to enforce this constraint. Here B^* is the conjugate transpose of B, and $B^*B = I_2$ ensures that the kernel vectors are linearly independent and do not tend towards zero.

The optimization is unstable if ω is reasonably large, since the largest terms appearing are proportional to $O((d+1)\|\mathcal{A}\|_\infty|\omega|^d)$. To remedy this, if we assume that a solution to the optimization problem (10) exists and has full rank, then we may transform $\mathcal{A}+\Delta\mathcal{A}$ into a degree-one matrix polynomial (also known as a matrix pencil) with the same spectral properties, known as a *linearization*. If there is no full-rank solution one can simply take a lower-rank approximation (Giesbrecht, Haraldson, and Labahn, 2017c) and extract a square matrix polynomial of full rank that may be linearized. Alternatively, one may forgo the linearization and work directly with a problem that is more poorly conditioned. For the rest of this section we will assume, without loss of generality, that $\mathcal{A}$ and the solutions to the low McCoy rank problem have full rank.

We can encode the spectral structure and SNF of $\mathcal{A}$ as the following degree-one matrix polynomial (sometimes referred to as the *companion linearization* (Gohberg et al., 2009)) of the form $\mathcal{P} \in \mathbb{R}[t]^{nd\times nd}$, defined as

$$\mathcal{P} = \begin{pmatrix} I & & \\ & \ddots & \\ & & A_d \end{pmatrix} t - \begin{pmatrix} & I & & \\ & & \ddots & \\ -A_0 & -A_1 & \cdots & -A_{d-1} \end{pmatrix}.$$

This particular linearization encodes the SNF of $\mathcal{A}$, as $\mathrm{SNF}(\mathcal{P}) = \mathrm{diag}(I, I, \ldots, I, \mathrm{SNF}(\mathcal{A}))$. It follows that $\mathcal{A}$ has a non-trivial SNF if and only if $\mathcal{P}$ has a non-trivial SNF. If we preserve the affine structure of $\mathcal{P}$ and only perturb blocks corresponding to $\mathcal{A}$, then the reduction to a pencil will be sufficient. Other linearizations are possible as well. The pencil is generally better behaved numerically since the largest entry upon evaluation at a $\omega \in \mathbb{C}$ is proportional to $O(d\|\mathcal{A}\|_\infty|\omega|)$ rather than $O(\|A\|_\infty|\omega|^d)$, albeit with matrices that are d times larger.

5.1. Fast Low McCoy Rank via Optimization

One way to approach the lower McCoy rank approximation problem is to study all the minors (or sufficiently many) of a matrix polynomial. This method immediately generalizes from the previous section, however is not practical for computational purposes since the number of minors grows exponentially in the dimension. Instead, we can approach the problem by formulating it as an optimization problem, one that is remarkably similar to structured lower rank approximation of scalar matrices. This similarity facilitates computing an initial guess for the following optimization problem using the SVD.

The lower McCoy rank approximation problem may be formulated as the following *real optimization problem*: to find the nearest matrix polynomial to $\mathcal{A} \in \mathbb{R}[t]^{n\times n}$ with McCoy rank $n-r$, find the perturbation $\Delta\mathcal{A} \in \mathbb{R}[t]^{n\times n}$ which minimizes

$$\min \|\Delta\mathcal{A}\|_F^2 \text{ subject to } \begin{cases} \mathfrak{R}((\mathcal{P}+\Delta\mathcal{P})(\omega)B) = 0, \\ \mathfrak{I}((\mathcal{P}+\Delta\mathcal{P})(\omega)B) = 0, \\ \mathfrak{R}(B^*B) = I_r, \\ \mathfrak{I}(B^*B) = 0 \end{cases} \quad \text{for some } \omega \in \mathbb{C} \text{ and } B \in \mathbb{C}^{nd\times r}. \tag{11}$$

Note that the perturbation $\Delta\mathcal{A}$ is *real valued* in this problem. The unitary constraint on B ensures that $\operatorname{rank}(B) = r$ and each column of B remains away from zero. Accordingly, $\omega \in \mathbb{C}$ will be an eigenvalue of $(\mathcal{P}+\Delta\mathcal{P})(\omega)$ since $\operatorname{rank}((\mathcal{P}+\Delta\mathcal{P})(\omega)) \leq nd - r$, and thus the McCoy rank of $\mathcal{A}+\Delta\mathcal{A}$ is at-most $n-r$.

Real matrix polynomials can have complex eigenvalues and so complex numbers must necessarily appear in the constraints. The constraints arising from the complex numbers may be divided into real parts and imaginary parts, denoted as $\mathfrak{R}(\cdot)$ and $\mathfrak{I}(\cdot)$, respectively. By dividing the constraint into real and imaginary parts, we are able to solve an equivalent optimization problem completely with real variables. This ensures that $\mathfrak{I}(\Delta\mathcal{A}) = 0$, that is, the perturbations are real. Since $\mathcal{A}+\Delta\mathcal{A}$ may have complex eigenvalues (but entries with real coefficients), we require that $\operatorname{SNF}(\mathcal{A}+\Delta\mathcal{A})$ has entries from $\mathbb{R}[t]$. Accordingly, we need to interpret the auxiliary variable ω. The instance of $\mathfrak{I}(\omega) = 0$ corresponds to $t-\omega$ as an invariant factor, while $\mathfrak{I}(\omega) \neq 0$ corresponds to the real irreducible quadratic $(t-\omega)(t-\overline{\omega})$. Thus at a solution, we are able to recover a real invariant factor regardless if ω has a non-zero imaginary part.

In order to approach the problem using the method of Lagrange multipliers we define the Lagrangian as

$$L = \|\Delta\mathcal{A}\|_F^2 + \lambda^T \operatorname{vec}\begin{pmatrix} \mathfrak{R}((\mathcal{P}+\Delta\mathcal{P})(\omega)B) \\ \mathfrak{I}((\mathcal{P}+\Delta\mathcal{P})(\omega)B) \\ \mathfrak{R}(B^*B) - I_r \\ \mathfrak{I}(B^*B) \end{pmatrix},$$

and proceed to solve $\nabla L = 0$. In our implementation we again make use of the LM method, although given the relatively cheap gradient cost, a first-order method will often be sufficient and faster. The problem is essentially tri-linear, and structurally similar to affinely structured low rank approximation, of which Lagrange multipliers will exist for most instances.

It is important to note that an attainable solution to this problem is not guaranteed, as it is possible for $\|\omega\| \to \infty$ as $\Delta\mathcal{A} \to \Delta\mathcal{A}^\star$. Such an instance is an unattainable solution in the context of Section 4.5. These solutions behave like an infinite eigenvalue and can be handed by specifically considering the eigenvalue $t = 0$ of the reversed matrix polynomial.

5.2. Computing an Initial Guess

In order to compute an initial guess to (11) we exploit the pseudo tri-linearity of the problem. If two of $\Delta\mathcal{A}$, ω and B are fixed then the problem is linear (or a linear surrogate can be solved) in the other variable. Despite the unitary constraint on B being non-linear, it is not challenging to handle. Any full rank B is suitable for an initial guess, since we may orthonormalize B to satisfy the constraint that $B^*B = I_r$.

First we approximate the determinant of $\mathcal{A}$ and consider initial guesses where $\sigma_{n-r}(\mathcal{A}(\omega^{init}))$ is reasonably small. If $\sigma_{n-r}(\mathcal{A}(\omega^{init}))$ is reasonably small, then ω^{init} is (approximately) an eigenvalue of a nearby matrix polynomial of reduced McCoy rank. The zeros and local extrema of $\det(\mathcal{A})$ are suitable candidates for computing an initial guess for ω. The kernel B^{init} can be approximated from the smallest r singular vectors of $\mathcal{A}(\omega^{init})$. This ensures that B^{init} is unitary and spans the kernel of a nearby rank deficient (scalar) matrix.

To compute an initial guess for $\Delta\mathcal{A}$ we can take $\Delta\mathcal{A}^{init} = 0$, or solve a linear least squares problem where B and ω are fixed. Alternatively, one may project to a feasible point by using a variant of Newton's method, using $\Delta\mathcal{A}^{init} = 0$, ω^{init} and B^{init} as an initial guess for the Newton iteration to solve $(\mathcal{A} + \Delta\mathcal{A})(\omega)B = 0$ and $B^*B = I_r$. A feasible point computed by Newton's method tends not to perturb $\Delta\mathcal{A}$ very much, whereas the least squares approximation may perturb $\mathcal{A}$ by an unnecessarily large amount.

5.3. About Global Optimization Methods

The problems previously discussed are NP hard to solve exactly and to approximate with coefficients from $\mathbb{Q}$. This follows since affinely structured low rank approximation (Braatz et al., 1994; Poljak and Rohn, 1993) is a special case. If we consider a matrix polynomial of degree zero, then this is a scalar matrix with an affine structure. The approximate SNF will be a matrix of rank at most $n - 2$, and finding the nearest affinely structured singular matrix is NP hard.

Despite the problem being intractable in the worst case, not all instances are necessarily hard. The formulation (11) is multi-linear and polynomial, hence amenable to the sum of squares hierarchy. Lasserre's sum of squares hierarchy (Lasserre, 2001) is a global framework for polynomial optimization that asymptotically approximates a lower bound. Accordingly, if $\|\omega^{opt}\|$ is bounded, then sum of squares techniques should yield insight into the problem.

6. Implementation and Examples

We have implemented our algorithms and techniques in the `Maple` computer algebra system[2]. We use the variant of Levenberg-Marquardt discussed in Section 4 in several instances to solve the first-order necessary condition. All computations are done using hardware precision and measured in floating point operations, or FLOPs. The input size of our problem is measured in the dimension and degree of $\mathcal{A}$, which are n and d respectively. The cost of most quasi-Newton methods is roughly proportional to inverting the Hessian matrix, which is $O(\ell^3)$, where ℓ is the number of variables in the problem.

The method of Section 4 requires approximately $O((n^3d)^3) = O(n^9d^3)$ FLOPs per iteration in an asymptotically optimal implementation with cubic matrix inversion, which is the cost of inverting the Hessian. Computing the Hessian costs roughly $O\tilde{}(n^4d^2 \times (n^2)^2) = O\tilde{}(n^8d^2)$ FLOPs

[2]Sample code is at `https://www.scg.uwaterloo.ca/software/GHL2018jsc-code-2018-11-28.tgz`.

using a blocking procedure, assuming the adjoint computation runs in $\tilde{O}(n^4 d)$ FLOPs (which can be done via interpolation in a straightforward manner)[3]. There are $O(n^3 d)$ Lagrange multipliers since the adjoint has degree at most $(n-1)d$. Using reverse-mode automatic differentiation to compute $\nabla^2 L$, this can be accomplished in $\tilde{O}(n^4 d \times n^3 d) = \tilde{O}(n^7 d^2)$ FLOPs.

The method of Section 5 has a Hessian matrix of size $O(n^2 d^2) \times O(n^2 d^2)$ in the case of a rank zero McCoy rank approximation. Accordingly, the per iteration cost is roughly $O(n^6 d^6)$ FLOPs. If the linearization is not performed, then the per-iteration cost is $O(n^6 d^3)$ FLOPs. Given the lack of expensive adjoint computation, a first-order method will typically require several orders of magnitude fewer FLOPs per iteration (ignoring the initial setup cost), with local linear convergence.

Example 6.1 (Nearest Interesting SNF)**.** *Consider the matrix polynomial $\mathcal{A}$ with a trivial SNF*

$$\begin{pmatrix} t^2+.1t+1 & 0 & .3t-.1 & 0 \\ 0 & .9t^2+.2t+1.3 & 0 & .1 \\ .2t & 0 & t^2+1.32+.03t^3 & 0 \\ 0 & .1t^2+1.2 & 0 & .89t^2+.89 \end{pmatrix}$$

of the form $\operatorname{diag}(1,\ldots,1,\det(\mathcal{A}))$.

If we prescribe the perturbations to leave zero coefficients unchanged, then using the methods of Section 4 and Section 5 results in a local minimizer $\mathcal{A}+\Delta\mathcal{A}_{opt}$ given by

$$\begin{pmatrix} 1.0619t^2+.018349t+.94098 & 0 & .27477t-.077901 & 0 \\ 0 & .90268t^2+.22581t+1.2955 & 0 & .058333 \\ .13670t & 0 & .027758t^3+.97840t^2+1.3422 & 0 \\ 0 & .10285t^2+1.1977 & 0 & .84057t^2+.93694 \end{pmatrix},$$

with $\|\Delta\mathcal{A}_{opt}\| \approx .164813183138322$*. The SNF of $\mathcal{A}+\Delta\mathcal{A}_{opt}$ is approximately*

$$\operatorname{diag}(1,1,s_1,s_1(t^5+35.388t^4+6.4540t^3+99.542t^2+5.6777t+70.015)),$$

where $s_1 \approx t^2+0.0632934647739423t+0.960572576466186$*. The factor s_1 corresponds to* $\omega_{opt} \approx -0.0316467323869714-0.979576980535687i$*.*

The method discussed in Section 4 converges to approximately 14 *decimal points of accuracy*[4] *after* 69 *iterations and the method of Section 5 converges to the same precision after approximately* 34 *iterations. The initial guess used in both instances was $\Delta\mathcal{A}_{init}=0$. The initial guesses of $\mathcal{F}^*$ and h were computed by an approximate GCD routine. For the initial guess of ω we chose a root or local extrema of* $\det(\mathcal{A})$ *that minimized the second-smallest singular value of $\mathcal{A}(\omega)$, one of which is* $\omega_{init} \approx -.12793-1.0223i$*.*

Example 6.2 (Lowest McCoy Rank Approximation)**.** *Let $\mathcal{A}$ be as in the previous example and consider the* 0*-McCoy rank approximation problem with the same prescribed perturbation structure.*

In this case we compute a local minimizer $\mathcal{A}+\Delta\mathcal{A}_{opt}$ given by

$$\begin{pmatrix} .80863t^2+1.1362 & 0 & 0 & 0 \\ 0 & .91673t^2+1.2881 & 0 & 0 \\ 0 & 0 & .95980t^2+1.3486 & 0 \\ 0 & .60052t^2+.84378 & 0 & .71968t^2+1.0112 \end{pmatrix},$$

[3] $\tilde{O}$ denotes O but with log factors removed.

[4] $\nabla L = 0$ is solved to 14 digits of accuracy; the extracted quantities are accurate to approximately the same amount.

with $\|\Delta\mathcal{A}_{opt}\| \approx .824645447014665$ *after* 34 *iterations to* 14 *decimal points of accuracy. We compute* $\omega_{opt} \approx -1.18536618732372i$ *which corresponds to the single invariant factor* $s_1 \approx t^2 + 1.4051$. *The SNF of* $\mathcal{A} + \Delta\mathcal{A}_{opt}$ *is of the form* (s_1, s_1, s_1, s_1).

7. Conclusion and Topics for Future Research

In this paper we have shown that the problem of computing a nearby matrix polynomial with a non-trivial spectral structure can be solved by (mostly local) optimization techniques. Regularity conditions were shown to hold for most instances of the problems in question, ensuring that Lagrange multipliers exist under mild assumptions about the solutions. When Lagrange multipliers do not exist, alternative formulations that admit Lagrange multipliers have been proposed. Several of these algorithms are shown to be theoretically robust with a suitable initial guess. In general, reasonable quasi-Newton methods will have rapid local convergence under normalization assumptions for all the problems considered.

There are a number of problems that remain open for future work. In particular in the case of nearby nontrivial Smith forms there is the question of obtaining such forms via polynomial row and column operations, that is, finding the unimodular matrix multipliers that will produce our nearest Smith form. Preliminary work on this topic, including the formulation as an optimization problem and the proving of the existence of Lagrange multipliers for the optimization can be found in the thesis Haraldson (2019). In some cases it may be practical to prescribe the degree structure, also called the *structural supports*, of the eigenvalues or the invariant factors of a nearby matrix polynomial. In this case, rather than look for a closest non-trivial SNF one would be interested in a closest SNF having a particular degree structure. As before this can be formulated as an optimization problem with early results available in Haraldson (2019).

References

Ahmad, S., Alam, R., 2009. Pseudospectra, critical points and multiple eigenvalues of matrix polynomials. Linear Algebra and its Applications 430 (4), 1171–1195.

Beckermann, B., Labahn, G., 1998. When are two numerical polynomials relatively prime? Journal of Symbolic Computation 26, 677–689.

Beelen, T., Van Dooren, P., 1988. An improved algorithm for the computation of Kronecker's canonical form of a singular pencil. Linear Algebra and its Applications 105, 9–65.

Bertsekas, D., 1999. Nonlinear programming. Athena Scientific, USA.

Braatz, R. P., Young, P. M., Doyle, J. C., Morari, M., 1994. Computational complexity of μ calculation. IEEE Transactions on Automatic Control 39 (5), 1000–1002.

Demmel, J., Kågström, B., 1993a. The generalized Schur decomposition of an arbitrary pencil $A - \lambda B$: robust software with error bounds and applications. Part I: theory and algorithms. ACM Transactions on Mathematical Software (TOMS) 19 (2), 160–174.

Demmel, J., Kågström, B., 1993b. The generalized Schur decomposition of an arbitrary pencil $A - \lambda$B : robust software with error bounds and applications. Part II: software and applications. ACM Transactions on Mathematical Software (TOMS) 19 (2), 175–201.

Demmel, J. W., Edelman, A., 1995. The dimension of matrices (matrix pencils) with given Jordan (Kronecker) canonical forms. Linear Algebra and its Applications 230, 61–87.

Edelman, A., Elmroth, E., Kågström, B., 1997. A geometric approach to perturbation theory of matrices and matrix pencils. Part I: Versal deformations. SIAM Journal on Matrix Analysis and Applications 18 (3), 653–692.

Edelman, A., Elmroth, E., Kågström, B., 1999. A geometric approach to perturbation theory of matrices and matrix pencils. Part II: A stratification-enhanced staircase algorithm. SIAM Journal on Matrix Analysis and Applications 20 (3), 667–699.

Fan, J.-Y., Yuan, Y.-X., 2005. On the quadratic convergence of the Levenberg-Marquardt method without nonsingularity assumption. Computing 74 (1), 23–39.

Fatouros, S., Karcanias, N., 2003. Resultant properties of gcd of many polynomials and a factorization representation of gcd. International Journal of Control 76 (16), 1666–1683.

Giesbrecht, M., Haraldson, J., Kaltofen, E., 2017a. Computing approximate greatest common right divisors of differential polynomials, Under revision.

Giesbrecht, M., Haraldson, J., Labahn, G., 2017b. Computing the nearest rank-deficient matrix polynomial. In: Proceeedings of International Symposium on Symbolic and Algebraic Computation (ISSAC'17). Kaiserslautern, Germany, pp. 181–188.

Giesbrecht, M., Haraldson, J., Labahn, G., 2017c. Lower rank approximations of matrix polynomials. Submitted to Journal of Symbolic Computation.

Giesbrecht, M., Haraldson, J., Labahn, G., 2018. Computing nearby non-trivial Smith forms. In: Proceedings of International Symposium on Symbolic and Algebraic Computation (ISSAC'18). New York, USA, pp. 159–166.

Gohberg, I., Lancaster, P., Rodman, L., 2009. Matrix polynomials. SIAM, USA.

Golub, G., Van Loan, C., 2012. Matrix Computations. Vol. 3. Johns Hopkins University Press, USA.

Haraldson, J., 2015. Computing Approximate GCRDs of Differential Polynomials. Master's thesis, Cheriton School of Computer Science, University of Waterloo.

Haraldson, J., 2019. Matrix Polynomials and their Lower Rank Approximations. Ph.D. thesis, Cheriton School of Computer Science, University of Waterloo.

Higham, N. J., 2002. Accuracy and stability of numerical algorithms. Vol. 80. SIAM.

Hoffman, A. J., 1952. On approximate solutions of systems of linear inequalities. Journal of Research of the National Bureau of Standards 49 (4).

Kailath, T., 1980. Linear systems. Vol. 156. Prentice-Hall, USA.

Kaltofen, E., Storjohann, A., 2015. The complexity of computational problems in exact linear algebra. In: Encyclopedia of Applied and Computational Mathematics. Springer, Germany, pp. 227–233.

Kaltofen, E., Yang, Z., Zhi, L., 2007. Structured low rank approximation of a sylvester matrix. In: Symbolic-Numeric Computation. Trends in Mathematics. Birkhäuser Verlag, Basel, Switzerland, pp. 69–83.

Karmarkar, N., Lakshman, Y. N., 1996. Approximate polynomial greatest common divisors and nearest singular polynomials. In: Proceedings of International Symposium on Symbolic and Algebraic Computation (ISSAC'96). ACM Press, Zurich, Switzerland, pp. 35–39.

Lasserre, J.-B., 2001. Global optimization with polynomials and the problem of moments. SIAM Journal on Optimization 11 (3), 796–817.

Lossers, O., 1974. Solution to problem 73-17: A Hadamard-type bound on the coefficients of a determinant of polynomials. SIAM Review 16 (3), 394–395.

Magnus, J., Neudecker, H., 1988. Matrix differential calculus with applications in statistics and econometrics. Wiley.

Poljak, S., Rohn, J., 1993. Checking robust nonsingularity is NP-hard. Mathematics of Control, Signals, and Systems (MCSS) 6 (1), 1–9.

Stewart, G., 1994. Perturbation theory for rectangular matrix pencils. Linear Algebra and its Applications 208, 297–301.

Van Dooren, P., 1979. The computation of Kronecker's canonical form of a singular pencil. Linear Algebra and its Applications 27, 103–140.

Van Dooren, P., Dewilde, P., 1983. The eigenstructure of an arbitrary polynomial matrix: computational aspects. Linear Algebra and its Applications 50, 545–579.

Vardulakis, A., Stoyle, P., 1978. Generalized resultant theorem. IMA Journal of Applied Mathematics 22 (3), 331–335.

Wright, S., 2005. An algorithm for degenerate nonlinear programming with rapid local convergence. SIAM Journal on Optimization 15 (3), 673–696.

Yamashita, N., Fukushima, M., 2001. On the rate of convergence of the Levenberg-Marquardt method. In: Topics in Numerical Analysis. Springer, pp. 239–249.

Zeng, Z., Dayton, B. H., 2004. The approximate GCD of inexact polynomials. In: Proceedings of International Symposium on Symbolic and Algebraic Computation (ISSAC'04). Santander, Spain, pp. 320–327.