

ACCURATE SOLUTION OF POLYNOMIAL EQUATIONS USING MACAULAY RESULTANT MATRICES

GUÐBJÖRN F. JÓNSSON AND STEPHEN A. VAVASIS

ABSTRACT. We propose an algorithm for solving two polynomial equations in two variables. Our algorithm is based on the Macaulay resultant approach combined with new techniques, including randomization, to make the algorithm accurate in the presence of roundoff error. The ultimate computation is the solution of a generalized eigenvalue problem via the QZ method. We analyze the error due to roundoff of the method, showing that with high probability the roots are computed accurately, assuming that the input data (that is, the two polynomials) are well conditioned. Our analysis requires a novel combination of algebraic and numerical techniques.

1. INTRODUCTION

In introductory linear algebra courses, eigenvalues of a matrix are computed by finding the roots of its characteristic polynomial. However, this is not how robust numerical eigenvalue algorithms work. The QR-algorithm, one of the major breakthroughs of scientific computing, computes the eigenvalues through an iteration of matrix factorizations. Having developed a robust and accurate eigenvalue algorithm, numerical analysts realized that they could go in the opposite direction, computing the roots of a univariate polynomial by computing the eigenvalues of its companion matrix. This approach has been analyzed in several papers [8, 16, 28, 30] and is, for example, used in the numerical software package Matlab.

Systems of multivariate polynomials can also be solved by conversion to eigenvalue problems. This paper focuses on the particular eigenvalue-based solution technique due to Macaulay [19, 20] and is mostly confined to the case of two polynomials in two variables. The goal is to analyze the numerical accuracy of the method. Our main contributions are twofold. First, we propose a modification to Macaulay's algorithm to make it numerically stable. Without this modification the method is unstable. Second, we provide an analysis of the accuracy of the modified method.

To our knowledge, this is the first analysis of a multivariate algebraic solver showing that the accuracy of the computed roots in the presence of roundoff error is directly connected to the conditioning of the original polynomial system. (See further remarks on the previous literature in Section 2.) Our accuracy bound is fairly weak, in the sense that it predicts very large errors (much larger than observed

Received by the editor March 7, 2001 and, in revised form, December 30, 2002.

2000 *Mathematics Subject Classification.* Primary: 13P10, 65F15. Secondary: 68W30.

This work was supported in part by NSF grants CCR-9619489 and EIA-9726388. Research also supported in part by NSF through grant DMS-9505155 and ONR through grant N00014-96-1-0050.

in practice) even in the presence of only moderate ill-conditioning of the data. The reasons for this are explained in Section 9. Our accuracy bounds are not obtained in closed form because one step in our proof is nonconstructive, relying on Hilbert’s Nullstellensatz.

The technique introduced by Macaulay is based on resultants. Given $n + 1$ polynomials in n variables, the *resultant* R is a polynomial expression in the coefficients with the property that $R = 0$ if the polynomials have a root in common. Because resultants can be expressed in terms of determinants, they can be used to translate root-finding problems into eigenvalue problems. We give the general description in later sections and in this section only consider the example of two quadratics in two variables:

$$\begin{aligned} f_1 &= a_1x^2 + a_2xy + a_3x + a_4y^2 + a_5y + a_6, \\ f_2 &= b_1x^2 + b_2xy + b_3x + b_4y^2 + b_5y + b_6. \end{aligned}$$

To solve for the roots (i.e., for pairs (x_*, y_*) such that $f_1(x_*, y_*) = f_2(x_*, y_*) = 0$), we would like to be able to reduce the number of variables, and the resultant allows us to do that. The trick is to add a third linear polynomial

$$f_3 = u_1x + u_2y + u_3.$$

The resultant of f_1, f_2, f_3 is called the *u-resultant*. For particular values of a_i ’s and b_j ’s, it is a polynomial in u_1, u_2 , and u_3 . Let $u_i = \alpha_i - \lambda\beta_i$, where α_i and β_i are some numbers. Then the resultant is a polynomial in the numeric data of the problem $(a_1, \dots, a_6, b_1, \dots, b_6, \alpha_1, \dots, \alpha_3, \text{ and } \beta_1, \dots, \beta_3)$ and in one variable λ .

Macaulay defined the resultant in terms of determinants of certain matrices. For the example in the previous paragraph, the Macaulay matrix is constructed as follows. The columns will correspond to monomials of degree 3 or less, the rows correspond to the polynomials of the form rf_i where r is a monomial and $\deg(rf_i) \leq 3$, and the entries are the coefficients of these polynomials written in the appropriate columns. This yields a 12×10 matrix (written here using lexicographical order):

$$\begin{matrix} & x^3 & x^2y & x^2 & xy^2 & xy & x & y^3 & y^2 & y & 1 \\ \begin{matrix} xf_1 \\ yf_1 \\ f_1 \\ xf_2 \\ yf_2 \\ f_2 \\ x^2f_3 \\ xyf_3 \\ xf_3 \\ y^2f_3 \\ yf_3 \\ f_3 \end{matrix} & \left[\begin{array}{cccccccccc} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & & & & & \\ & a_1 & & a_2 & a_3 & & a_4 & a_5 & a_6 & & \\ & & a_1 & & a_2 & a_3 & & a_4 & a_5 & a_6 & \\ b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & & & & & \\ & b_1 & & b_2 & b_3 & & b_4 & b_5 & b_6 & & \\ & & b_1 & & b_2 & b_3 & & b_4 & b_5 & b_6 & \\ u_1 & u_2 & u_3 & & & & & & & & \\ & u_1 & & u_2 & u_3 & & & & & & \\ & & u_1 & & u_2 & u_3 & & & & & \\ & & & u_1 & & & u_2 & u_3 & & & \\ & & & & u_1 & & & u_2 & u_3 & & \\ & & & & & u_1 & & & u_2 & u_3 & \end{array} \right]. \end{matrix}$$

It is clear that this matrix is rank deficient if f_1, f_2, f_3 have a common root (x_*, y_*) , since the nullvector is given by the column labels with the values of x_* and y_* substituted. Macaulay defined the resultant R to be the greatest common divisor (GCD) of all the 10×10 subdeterminants (as polynomials in the coefficients of f_1, f_2, f_3). We want to work with a square matrix, so we need to choose one of

the 10×10 submatrices, or equivalently choose two rows to drop. Macaulay had a specific method for dropping rows, which turns out to be unstable in some cases. We will consider dropping any two of the f_3 -rows, showing that stability is ensured provided the right choice is made. Let M be the resulting matrix and D be its determinant. Then $D = SR$, where S is called the *extraneous factor*.

Our method for choosing rows to drop is as follows. Note that the f_3 -rows correspond to monomials of degree 2 or less. Construct, in the same way as above, a matrix whose columns correspond to these monomials and where we use only f_1 and f_2 (i.e., no f_3 -rows):

$$G = \begin{bmatrix} x^2 & xy & x & y^2 & y & 1 \\ a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ b_1 & b_2 & b_3 & b_4 & b_5 & b_6 \end{bmatrix}.$$

We will show (Corollary 3) that S is (up to a sign) the 2×2 subdeterminant of G taking the columns corresponding to the same monomials as the two rows we dropped. So if we dropped, say, the rows for $x^2 f_3$ and $y f_3$, then $S = \pm(a_1 b_5 - a_5 b_1)$.

We could have $S = 0$ and therefore $D = 0$ even though the three polynomials have no roots in common. So we want to make sure we choose rows to drop so that $S \neq 0$. For numerical stability it makes sense to try to get $|S|$ as big as possible. In general, it is not practical to compute and compare all the subdeterminants of G , so we will use QR-factorization with column pivoting, which will get us within a constant factor of the best choice.

After dropping rows we have a square matrix M . If we let $u_i = \alpha_i - \lambda \beta_i$ for some numbers α_i, β_i , we get a matrix pencil, $M = A - \lambda B$. Assuming $S \neq 0$, we know the polynomials will have a common root only if $D = \det(M) = \det(A - \lambda B) = 0$. Thus, we have reduced the root-finding problem to a generalized eigenvalue problem. If λ is a simple eigenvalue, the corresponding right eigenvector has the form

$$[u^3, u^2v, u^2w, uv^2, uvw, uw^2, v^3, v^2w, vw^2, w^3]^T,$$

for some $u, v, w \in \mathbb{C}$. If $w \neq 0$, then $(x_*, y_*) = (u/w, v/w)$ is a root of f_1 and f_2 . Hence, we can determine the coordinates of the root from the eigenvector.

The choice of α_i and β_i determines how the line $u_1x + u_2y + u_3 = 0$ “sweeps” through the plane as λ is varied. We want to make a good choice to ensure that the line actually sweeps through the entire plane and that it does not pass through two or more roots simultaneously. We do not know a way of determining good choices a priori, so we just choose α_i and β_i randomly.

Bezout’s theorem (see [6]) tells us that generically f_1 and f_2 have four roots in common, counted with multiplicity. But our generalized eigenvalue problem has dimension 10. However, at least six of the eigenvalues are infinite, because

$$A - \lambda B = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} - \lambda \begin{bmatrix} 0 \\ B_2 \end{bmatrix},$$

where we have separated the top six and the bottom four rows. The eigenvectors corresponding to finite eigenvalues are in the nullspace of A_1 . We will reduce the dimension of the problem to 4 by computing an orthogonal basis Z for the nullspace of A_1 , so our reduced matrix pencil is $A_2 Z - \lambda B_2 Z$.

Thus, our algorithm consists of the following steps. First, α_i ’s and β_i ’s are chosen randomly. The rectangular matrix pencil, involving α_i ’s and β_i ’s, is formed, and then some rows are deleted to make it a square pencil $M = A - \lambda B$. Next, the

pencil is reduced to a smaller square problem via an orthogonal basis computation. The eigenvectors of the smaller problem are computed and transformed back to eigenvectors of $A - \lambda B$. The roots are determined from those eigenvectors, one root per eigenvector. In Section 5 we will give a detailed description of each step.

When does this algorithm fail? First consider the exact case, i.e., pretend all the calculations are done in exact precision. The algorithm can find only isolated roots, so it will fail if f_1 and f_2 have a factor in common, i.e., infinitely many common roots. This is equivalent to the existence of nonzero polynomials g_1 and g_2 of degrees 0 or 1, such that $g_1 f_1 + g_2 f_2 = 0$, which is equivalent to the top six rows of M being linearly dependent (since the coefficients of g_1 and g_2 form a left nullvector). If the top rows are linearly dependent, then $\det(M) = \det(A - \lambda B) = 0$ for all λ .

Since we use the eigenvectors to determine the roots, the algorithm will also fail if there is a multiple eigenvalue because then the corresponding eigenvector will probably not be uniquely determined. This will happen if either f_1 and f_2 have a root in common with multiplicity 2 or higher, or the sweepline goes through more than one root at the same time. The former is equivalent to the Jacobian of (f_1, f_2) at the root being zero. For random sweeplines the latter will almost surely not happen.

So in the exact case the method will work almost surely if the top rows of M are linearly independent and the Jacobian at the root is nonzero. But in floating-point arithmetic nonsingularity is not enough; we want the top part of M to be well conditioned and the Jacobian to have large singular values, which basically means that the root-finding problem is well conditioned.

The accuracy of a computed root depends how well conditioned the eigenvector is. Our main result is that the condition number of the eigenvector is, with high probability, bounded by a function depending only on the degrees of the polynomials, the condition number of the top part of M , and the smallest singular value of the Jacobian at the root. The probability in question is taken over the randomized choice of sweep direction.

2. RELATED WORK

The problem of numerically solving systems of nonlinear equations is among the oldest in scientific computing. In this section we give a brief overview of some of the methods in the previous literature. It is beyond the scope of this paper to provide detailed comparisons among methods since the focus of this paper is the analysis of one particular method, namely Macaulay's resultant with a linear polynomial appended.

Newton's method and its relatives [7] are general-purpose numerical methods for solving equations. In general these methods find only one root, in contrast to algebraic methods like resultants that yield all roots. Finding all roots is desirable in some settings. For example, in geometric modeling, solution of a system of polynomial equations is used to find the intersection of a line with a parametric surface. In this case, it is important to find exactly how many intersections occur in the parametric domain, since an incorrect answer could lead to an incorrect answer from a sidedness test.

Homotopy methods [18, 32] are a hybrid between Newton and algebraic methods. These methods deform an initial system of polynomial equations with a known

solution set to the particular system under consideration. During the deformation of the coefficients, the trajectories of the solutions are tracked via Newton's method. Homotopy methods have some advantages, but they run into problems if a solution trajectory passes through a region of ill-conditioning.

There are two main purely algebraic techniques, Gröbner bases and resultants. Gröbner bases (see, e.g., [6]) require computation with polynomials and determining when a polynomial is zero. In floating-point arithmetic this introduces errors. Thus, there must be some criteria to determine when a polynomial is close enough to zero to be called zero. Gröbner bases are also very computationally expensive, especially if they are done in exact arithmetic. Resultant-based methods have the advantage from a numeric standpoint in that there is no floating-point polynomial calculation needed to form the resultant matrices, and all the floating-point computation is then encapsulated in well-understood matrix algorithms. The main disadvantage of resultants is that they require solution of an eigensystem of size $s \times s$, where s is the number of solutions to the polynomial equations. Standard eigenvalue algorithms require $O(s^3)$ operations. Since s grows exponentially in the number of variables in the polynomial system, resultant computations are feasible only for small numbers of variables. The technique of *sparse resultants* [6, 9, 10], applicable to sparse polynomial systems, somewhat ameliorates this drawback.

Although our focus is on Macaulay resultants, other resultants can also be applied to solve polynomial systems [9]. For example, in the case of two equations in two variables (the primary focus of this work), the Sylvester resultant described in Section 3 can be used to eliminate one variable. Then the system can be solved for the other variable. Once the roots of one variable are known, those values can be substituted into either polynomial to solve for the other variable. Unlike the Macaulay approach, this method (as well as some other resultant methods) has the disadvantage that the system of equations ultimately solved does not contain the original coefficients, but rather contains quantities derived from the original coefficients. The intermediate operations will complicate the analysis and may even lead to subtle numerical accuracy problems.

For example, when using the Sylvester resultant algorithm sketched in the previous paragraph, one expects difficulties when there are two well-separated roots with the same abscissa, e.g., (x_1, y_1) and (x_1, y_2) , since elimination of y will leave a univariate polynomial with a double-root that cannot be accurately computed. This problem can be eliminated (with high probability) by first applying a random linear change of variables. Indeed, this random linear change of variables in the Sylvester resultant is closely related to the randomized line sweep proposed herein, as shown by our analysis.

For a good introduction to the theory of resultants, see [6]. Emiris and Mourrain [9] give a survey of the various resultant matrices, and the structure of these matrices is investigated in [24]. We are certainly not the first to explore the use of resultants for the numeric solution of polynomial equations, see, e.g., [1, 5, 21, 22]. These papers discuss various aspects of the problem. But none carries out the complete analysis that connects conditioning of the original polynomial system to accuracy of the computed roots as we do in this paper.

3. MACAULAY RESULTANT MATRICES

The idea behind resultants is to find a condition on the coefficients of a system of polynomials that determines when there is a common zero. To better understand this, it is helpful to first consider the simplest resultant, the *Sylvester resultant*. Suppose we are given two univariate polynomials

$$p = a_m x^m + \cdots + a_1 x + a_0 \quad \text{and} \quad q = b_n x^n + \cdots + b_1 x + b_0.$$

Assume that not both leading coefficients are zero. Then the two polynomials have a root in common if and only if they have a common factor, which happens if and only if there exist nonzero polynomials r and s with $\deg(r) \leq n - 1$, $\deg(s) \leq m - 1$, such that

$$rp + sq = 0.$$

This leads to a system of $m + n$ linear equations for the $m + n$ coefficients of r and s . The corresponding matrix is

$$\begin{bmatrix} a_0 & a_1 & \cdots & a_m & & & \\ & \ddots & \ddots & & \ddots & & \\ & & a_0 & a_1 & \cdots & a_m & \\ b_0 & b_1 & \cdots & b_n & & & \\ & \ddots & \ddots & & \ddots & & \\ & & b_0 & b_1 & \cdots & b_n & \end{bmatrix} \left. \begin{array}{l} \vphantom{\begin{bmatrix} \\ \\ \\ \\ \\ \end{bmatrix}} \right\} n \text{ rows} \\ \vphantom{\begin{bmatrix} \\ \\ \\ \\ \\ \end{bmatrix}} \right\} m \text{ rows.}$$

The Sylvester resultant is the determinant of this matrix. The polynomials have a root in common if and only if the linear system has a nontrivial solution which happens if and only if the determinant is zero.

Now we will describe Macaulay's construction (see Chapter 1 in his book [20]). Suppose we have $n + 1$ homogeneous polynomials F_0, \dots, F_n in $n + 1$ variables x_0, \dots, x_n . Let d_0, \dots, d_n denote the degrees of the polynomials and assume all the degrees are positive. Construct a matrix, which we will denote by $M_t(F_0, \dots, F_n)$, as follows. The columns correspond to monomials of degree t , the rows correspond to polynomials of the form rF_i where r is a monomial and $\deg(rF_i) = t$, and the entries are the coefficients of these polynomials written in the appropriate columns.

$$M_t(F_0, \dots, F_n) = \begin{array}{c} \begin{array}{c} x_0^t, x_0^{t-1}x_1, \dots, x_0^{t-1}x_n, \dots, x_n^t \\ \hline F_0\text{-rows} \\ \hline \vdots \\ \hline F_n\text{-rows} \end{array} \end{array}.$$

This matrix has $\binom{t+n}{n}$ columns and the number of F_i -rows is $\binom{t-d_i+n}{n}$. It can also be described as the transpose of the matrix for the linear map

$$(P_0, \dots, P_n) \mapsto P_0 F_0 + \cdots + P_n F_n,$$

where P_i is homogeneous of degree $t - d_i$. We will also use the notation $M_i(F_0, \dots, F_k)$ for the matrix where we only take the F_i -rows for $i = 1, \dots, k$. It should always be clear from the context what the degrees of the polynomials are and how many variables they have.

Let $d = d_0 + \dots + d_n - n$ and consider the matrix $M_d = M_d(F_0, \dots, F_n)$. If $n = 1$ or $d_0 = \dots = d_n = 1$, then this matrix is square, but otherwise it has more rows than columns. Consider all the submatrices where we take all the columns and an equal number of rows. The determinants of these submatrices are polynomials in the coefficients of F_0, \dots, F_n , where we are thinking of the coefficients as symbols that have not yet been assigned values. Let R be the greatest common divisor (GCD) of all these determinants (as elements in the ring $\mathbb{Z}[\text{coefficients}]$, and choosing the sign so that part 3 of the following theorem is satisfied).

Theorem 1. *Let R be defined as above. Then R is the resultant of F_0, \dots, F_n , denoted by $\text{Res}(F_0, \dots, F_n)$. That is, it satisfies the following.*

- (1) *For particular values of the coefficients, R is zero if and only if the system*

$$F_0(x_0, \dots, x_n) = \dots = F_n(x_0, \dots, x_n) = 0$$

has a nontrivial solution.

- (2) *R is irreducible as a polynomial in $\mathbb{C}[\text{coefficients}]$ and $\mathbb{Z}[\text{coefficients}]$.*
- (3) *$\text{Res}(x_0^{d_0}, \dots, x_n^{d_n}) = 1$.*
- (4) *R is homogeneous of degree $d_0 \dots d_{i-1} d_{i+1} \dots d_n$ in the coefficients of F_i .*

Macaulay [20] defines the resultant as the GCD and then proves it has these properties. It is more common (see, e.g., [6]) to define the resultant as the unique polynomial satisfying the first three properties (for a proof that such a polynomial exists in general, see [10, 29]) and then show that the GCD equals the resultant. As already mentioned in Section 2, the theory of resultants can be generalized for sparse polynomials [6, 9] and even further [10].

Example 1. Suppose we have

$$\begin{aligned} F_0 &= a_1x^2 + a_2xy + a_3xz + a_4y^2 + a_5yz + a_6z^2, \\ F_1 &= b_1x + b_2y + b_3z, \\ F_2 &= c_1x + c_2y + c_3z. \end{aligned}$$

Then $d = 2 + 1 + 1 - 2 = 2$, $M_2(F_0, F_1, F_2)$ is

$$\begin{matrix} & x^2 & xy & xz & y^2 & yz & z^2 \\ \begin{matrix} F_0 \\ xF_1 \\ yF_1 \\ zF_1 \\ xF_2 \\ yF_2 \\ zF_2 \end{matrix} & \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \\ b_1 & b_2 & b_3 & 0 & 0 & 0 \\ 0 & b_1 & 0 & b_2 & b_3 & 0 \\ 0 & 0 & b_1 & 0 & b_2 & b_3 \\ c_1 & c_2 & c_3 & 0 & 0 & 0 \\ 0 & c_1 & 0 & c_2 & c_3 & 0 \\ 0 & 0 & c_1 & 0 & c_2 & c_3 \end{bmatrix} & , \end{matrix}$$

and the resultant is

$$\begin{aligned} R = & a_1b_2^2c_3^2 - 2a_1b_2b_3c_2c_3 + a_1b_3^2c_2^2 - a_2b_1b_2c_3^2 + a_2b_1b_3c_2c_3 + a_2b_2b_3c_1c_3 \\ & - a_2b_3^2c_1c_2 + a_3b_1b_2c_2c_3 - a_3b_1b_3c_2^2 - a_3b_2^2c_1c_3 + a_3b_2b_3c_1c_2 + a_4b_1^2c_3^2 \\ & - 2a_4b_1b_3c_1c_3 + a_4b_3^2c_1^2 - a_5b_1^2c_2c_3 + a_5b_1b_2c_1c_3 + a_5b_1b_3c_1c_2 - a_5b_2b_3c_1^2 \\ & + a_6b_1^2c_2^2 - 2a_6b_1b_2c_1c_2 + a_6b_2^2c_1^2. \end{aligned}$$

We have defined the resultant for homogeneous polynomials because it is, in some sense, natural to work in projective space. However, resultants can also be defined for inhomogeneous polynomials. Suppose we have $n+1$ (inhomogeneous) polynomials f_0, \dots, f_n in n variables x_1, \dots, x_n . The resultant $\text{Res}(f_0, \dots, f_n)$ is the same as the resultant where the polynomials have been homogenized by introducing a new variable x_0 , i.e., it equals $\text{Res}(F_0, \dots, F_n)$ where F_i is a homogeneous polynomial of the same degree as f_i such that $F_i|_{x_0=1} = f_i$. Equivalently, we could have repeated the matrix construction, replacing monomials of degree exactly t with monomials of degree at most t , and so on, creating the matrix $M_d(f_0, \dots, f_n) = M_d(F_0, \dots, F_n)$.

For inhomogeneous polynomials it is still true that if they have a root in common, then the resultant is zero, but the converse does not always hold. A nontrivial root of F_i is either a root of $f_i = F_i|_{x_0=1}$ or it is a nontrivial root of $\bar{F}_i = F_i|_{x_0=0}$. So we have $\text{Res}(f_0, \dots, f_n) = 0$ if and only if the inhomogeneous system

$$f_0(x_1, \dots, x_n) = \dots = f_n(x_1, \dots, x_n) = 0$$

has a solution or the homogeneous system

$$\bar{F}_0(x_1, \dots, x_n) = \dots = \bar{F}_n(x_1, \dots, x_n) = 0$$

has a nontrivial solution. In [6], a solution to $\bar{F}_0 = \dots = \bar{F}_n = 0$ is referred to as a “solution at ∞ ” to $f_0 = \dots = f_n = 0$.

As we have already mentioned, the matrix $M_d = M_d(F_0, \dots, F_n)$ usually has more rows than columns, so to get a square matrix we need to drop some rows. Let R be the resultant and D be the determinant of the square matrix we get after dropping rows. Then

$$D = SR,$$

where S is called the *extraneous factor*.

Macaulay had a specific way of dropping rows. He kept all the F_0 -rows, and for $i = 1, \dots, n$, he dropped the F_i -rows corresponding to monomials divisible by one of $x_0^{d_0}, \dots, x_{i-1}^{d_{i-1}}$. He then shows that this gives a square matrix M and that S can be found as a certain subdeterminant of M (see [19]). This gives a method for computing the resultant as a ratio of two determinants. In Example 1, Macaulay would have dropped the yF_2 -row and found that $S = b_2$. In our numerical experiments we found that Macaulay’s choice of dropped rows sometimes leads to very inaccurate solutions. The explanation for this follows below.

4. CAYLEY’S RESULTANT FORMULA

Below we present Cayley’s formula for the resultant given in [3] and proved in [10]. Cayley’s formula gives a way to compute the extraneous factor. It also yields an expression for the dropped rows in terms of the remaining rows that will be used in later analysis.

Suppose we have a system of r_1 linear equations $A_1\mathbf{x} = \mathbf{0}$ in r_0 variables. Also suppose that these equations are not all independent, but connected by r_2 linear

equations that may not all be independent but connected by r_3 linear equations, and so on for some number of levels k . At the last level, the equations are independent. Let A_1, \dots, A_k be the matrices for these linear systems. Then A_j is a $r_j \times r_{j-1}$ matrix and

$$A_{j+1}A_j = 0, \quad j = 1, \dots, k - 1.$$

This can also be viewed as a sequence of linear operators on vector spaces over a field K ,

$$0 \longrightarrow K^{r_0} \xrightarrow{A_1} K^{r_1} \xrightarrow{A_2} \dots \xrightarrow{A_k} K^{r_k} \longrightarrow 0.$$

A sequence like this is also called a complex of K -vector spaces [10]. Assume the sequence (or the complex) is *exact*, i.e., $\text{Im}(A_j) = \text{Ker}(A_{j+1})$.

For $j = 1, \dots, k$, let

$$s_j = r_{j-1} - r_{j-2} + \dots - (-1)^j r_0.$$

Assuming the sequence is exact, then it must be the case that $\text{rank}(A_j) = s_j$. Since A_k has full row rank by assumption, $s_k = r_k$.

Next, we select a sequence of determinants using the following procedure. Choose any r_0 out of the r_1 rows of A_1 and let D_1 be the determinant of the resulting matrix. For $j = 2, \dots, k$, choose any s_j rows of A_j and let D_j be the determinant of the matrix formed by these rows and the columns from A_j with complementary (with respect to $\{1, 2, \dots, r_{j-1}\}$) indices of the rows we chose for A_{j-1} . (Cayley used “supplementary” in this context, although “complementary” would seem to be clearer to the modern reader.)

Theorem 2. *There exists $R \in K$ such that for any sequence of determinants D_1, \dots, D_k chosen as above,*

$$(D_2 D_4 \dots) R = \pm (D_1 D_3 \dots).$$

If the determinants are nonzero, $R = \pm D_1 / (D_2 / \dots / (D_{k-1} / D_k))$.

Proof. See [10]. □

Consider the case of three homogeneous polynomials F_1, F_2, F_3 in x, y, z of degrees d_1, d_2, d_3 , respectively. The matrix $M_d = M_d(F_1, F_2, F_3)$ with $d = d_1 + d_2 + d_3 - 2$ can be considered as a linear system of equations

$$qF_i = 0, \quad q \text{ monomial of degree } d - d_i \text{ and } i = 1, 2, 3,$$

in variables $x^d, x^{d-1}y, \dots, z^d$. There are $r_1 = \binom{d_2+d_3}{2} + \binom{d_1+d_3}{2} + \binom{d_1+d_2}{2}$ equations in $r_0 = \binom{d_1+d_2+d_3}{2}$ variables, so there are $\binom{d_1}{2} + \binom{d_2}{2} + \binom{d_3}{2}$ more equations than variables.

The matrix M_d can also be viewed as the transpose of the matrix for the linear map

$$(P_1, P_2, P_3) \mapsto P_1F_1 + P_2F_2 + P_3F_3,$$

where P_i is homogeneous of degree $d - d_i$. Consider the nullspace for the map given by the equation

$$P_1F_1 + P_2F_2 + P_3F_3 = 0.$$

We see that $P_1 = QF_2, P_2 = -QF_1, P_3 = 0$ is a solution for the equation for any homogeneous Q of degree $d - d_1 - d_2 = d_3 - 2$. There are $\binom{d_3}{2}$ such Q . Similarly we get $\binom{d_2}{2}$ solutions $P_1 = QF_3, P_2 = 0, P_3 = -QF_1$ and $\binom{d_1}{2}$ solutions $P_1 = 0, P_2 = QF_3, P_3 = -QF_2$.

This gives $r_2 = \binom{d_1}{2} + \binom{d_2}{2} + \binom{d_3}{2}$ linearly independent equations for the rows of M_d , written in matrix form as

$$H_d(F_1, F_2, F_3) = \begin{bmatrix} -M_{d-d_1}(F_2) & M_{d-d_2}(F_1) & 0 \\ -M_{d-d_1}(F_3) & 0 & M_{d-d_3}(F_1) \\ 0 & -M_{d-d_2}(F_3) & M_{d-d_3}(F_2) \end{bmatrix}.$$

Then we have an exact complex,

$$0 \longrightarrow K^{r_0} \xrightarrow{M_d} K^{r_1} \xrightarrow{H_d} K^{r_2} \longrightarrow 0,$$

where $K = \mathbb{C}(\text{coefficients})$ is the field of rational functions in the coefficients of the polynomials F_1, F_2, F_3 . So we can apply Theorem 2, with R clearly being the resultant, to get the extraneous factor. Note that the complex is exact in the symbolic case, i.e., the case in which all matrix entries are indeterminates. Once we substitute actual numbers for the coefficients, it may turn out that M_d is rank deficient. Indeed, this is exactly what happens when F_1, F_2, F_3 have a common root.

Corollary 3. *The extraneous factor for the matrix we get after dropping $\binom{d_1}{2} + \binom{d_2}{2} + \binom{d_3}{2}$ rows from $M_d(F_1, F_2, F_3)$ is (up to a sign) the determinant of the matrix where we take the columns of $H_d(F_1, F_2, F_3)$ that correspond to the rows we dropped.*

5. SOLVING POLYNOMIAL EQUATIONS

For the rest of the paper we will focus on the case of finding the roots of two polynomials in two variables or, equivalently, two homogeneous polynomials in three variables. In this section we explain how Macaulay's matrices can be used to convert the problem of solving polynomial equations into an eigenvalue problem. We describe our algorithm, which is based on Macaulay's work, but with a few modifications to increase numerical stability.

Suppose we are given polynomials f_1 and f_2 in $\mathbb{C}[x, y]$ of degrees d_1 and d_2 ,

$$\begin{aligned} f_1(x, y) &= a_1x^{d_1} + a_2x^{d_1-1}y + \cdots + a_\mu, \\ f_2(x, y) &= b_1x^{d_2} + b_2x^{d_2-1}y + \cdots + b_\nu. \end{aligned}$$

We want to determine all the solutions to

$$(1) \quad f_1(x, y) = f_2(x, y) = 0$$

with as much accuracy as possible. The trick to using resultants to solve this problem is to add a third linear polynomial,

$$f_3(x, y) = u_1x + u_2y + u_3.$$

The resultant $R = \text{Res}(f_1, f_2, f_3)$ is called the *u-resultant*, and it is a polynomial in u_1, u_2, u_3 given particular values for the coefficients of f_1 and f_2 .

The algorithm will actually solve

$$(2) \quad F_1(x, y, z) = F_2(x, y, z) = 0,$$

where we have homogenized the polynomials,

$$\begin{aligned} F_1(x, y, z) &= a_1x^{d_1} + a_2x^{d_1-1}y + \cdots + a_\mu z^{d_1}, \\ F_2(x, y, z) &= b_1x^{d_2} + b_2x^{d_2-1}y + \cdots + b_\nu z^{d_2}. \end{aligned}$$

We also homogenize the linear polynomial

$$F_3(x, y, z) = u_1x + u_2y + u_3z,$$

and note that $\text{Res}(F_1, F_2, F_3) = \text{Res}(f_1, f_2, f_3)$. It is more natural to analyze the homogeneous case, because solutions at ∞ (with $z = 0$) are not harder to compute than any other solutions. We can then always specialize the results to the inhomogeneous case: Given a solution (x_*, y_*, z_*) of (2) with $z_* \neq 0$, then $(x_*/z_*, y_*/z_*)$ is a solution of (1).

Example 2. The example of the two quadratics used in the Introduction is a little too simple to exhibit the full complexity of the problem. Therefore, we will use the example of a cubic and a quadratic:

$$\begin{aligned} F_1 &= a_1x^3 + a_2x^2y + a_3x^2z + a_4xy^2 + a_5xyz + a_6xz^2 + a_7y^3 + a_8y^2z \\ &\quad + a_9yz^2 + a_{10}z^3, \\ F_2 &= b_1x^2 + b_2xy + b_3xz + b_4y^2 + b_5yz + b_6z^2, \\ F_3 &= u_1x + u_2y + u_3z. \end{aligned}$$

Then $d = 3 + 2 + 1 - 2 = 4$ and $M_d(F_1, F_2, F_3)$ is as follows:

$$\begin{matrix} & x^4 & x^3y & x^3z & x^2y^2 & x^2yz & x^2z^2 & xy^3 & xy^2z & xyz^2 & xz^3 & y^4 & y^3z & y^2z^2 & yz^3 & z^4 \\ \begin{matrix} x \\ y \\ z \\ x^2 \\ xy \\ xz \\ y^2 \\ yz \\ z^2 \\ x^3 \\ x^2y \\ x^2z \\ xy^2 \\ xyz \\ xz^2 \\ y^3 \\ y^2z \\ yz^2 \\ z^3 \end{matrix} & \left[\begin{array}{cccccccccccccccc} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & a_{10} & & & & & & \\ & a_1 & & a_2 & a_3 & & a_4 & a_5 & a_6 & & a_7 & a_8 & a_9 & a_{10} & & \\ & & a_1 & & a_2 & a_3 & & a_4 & a_5 & a_6 & & a_7 & a_8 & a_9 & a_{10} & \\ b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & & & & & & & & & & \\ & b_1 & & b_2 & b_3 & & b_4 & b_5 & b_6 & & & & & & & \\ & & b_1 & & b_2 & b_3 & & b_4 & b_5 & b_6 & & & & & & \\ & & & b_1 & & & b_2 & b_3 & & & b_4 & b_5 & b_6 & & & \\ & & & & b_1 & & & & b_2 & b_3 & & & & b_4 & b_5 & b_6 \\ u_1 & u_2 & u_3 & & & & & & & & & & & & & \\ & u_1 & & u_2 & u_3 & & & & & & & & & & & \\ & & u_1 & & u_2 & u_3 & & & & & & & & & & \\ & & & u_1 & & & u_2 & u_3 & & & & & & & & \\ & & & & u_1 & & & & u_2 & u_3 & & & & & & \\ & & & & & u_1 & & & & & u_2 & u_3 & & & & \\ & & & & & & u_1 & & & & & u_2 & u_3 & & & \\ & & & & & & & u_1 & & & & & u_2 & u_3 & & \\ & & & & & & & & u_1 & & & & & u_2 & u_3 & \end{array} \right]. \end{matrix}$$

This is a 19×15 matrix, and the resultant is the GCD of the determinants of all the 15×15 submatrices.

Now $d = d_1 + d_2 - 1$ and the matrix $M_d = M_d(F_1, F_2, F_3)$ has

$$\binom{d_2 + 1}{2} + \binom{d_1 + 1}{2} + \binom{d_1 + d_2}{2} - \binom{d_1 + d_2 + 1}{2} = \binom{d_1}{2} + \binom{d_2}{2}$$

more rows than columns. So which rows should we drop to turn M_d into a square matrix? If D is the determinant of the resulting square matrix M , then it is a multiple of the resultant, $D = SR$. By Corollary 3, the extraneous factor S is (up to a sign) the determinant of the matrix where we take the columns of

$$(3) \quad H_d(F_1, F_2, F_3) = \begin{bmatrix} -M_{d-d_1}(F_3) & 0 & M_{d-1}(F_1) \\ 0 & -M_{d-d_2}(F_3) & M_{d-1}(F_2) \end{bmatrix}$$

that correspond to the rows we dropped.

Suppose we drop only F_3 -rows. Then S is, up to a sign, the determinant of the matrix we get by taking the columns of $M_{d-1}(F_1, F_2)$ corresponding to the same monomials as the rows we dropped from M_d .

Example 3. For the polynomials in Example 2, $M_{d-1}(F_1, F_2)$ is

$$\begin{array}{c} 1 \\ x \\ y \\ z \end{array} \begin{array}{cccccccccc} x^3 & x^2y & x^2z & xy^2 & xyz & xz^2 & y^3 & y^2z & yz^2 & z^3 \\ \left[\begin{array}{cccccccccc} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & a_{10} \\ b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & & & & \\ & b_1 & & b_2 & b_3 & & b_4 & b_5 & b_6 & \\ & & b_1 & & b_2 & b_3 & & b_4 & b_5 & b_6 \end{array} \right] \end{array}.$$

Macaulay's rule for dropping rows specifies that the F_3 -rows corresponding to monomials divisible by x^3 or y^2 , that is, monomials x^3 , xy^2 , y^3 and y^2z , are dropped. In this case, the extraneous factor is given by

$$(4) \quad S = \pm \begin{vmatrix} a_1 & a_4 & a_7 & a_8 \\ b_1 & b_4 & & \\ & b_2 & b_4 & b_5 \\ & & & b_4 \end{vmatrix}.$$

If this matrix is singular or nearly singular, then the algorithm finding the roots of the polynomial system $F_1 = F_2 = 0$ will work poorly regardless of the choice of u_1, u_2, u_3 . For example, we could consider the system given in inhomogeneous form by

$$(5) \quad \begin{aligned} F_1 &= x^3 - xy^2 + y^3 - 2, \\ F_2 &= x^2 - y^2 + 1, \end{aligned}$$

in which the first two columns on the right-hand side of (4) agree, hence the determinant is 0.

We want a matrix whose determinant is a nonzero multiple of the resultant; i.e., we want to choose rows to drop so that $S \neq 0$. Better yet, to increase numerical stability of the root-finding algorithm, we would like to make $|S|$ as big as possible. We drop $h = \binom{d_2}{2} + \binom{d_1}{2}$ out of the $k = \binom{d_1+d_2}{2}$ rows corresponding to the linear polynomial F_3 . The number of different choices is $\binom{k}{h}$, which grows exponentially in d_1 and d_2 . This means that trying all possibilities to choose the best is computationally infeasible.

Instead we use QR-factorization with column pivoting. Let $G = M_{d-1}(F_1, F_2)$. Then G is an $h \times k$ matrix (with $h \leq k$). QR-factorization with column pivoting (see Section 5.4.1 in [11]) gives

$$GP = QT,$$

where P is a permutation matrix, Q is an $h \times h$ unitary matrix, and T is an $h \times k$ upper triangular matrix (actually "upper trapezoidal", since $h < k$ in general) with

$$t_{11} \geq t_{22} \geq \cdots \geq t_{hh}$$

and

$$\|[t_{ij}, \dots, t_{hj}]\| \leq t_{ii}, \quad 1 \leq i \leq h, \quad i+1 \leq j \leq k.$$

If T' is any $h \times h$ submatrix of T , then we can write $T' = \text{diag}([t_{11}, \dots, t_{hh}])T''$, where all the entries of T'' are at most 1 in absolute value. The determinant of a matrix is bounded above by the product of the 2-norms of its columns (Hadamard's inequality). The 2-norm of each column of T'' is bounded by \sqrt{h} , so $\det(T'') \leq h^{h/2}$, and therefore

$$|\det(T')| \leq h^{h/2} t_{11} \cdots t_{hh}.$$

Let H consist of the first h columns of GP and let H' be any $h \times h$ submatrix of G . We will drop the rows corresponding to the same monomials as the columns of H . Then

$$|S| = |\det(H)| \geq \frac{1}{h^{h/2}} |\det(H')|,$$

so we are guaranteed to be within a factor of $h^{h/2}$ of the maximum $h \times h$ subdeterminant of G .

We are also interested in the smallest singular value of H . Let $\sigma_i(C)$ denote the i -th biggest singular value of a matrix C . Gu and Eisenstat [13] provide a bound (Theorem 7.2 in their paper)

$$\sigma_i(H) \geq \frac{\sigma_i(G)}{\sqrt{k-i} 2^i},$$

and for $i = h$ this gives

$$(6) \quad \sigma_h(H) \geq \frac{\sigma_h(G)}{\sqrt{d_1 d_2} 2^h}.$$

We will use this bound in the analysis in Section 8.

After dropping rows, we have a square matrix M with the property that if

$$(7) \quad F_1(x, y, z) = F_2(x, y, z) = F_3(x, y, z) = 0$$

has a solution, then $D = \det(M) = 0$. Let

$$u_i = \alpha_i - \lambda\beta_i, \quad \text{for } i = 1, 2, 3,$$

where α_i, β_i are randomly chosen numbers. Then $M = A - \lambda B$, and (7) has a solution only if $\det(A - \lambda B) = 0$. This reduces the root-finding problem to a generalized eigenvalue problem,

$$(8) \quad A\mathbf{x} = \lambda B\mathbf{x}.$$

Moreover, in the generic case, we will have eigenvectors of the form

$$[x_*^d, x_*^{d-1}y_*, x_*^{d-1}z_*, \dots, z_*^{dT}]^T,$$

where (x_*, y_*, z_*) is a solution to (7). If $z_* \neq 0$, then $(x_*/z_*, y_*/z_*)$ is a solution of equation (1). But if $z_* = 0$, we have a root at ∞ .

The geometric picture is as follows. The equations $f_1(x, y) = 0$ and $f_2(x, y) = 0$ define two curves in the plane. The third equation, $f_3(x, y) = 0$, defines a line, and as we vary λ this line sweeps through the plane. Thus, the eigenvalues of (8) correspond to the instances when the ‘‘sweep line’’ passes through a common root of f_1 and f_2 . (This could possibly be a root at ∞ .) Figure 1 shows an example of a good and a bad sweep line.

For the homogeneous system, roots are points in the two-dimensional projective space \mathbb{P}^2 , where the points are lines through the origin in \mathbb{C}^3 . The sweep line becomes a plane in \mathbb{C}^3 that rotates around the origin. If we visualize this, we see that there are two ways in which the line sweeps through the plane in the affine case. It is either translated or it rotates around a fixed point.

If $[\alpha_1, \alpha_2, \alpha_3]$ is a multiple of $[\beta_1, \beta_2, \beta_3]$, then the sweep line does not move at all. We also want to avoid choices of α_i, β_i where the sweep line simultaneously moves through two or more roots. In this case we get a multiple eigenvalue and we will not be able to determine the roots from the computed eigenvectors. We do not know of a way to prevent this without first knowing the roots, which is why we make a random choice of sweep line.

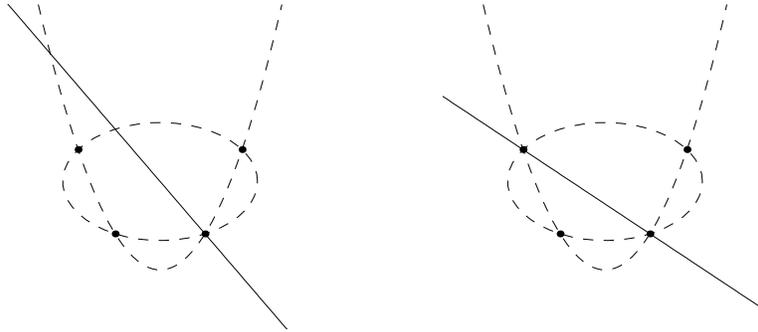


FIGURE 1. An example with two quadratic curves. The left plot shows a “good” swepline going through exactly one root. The right plot shows a “bad” swepline going through two roots at the same time.

By Bezout’s theorem (see [6]) the system of equations $F_1(x, y, z) = F_2(x, y, z) = 0$ has $d_1 d_2$ solutions, counted with multiplicity. However, the generalized eigenvalue problem (8) has dimension $\binom{d_1+d_2+1}{2}$. Note that the upper part of the matrix B is zero; i.e., our matrix pencil has the structure

$$A - \lambda B = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} - \lambda \begin{bmatrix} 0 \\ B_2 \end{bmatrix},$$

where A_1 consists of the F_1 -rows and F_2 -rows, while A_2 and B_2 consist of the remaining F_3 -rows (the F_3 -rows that were not dropped). We will refer to A_1 and A_2 as the *top* and *bottom* parts of A , respectively. The bottom part has $d_1 d_2$ rows, matching the number of solutions.

Because the top part of B is zero, there are infinite eigenvalues that are irrelevant to the problem, i.e., they have nothing to do with the polynomials F_1 and F_2 . We want the $d_1 d_2$ relevant eigenvalues/eigenvectors, without also computing the irrelevant ones. (The eigenvalues for the relevant eigenvectors are usually finite, but may be infinite if the swepline is chosen so that it hits a root at $\lambda = \infty$.) In the symbolic computation literature, it has been proposed to carry this out via a Schur complement. If

$$A - \lambda B = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} - \lambda \begin{bmatrix} 0 & 0 \\ B_{21} & B_{22} \end{bmatrix},$$

with A_{11} square and nonsingular, then the finite eigenvalues are also eigenvalues of

$$(A_{22} - A_{21}A_{11}^{-1}A_{12}) - \lambda(B_{22} - B_{21}A_{11}^{-1}A_{12}).$$

Numerically we would need A_{11} to be well conditioned, so we would need to permute the columns to ensure that.

We suggest a different approach. For numerical stability it is usually a good idea to use orthogonal transformations where possible. Note that the eigenvectors of (8) corresponding to finite eigenvalues are in the nullspace of A_1 . Compute a matrix Z whose columns form an orthonormal basis for the nullspace of A_1 using

QR-factorization,

$$A_1^H = [Q_1, Z] \begin{bmatrix} R_1 \\ 0 \end{bmatrix},$$

where $[Q_1, Z]$ is unitary and R_1 is upper triangular. Then the finite eigenvalues are also eigenvalues of the smaller eigenvalue problem

$$(9) \quad A_2 Z \mathbf{y} = \lambda B_2 Z \mathbf{y}.$$

Having solved this, the eigenvector for the bigger eigenvalue problem (8) can be recovered simply by multiplying by Z , i.e., $\mathbf{x} = Z \mathbf{y}$.

In the generic case, the exact eigenvector \mathbf{x} has a structure corresponding to the monomial structure of the columns of M_d ,

$$(10) \quad \mathbf{x} = [x_*^d, x_*^{d-1} y_*, x_*^{d-1} z_*, \dots, x_* z_*^{d-1}, y_*^d, y_*^{d-1} z_*, \dots, z_*^d]$$

for some $x_*, y_*, z_* \in \mathbb{C}$. The argument that \mathbf{x} is of this form has two parts. The first part, which follows directly by construction of M_d , is that if F_1, F_2, F_3 have a common root (x_*, y_*, z_*) not all zero, then \mathbf{x} given by (10) is in the nullspace of $A - \lambda B$. The second part of the argument is that this nullspace is one-dimensional. Macaulay’s argument for this second part is outlined in Example 5.

Assume the computed eigenvector $\hat{\mathbf{x}}$ is close to \mathbf{x} . We determine x_*, y_*, z_* as follows. find the maximum in absolute value of the entries of $\hat{\mathbf{x}}$ that correspond to x^d, y^d , and z^d . If it corresponds to, say x^d , compute x_*, y_*, z_* by taking the entries corresponding to $x^d, x^{d-1}y$, and $x^{d-1}z$.

We make one more notational change to our algorithm. Instead of writing the pencil of the generalized eigenvalue problem as $A - \lambda B$, we write it as $sA - tB$, where s, t are complex scalars that are not both zero. This makes the problem symmetric with respect to A and B (e.g., the case of “infinite” eigenvalues of $A - \lambda B$ now corresponds to the case $s = 0, t = 1$). Because the rank of $sA - tB$ is scale invariant, we assume s, t are normalized so that $|s|^2 + |t|^2 = 1$. Thus, the generalized eigenvalue problem is to find solutions to

$$(11) \quad sA\mathbf{x} = tB\mathbf{x},$$

with $|s|^2 + |t|^2 = 1$ and $\mathbf{x} \neq \mathbf{0}$.

Here is a summary of the complete algorithm.

- (1) Choose $\alpha_i, \beta_i, i = 1, 2, 3$, randomly. To be specific, choose $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \alpha_3]$ from a uniform distribution over the set $\{\mathbf{z} \in \mathbb{C}^3 : \|\mathbf{z}\|_2 = 1\}$, and then choose $\boldsymbol{\beta} = [\beta_1, \beta_2, \beta_3]$ from a uniform distribution over $\{\mathbf{z} \in \mathbb{C}^3 : \boldsymbol{\alpha}^* \mathbf{z} = 0, \|\mathbf{z}\|_2 = 1\}$. Also normalize the coefficients of F_1 and F_2 so that $\|\mathbf{a}\|_2 = \|\mathbf{b}\|_2 = 1$.
- (2) Form the rectangular matrix pencil $M_d(F_1, F_2, F_3) = sA_d - tB_d$.
- (3) Compute a QR-factorization with column pivoting of $M_{d-1}(F_1, F_2)$, and let P be the resulting permutation matrix. Drop the F_3 -rows of $M_d = sA_d - tB_d$ corresponding to the same monomials as the first $\binom{d_2}{2} + \binom{d_1}{2}$ columns of $M_{d-1}(F_1, F_2)P$. Let

$$M = sA - tB = s \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} - t \begin{bmatrix} 0 \\ B_2 \end{bmatrix}$$

be the resulting square matrix pencil.

- (4) Perform a QR-factorization on A_1^* to compute a matrix Z whose columns form an orthonormal basis for the nullspace of A_1 . (In practice, Z is not formed explicitly but is represented as a product of Householder reflections, as in [11].)
- (5) Solve the smaller generalized eigenvalue problem $sA_2Z\mathbf{y} = tB_2Z\mathbf{y}$ using the QZ-algorithm, computing both eigenvalues and eigenvectors.
- (6) For each eigenvector \mathbf{y} computed in the previous step, compute $\mathbf{x} = Z\mathbf{y}$. Find the maximum (in absolute value) of the components of \mathbf{x} corresponding to the monomials $\{x^d, y^d, z^d\}$. If it is w^d where w is one of $\{x, y, z\}$, then let x_*, y_*, z_* be the components of \mathbf{x} corresponding to $\{w^{d-1}x, w^{d-1}y, w^{d-1}z\}$. If we want a solution to $f_1(x, y) = f_2(x, y) = 0$, we compute $(x_*/z_*, y_*/z_*)$.

Example 4. If we apply the above algorithm to (5) used in Example 3, then the F_3 -rows labeled with monomials z^3, xy^2, y^3, x^2z are dropped. The six roots are computed in Matlab 6.1 using this algorithm in double-precision arithmetic with a maximum residual of about $3.3 \cdot 10^{-13}$. In contrast, if we use the Macaulay choices for the dropped rows, the residuals are $O(1)$ for each of the computed roots.

6. WHEN DOES THIS WORK?

In the previous section we gave an algorithm for computing the common roots of two polynomials in two variables or two homogeneous polynomials in three variables. Clearly, it is not always going to work. For example, if there is a multiple eigenvalue, then the computed eigenvector is probably not going to be of the form (10), so we cannot determine the root from the eigenvector. The goal of this section is to identify the cases in which the algorithm is going to fail, assuming the computation is done in exact arithmetic.

We show (assuming exact arithmetic) that failure occurs only if

- (1) the polynomials F_1 and F_2 have a common factor,
- (2) the Jacobian at the root is zero, or
- (3) we picked a bad swepline.

We also show that if (1) and (2) do not hold, then the set of bad sweplines is of measure zero. This suggests that in floating-point arithmetic failure only occurs if the polynomial system is ill conditioned, because of a near common factor or the root under consideration is ill conditioned, or the random swepline is chosen from a small bad set. Floating-point behavior is more carefully analyzed in Section 8.

First we need to be able to choose rows to drop so that the extraneous factor is nonzero. The only problem here would be that $M_{d-1}(F_1, F_2)$ does not have a full row-rank, which implies that $M_d(F_1, F_2)$, the top part of M , has a left nullvector. But this is equivalent to F_1 and F_2 having a common factor.

When Macaulay [20] proves that a zero resultant implies that there is a common root, he is assuming generic coefficients. It is interesting to note that Cayley and Sylvester used generic reasoning, while some of their contemporaries had started to recognize its shortcomings (see [14]). Macaulay, building upon the works of Cayley and Sylvester, follows this tradition, but seems to be more aware that he is making these assumptions.

Macaulay's proof can be stated in the language of modern linear algebra and for the case $n = 2$ as follows. Let \mathbf{x} be a right nullvector for the matrix M we get after

dropping rows (the way Macaulay did it). Since any row of M_d can be written as a linear combination of the rows of M , \mathbf{x} is also a nullvector for M_d . The matrix $M_{d-1} = M_{d-1}(F_1, F_2, F_3)$ is generically of rank $\binom{d+1}{2} - 1$; i.e., one less than the number of columns. This means that M_{d-1} has (up to multiplication by a scalar) a unique right nullvector. But we can find three copies of M_{d-1} within M_d , and this forces \mathbf{x} to have the form (10).

Example 5. For the polynomials in Example 2, $M_{d-1}(F_1, F_2, F_3)$ is the following 10×10 matrix:

$$\begin{array}{c}
 x^3 \quad x^2y \quad x^2z \quad xy^2 \quad xyz \quad xz^2 \quad y^3 \quad y^2z \quad yz^2 \quad z^3 \\
 \begin{array}{l}
 1 \\
 x \\
 y \\
 z \\
 x^2 \\
 xy \\
 xz \\
 y^2 \\
 yz \\
 z^2
 \end{array}
 \begin{bmatrix}
 a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 & a_9 & a_{10} \\
 b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & & & & \\
 & b_1 & & b_2 & b_3 & & b_4 & b_5 & b_6 & \\
 & & b_1 & & b_2 & b_3 & & & b_4 & b_5 & b_6 \\
 u_1 & u_2 & u_3 & & & & & & & \\
 & u_1 & & u_2 & u_3 & & & & & \\
 & & u_1 & & u_2 & u_3 & & & & \\
 & & & u_1 & & & u_2 & u_3 & & \\
 & & & & u_1 & & & u_2 & u_3 & \\
 & & & & & u_1 & & & u_2 & u_3
 \end{bmatrix}
 \end{array}$$

We can find three copies of this matrix inside the matrix $M_d(F_1, F_2, F_3)$ given in Example 2. For example, if we multiply all the row and column labels above by x , then the submatrix of $M_d(F_1, F_2, F_3)$ taking rows and columns with these modified labels equals $M_{d-1}(F_1, F_2, F_3)$.

We now begin our own analysis of the exact-arithmetic case without any assumption about generic coefficients in F_1, F_2 . Since the polynomial F_3 is linear, it is easy to eliminate one of the variables. For example, if $u_1 \neq 0$, we can eliminate x from the other two polynomials by making the substitution $x = -(u_2y + u_3z)/u_1$. This results in two homogeneous polynomials in y and z :

$$\begin{aligned}
 F_1^{\text{sub}}(y, z) &= F_1(-(u_2y + u_3z)/u_1, y, z), \\
 F_2^{\text{sub}}(y, z) &= F_2(-(u_2y + u_3z)/u_1, y, z).
 \end{aligned}$$

Now we could consider the Sylvester type matrix $M_t(F_1^{\text{sub}}, F_2^{\text{sub}})$, instead of the matrix $M_t(F_1, F_2, F_3)$, where t is an integer. These two matrices are related:

Lemma 4. *Let $t \geq \max(d_1, d_2)$ be an integer. If $F_3 = u_1x + u_2y + u_3z$ with $u_1 \neq 0$, then the matrix $M_t(F_1^{\text{sub}}, F_2^{\text{sub}})$ is the Schur complement of the block with u_1 on the diagonal in $M_t(F_1, F_2, F_3)$ after dropping rows corresponding to rF_i , where r is a monomial divisible by x and $i = 1, 2$.*

Proof. Write

$$M_t(F_1, F_2, F_3) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix},$$

where M_{21} is the block with u_1 on the diagonal. The Schur complement of M_{21} in $M_t(F_1, F_2, F_3)$ is

$$M_t^{\text{sub}} = M_{12} - M_{11}M_{21}^{-1}M_{22}.$$

The columns of this matrix correspond to monomials in y and z of degree t , and the rows correspond to polynomials of the type rF_i^{sub} , where r is a monomial in x, y, z of degree $t - d_i$ for $i = 1, 2$.

To see this, note that if $x = -(u_2y + u_3z)/u_1$, then

$$M_{21} \begin{bmatrix} x^t \\ x^{t-1}y \\ \vdots \\ xz^{t-1} \end{bmatrix} + M_{22} \begin{bmatrix} y^t \\ y^{t-1}z \\ \vdots \\ z^t \end{bmatrix} = 0$$

and therefore

$$\begin{bmatrix} x^t \\ x^{t-1}y \\ \vdots \\ xz^{t-1} \end{bmatrix} = -M_{21}^{-1}M_{22} \begin{bmatrix} y^t \\ y^{t-1}z \\ \vdots \\ z^t \end{bmatrix}.$$

In other words, the rows of the matrix $-M_{21}^{-1}M_{22}$ correspond to the monomials in x, y, z of degree t divisible by x , written in terms of the monomials in only y and z .

A row of M_t^{sub} whose monomial is divisible by x is a linear combination of the rows whose monomials are independent of x , because x is a linear combination of y and z . And if we drop the rows whose monomials depend on x we get $M_t(F_1^{\text{sub}}, F_2^{\text{sub}})$. Alternatively, we could have first dropped the F_1 - and F_2 -rows whose monomials depend on x and then computed the Schur complement. \square

Suppose (x_*, y_*, z_*) is a common root of F_1, F_2, F_3 . We want to determine when $M_{d-1}(F_1, F_2, F_3)$ has rank one less than the number of columns, because then we know that

$$[x_*^d, x_*^{d-1}y_*, x_*^{d-1}z_*, \dots, z_*^d]^T$$

is (up to multiplication by a scalar) the unique right nullvector for $M_d(F_1, F_2, F_3)$.

Rotate the coordinates (x, y, z) so that the root will be on the z -axis. Let F_1^0, F_2^0, F_3^0 be the polynomials in these new coordinates. Note that the highest z coefficients of F_1^0, F_2^0, F_3^0 are all necessarily 0 in order for $F_3(0, 0, 1) = 0$. We will show in Section 7 that for any positive integer t ,

$$M_t(F_1^0, F_2^0, F_3^0) = UM_t(F_1, F_2, F_3)V,$$

where U and V are matrices with determinant equal to 1. Now $(0, 0, 1)$ is a common root of F_1^0, F_2^0, F_3^0 , so $[0, \dots, 0, 1]^T$ is a right nullvector for $M_t(F_1^0, F_2^0, F_3^0)$, and therefore the last column of this matrix is zero. Let M_t^0 denote $M_t(F_1^0, F_2^0, F_3^0)$ with the last column deleted. Since U and V are nonsingular, the matrices M_t^0 and $M_t(F_1, F_2, F_3)$ have the same rank.

The matrix M_{d-1}^0 has $\binom{d_1-1}{2} + \binom{d_2-1}{2} + \binom{d_3-1}{2}$ more rows than columns, and its rows are related by the same number of linear equations given by the rows of

$$(12) \quad H_{d-1}(F_1^0, F_2^0, F_3^0) = \begin{bmatrix} -M_{d-d_1-1}(F_2^0) & M_{d-d_2-1}(F_1^0) & 0 \\ -M_{d-d_1-1}(F_3^0) & 0 & M_{d-d_3-1}(F_1^0) \\ 0 & -M_{d-d_2-1}(F_3^0) & M_{d-d_3-1}(F_2^0) \end{bmatrix}.$$

For general symbolic coefficients the rows of this matrix are linearly independent, so we can apply Theorem 2 to M_{d-1}^0 and $H_{d-1}(F_1^0, F_2^0, F_3^0)$. The following theorem is stated for general d_1, d_2, d_3 , even though the rest of the section applies only to the special case of u -resultants, i.e., when $d_3 = 1$.

Theorem 5. *There exists an integer polynomial R_0 in the coefficients of F_1^0, F_2^0, F_3^0 such that*

- (1) R_0 divides all the $m_0 \times m_0$ subdeterminants of M_{d-1}^0 , where m_0 is the number of columns of M_{d-1}^0 . If D is such a subdeterminant, then $D = SR_0$, where S is (up to a sign) the determinant of the matrix we get by taking the columns of $H_{d-1}(F_1^0, F_2^0, F_3^0)$ corresponding to the rows not in D .
- (2) R_0 is homogeneous of degrees $d_2d_3 - 1, d_1d_3 - 1, d_1d_2 - 1$ in the coefficients of F_1^0, F_2^0, F_3^0 , respectively.

The proof of this theorem is omitted, but follows directly from the existence of the toric resultant for polynomials with the highest z power omitted; see Chapter 8 of [10]. The degree bounds follow by explicitly evaluating the relevant mixed volume. A more explicit proof is given in [15]. This quantity R_0 is also related to a residual resultant [2] and the subresultant [4].

We apply Lemma 4 to M_{d-1}^0 . We can write

$$F_3^0(x, y, z) = vx - uy,$$

where (u, v) points in the direction in which the sweepline goes through $(0, 0)$ in the xy plane. Note that these quantities v, u depend on the initial random numbers α_i 's and β_i 's as well as on the actual root (since the rotation to produce F_3^0 depends on the root). We will treat them for now as new independent parameters and investigate their distribution in Section 7.

With this choice of u, v , any point (x, y, z) satisfying $F_3^0(x, y, z) = 0$ is of the form (uw, vw, z) , for some w and z . Let

$$\begin{aligned} F_1^{u,v}(w, z) &= F_1^0(uw, vw, z)/w, \\ F_2^{u,v}(w, z) &= F_2^0(uw, vw, z)/w. \end{aligned}$$

(All the terms of $F_i^0(uw, vw, z)$ are divisible by w because $(0, 0, 1)$ is a root of F_i^0 .) Write

$$\begin{aligned} F_1^{u,v}(w, z) &= p_{d_1}w^{d_1-1} + \dots + p_2wz^{d_1-2} + p_1z^{d_1-1}, \\ F_2^{u,v}(w, z) &= q_{d_2}w^{d_2-1} + \dots + q_2wz^{d_2-2} + q_1z^{d_2-1}. \end{aligned}$$

Then p_j and q_j are homogeneous polynomials in u and v of degree j .

Theorem 6. *The quantity R_0 is the Sylvester resultant of $F_1^{u,v}$ and $F_2^{u,v}$,*

$$\begin{aligned} R_0 &= \det(M_{d-2}(F_1^{u,v}, F_2^{u,v})) \\ &= \left| \begin{array}{cccc} pd_1 & pd_{1-1} & \dots & p_1 \\ & \ddots & \ddots & \ddots \\ & & pd_1 & pd_{1-1} & \dots & p_1 \\ qd_2 & qd_{2-1} & \dots & q_1 \\ & \ddots & \ddots & \ddots \\ & & qd_2 & qd_{2-1} & \dots & q_1 \end{array} \right| \left. \begin{array}{l} \vphantom{\left| \right.} \\ \vphantom{\left| \right.} \\ \vphantom{\left| \right.} \\ \vphantom{\left| \right.} \\ \vphantom{\left| \right.} \end{array} \right\} \begin{array}{l} d_2 - 1 \text{ rows,} \\ \\ \\ d_1 - 1 \text{ rows.} \end{array} \end{aligned}$$

Proof. The number of F_1 - and F_2 -rows in M_{d-1}^0 whose monomials are divisible by x is $\binom{d_1-1}{2} + \binom{d_2-1}{2}$, so if we drop these rows we get a square matrix \tilde{M}_{d-1}^0 . Write

$$\tilde{M}_{d-1}^0 = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix},$$

where M_{21} is the block with v on the diagonal. Then by Lemma 4

$$\begin{aligned} M_{12} - M_{11}M_{21}^{-1}M_{22} &= M_{d-1}^0(F_1^0(uy/v, y, z), F_2^0(uy/v, y, z)) \\ &= M_{d-2}(F_1^{u/v,1}, F_2^{u/v,1}) \\ &= \Delta_1 M_{d-2}(F_1^{u,v}, F_2^{u,v}) \Delta_2^{-1}, \end{aligned}$$

where the 0 superscript means we delete the last column and where

$$\begin{aligned} \Delta_1 &= \text{diag}([v^{d_2-2}, \dots, v, 1, v^{d_1-2}, \dots, v, 1]), \\ \Delta_2 &= \text{diag}([v^{d-1}, v^{d-2}, \dots, v]). \end{aligned}$$

Since \tilde{M}_{d-1}^0 is square

$$\begin{aligned} \det(\tilde{M}_{d-1}^0) &= \det(M_{21}) \det(M_{12} - M_{11}M_{21}^{-1}M_{22}) \\ &= v^k \det(M_{d-1}^0(F_1^{u,v}, F_2^{u,v})), \end{aligned}$$

where

$$k = \binom{d_1+d_2-1}{2} + \binom{d_2-1}{2} + \binom{d_1-1}{2} - (d_1+d_2-1) = \binom{d_1-1}{2} + \binom{d_2-1}{2}.$$

But we also know that $\det(\tilde{M}_{d-1}^0) = S_0 R_0$, where S_0 is up to a sign the determinant of the submatrix of

$$\begin{bmatrix} M_{d_2-2}(F_3^0) & 0 \\ 0 & M_{d_1-2}(F_3^0) \end{bmatrix},$$

taking only the columns whose monomials are divisible by x . This submatrix is upper triangular with all the diagonal entries equal to v , so $S_0 = \pm v^k$. \square

Example 6. Continuing Example 5, we have that

$$M_{d-1}^0 = \left[\begin{array}{cccccc|ccc} a'_1 & a'_2 & a'_3 & a'_4 & a'_5 & a'_6 & a'_7 & a'_8 & a'_9 \\ b'_1 & b'_2 & b'_3 & b'_4 & b'_5 & & & & \\ & b'_1 & & b'_2 & b'_3 & & b'_4 & b'_5 & \\ & & b'_1 & & b'_2 & b'_3 & b'_4 & b'_4 & b'_5 \\ \hline v & -u & & & & & & & \\ & v & & -u & & & & & \\ & & v & & -u & & & & \\ & & & v & & & -u & & \\ & & & & v & & & -u & \\ & & & & & v & & & -u \end{array} \right],$$

where a'_i and b'_j are the coefficients of the polynomials F_1^0 and F_2^0 . Also,

$$H_{d-1}^0(F_1^0, F_2^0, F_3^0) = [0 \quad -v \quad u \quad 0 \quad b'_1 \quad b'_2 \quad b'_3 \quad b'_4 \quad b'_5 \quad 0].$$

The Schur complement of the lower left block in M_{d-1}^0 is

$$\begin{bmatrix} a'_1 \frac{u^3}{v^3} + a'_2 \frac{u^2}{v^2} + a'_4 \frac{u}{v} + a'_7 & a'_3 \frac{u^2}{v^2} + a'_5 \frac{u}{v} + a'_8 & a'_6 \frac{u}{v} + a'_9 \\ b'_1 \frac{u^3}{v^3} + b'_2 \frac{u^2}{v^2} + b'_4 \frac{u}{v} & b'_3 \frac{u^2}{v^2} + b'_5 \frac{u}{v} & \\ b'_1 \frac{u^2}{v^2} + b'_2 \frac{u}{v} + b'_4 & b'_3 \frac{u}{v} + b'_5 & \\ b'_1 \frac{u^2}{v^2} + b'_2 \frac{u}{v} + b'_4 & b'_3 \frac{u}{v} + b'_5 & b'_3 \frac{u}{v} + b'_5 \end{bmatrix},$$

and if we delete the second row and multiply rows and columns with suitable powers of v , we get

$$(13) \quad R_0 = \begin{vmatrix} a'_1 u^3 + a'_2 u^2 v + a'_4 u v^2 + a'_7 v^3 & a'_3 u^2 + a'_5 u v + a'_8 v^2 & a'_6 u + a'_9 v \\ b'_1 u^2 + b'_2 u v + b'_4 v^2 & b'_3 u + b'_5 v & \\ & b'_1 u^2 + b'_2 u v + b'_4 v^2 & b'_3 u + b'_5 v \end{vmatrix}.$$

The quantity R_0 is a homogeneous polynomial in u and v , and (u, v) is the direction in which the swepline goes through the origin in the xy plane. A homogeneous polynomial in two variables is either identically zero or it has finitely many roots in \mathbb{P}^1 . Since we choose the swepline randomly, we are almost surely not going to hit a root of R_0 in the latter case. So we need to identify when $R_0 = 0$ for all u, v .

Theorem 7. $R_0 = 0$ for all u, v if and only if one of the following holds:

- (1) The Jacobian of F_1^0 and F_2^0 at $(0, 0, 1)$ is zero.
- (2) F_1^0 and F_2^0 have a common factor of degree at least 1.

Proof. By Theorem 6, R_0 is a Sylvester resultant and therefore vanishes if and only if the homogeneous polynomials $F_1^{u,v}(w, z)$ and $F_2^{u,v}(w, z)$ have a (homogeneous) common factor f_1 of degree at least 1. Note the Sylvester theorem requires divisions in the coefficients, so that this factor f_1 lies in $\mathbb{C}(u, v)[w, z]$. In other words, f_1 is a homogeneous polynomial in w and z and a rational function in u, v , i.e., f_1 has the form

$$f_1 = \alpha_0(u, v)z^k/\beta_0(u, v) + \alpha_1(u, v)z^{k-1}w/\beta_1(u, v) + \dots + \alpha_k(u, v)w^k/\beta_k(u, v),$$

where the α_i 's and β_i 's are polynomials. Thus, $F_1^{u,v}(w, z) = f_1 f_2$ and $F_2^{u,v}(w, z) = f_1 f_3$, where $f_2, f_3 \in \mathbb{C}(u, v)[w, z]$. Recall that $F_1^0(uw, vw, z)/w = F_1^{u,v}(w, z)$ and $F_2^0(uw, vw, z) = F_2^0(uw, vw, z)/w$; hence $F_1^0(uw, vw, z)/w = f_1 f_3$ and $F_2^0(uw, vw, z) = f_1 f_2$. There are now two cases to consider. The first is that there is a term in f_1 of degree at least 1 with respect to z . Substitute $w = 1, u = x, v = y$ to obtain $F_1^0(x, y, z) = f_1 f_2$ and $F_2^0(x, y, z) = f_1 f_3$, where now f_1, f_2, f_3 are rational in x, y and polynomial in z . Then case 2 of the theorem follows because the numerator of f_1 , a polynomial in z (and perhaps x, y) of degree at least 1, is the common factor that is case 2 of the theorem. The other case is that f_1 does not depend on z , in which case it must be a multiple of w . This means that $p_1 = q_1 = 0$ identically (else $F_1^{u,v}(w, z)$ or $F_2^{u,v}(w, z)$ could not be divisible by w). Tracing back the definition of p_1, q_1 shows that the Jacobian at the root is zero, i.e., $J^0(0, 0, 1) = 0$, where

$$J^0(x, y, z) = \begin{bmatrix} \frac{\partial F_1^0}{\partial x} & \frac{\partial F_1^0}{\partial y} & \frac{\partial F_1^0}{\partial z} \\ \frac{\partial F_2^0}{\partial x} & \frac{\partial F_2^0}{\partial y} & \frac{\partial F_2^0}{\partial z} \end{bmatrix}.$$

(For example, observe that the coefficients of u, v in the right-most column of the right-hand side of (13) are exactly the entries of the Jacobian at $(0, 0, 1)$.) The proof of the converse argument is similar. \square

This tells us that if the Jacobian at the root is nonzero and there is not a common factor, then we have a unique eigenvector (up to a scalar multiple, of course) except for only finitely many sweelines through the root. Those finite directions are choices of (u, v) that are roots of R_0 .

7. ROTATING COORDINATES

In last section we rotated the coordinates x, y, z so that the root we were analyzing would lie on the z -axis. The rotation was a central piece of the analysis and in some sense localized the coordinates at the root. Here we explore how this rotation affects the matrices $M_t(F_1, F_2, F_3)$. In the exact arithmetic analysis we only needed to know that the rank did not change, but for the floating-point analysis we will need a bound on the change in singular values. At the end of the section, we also study the relationship between u, v , the rotation, and the initial random parameters.

This rotation takes place in \mathbb{C}^3 . We split the rotation into two steps. First we rotate each complex coordinate so that the root has real coordinates, and then we rotate in the real (x, z) plane and the real (y, z) plane to move the root to the z -axis. Actually one of these five rotations is redundant (in particular one of the three complex coordinate rotations), because the root is defined up to a multiple of a complex number.

Suppose the root being analyzed is

$$(r_1 e^{i\theta_1}, r_2 e^{i\theta_2}, r_3 e^{i\theta_3}),$$

where r_j and θ_j are real numbers. Change coordinates by rotating each complex coordinate

$$x' = e^{-i\theta_1} x, \quad y' = e^{-i\theta_2} y, \quad z' = e^{-i\theta_1} z,$$

so that in the new coordinates the root is (r_1, r_2, r_3) . If F'_1, F'_2, F'_3 are the polynomials F_1, F_2, F_3 written in the new coordinates, then

$$M_t(F'_1, F'_2, F'_3) = \begin{bmatrix} \Lambda_{t-d_1} & & & \\ & \Lambda_{t-d_2} & & \\ & & \Lambda_{t-d_2} & \\ & & & \Lambda_t \end{bmatrix} M_t(F_1, F_2, F_3) \Lambda_t,$$

where

$$\Lambda_k = \text{diag}([e^{ik\theta_1}, e^{i((k-1)\theta_1+\theta_2)}, \dots, e^{ik\theta_3}]).$$

This is a unitary matrix; multiplying by a unitary matrix does not change the singular values.

Now we have a root with real coordinates. In \mathbb{R}^3 a rotation carrying one vector into the direction of another can be written as a combination of two rotations around the coordinate axis. This is a linear transformation, and if we apply it to \mathbb{C}^3 we can rotate a vector with real coordinates into the direction of the vector $[0, 0, 1]$.

Consider a rotation of the coordinates in the (y, z) plane:

$$\begin{aligned} x' &= x, \\ y' &= (\cos \theta)y - (\sin \theta)z, \\ z' &= (\sin \theta)y + (\cos \theta)z. \end{aligned}$$

We can write the monomials in y', z' of order k in terms of the monomials in y, z of order k . As an example take $k = 3$:

$$\begin{bmatrix} (y')^3 \\ (y')^2 z' \\ y' (z')^2 \\ (z')^3 \end{bmatrix} = \begin{bmatrix} c^3 & -3c^2 s & 3cs^2 & -s^3 \\ c^2 s & c^3 - 2cs^2 & -2c^2 s + s^3 & cs^2 \\ cs^2 & 2c^2 s - s^3 & c^3 - 2cs^2 & -c^2 s \\ s^3 & 3cs^2 & 3c^2 s & c^3 \end{bmatrix} \begin{bmatrix} y^3 \\ y^2 z \\ y z^2 \\ z^3 \end{bmatrix},$$

where $c = \cos \theta$ and $s = \sin \theta$. Let $U_{\theta,k}$ denote the monomial rotation matrix of order k (so the matrix above is $U_{\theta,3}$). In general, the i th row of $U_{\theta,k}$ consists of the coefficients of

$$(14) \quad (cy - sz)^{k-i+1}(sy + cz)^{i-1},$$

as a polynomial in y and z . It is not hard to verify that $\det(U_{\theta,k}) = (c^2 + s^2)^k = 1$. Also note that $U_{\theta,k}^{-1} = U_{-\theta,k}$.

Let $V_{\theta,k}^{y,z}$ denote the matrix that maps the monomials in x, y, z of order k into the monomials in x', y', z' of order k . If we use lexicographical order,

$$V_{\theta,k}^{y,z} = \begin{bmatrix} U_{\theta,k} & & & \\ & \ddots & & \\ & & U_{\theta,1} & \\ & & & 1 \end{bmatrix}.$$

Let F'_1, F'_2, F'_3 be the polynomials F_1, F_2, F_3 in the new coordinates x', y', z' . Then

$$(15) \quad M_t(F'_1, F'_2, F'_3) = W_{\theta,t}^{y,z} M_t(F_1, F_2, F_3) V_{-\theta,t}^{y,z},$$

where

$$W_{\theta,t}^{y,z} = \begin{bmatrix} V_{\theta,t-d_1}^{y,z} & & & \\ & V_{\theta,t-d_2}^{y,z} & & \\ & & V_{\theta,t-d_3}^{y,z} & \\ & & & \end{bmatrix}^T.$$

In our analysis in Section 8 we need to estimate the effect of the rotation on the singular values of $M_t(F_1, F_2, F_2)$. To simplify notation, write (15) as

$$M'_t = W M_t V.$$

We will use the max-min property of singular values (Theorem 8.6.1 of [11]), which gives the following formula for the k th biggest singular value:

$$\sigma_k(M_t) = \max_{\dim(\mathcal{S})=k} \min_{\mathbf{x} \in \mathcal{S}} \frac{\|M_t \mathbf{x}\|_2}{\|\mathbf{x}\|_2}.$$

So, since V and W are invertible,

$$\begin{aligned} \sigma_k(M'_t) &= \max_{\dim(\mathcal{S})=k} \min_{\mathbf{x} \in \mathcal{S}} \frac{\|W M_t V \mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \max_{\dim(\mathcal{S})=k} \min_{\mathbf{x} \in \mathcal{S}} \frac{\|W M_t \mathbf{x}\|_2}{\|V^{-1} \mathbf{x}\|_2} \\ &\leq \max_{\dim(\mathcal{S})=k} \min_{\mathbf{x} \in \mathcal{S}} \frac{\|W\|_2 \|M_t \mathbf{x}\|_2}{\|V\|_2^{-1} \|\mathbf{x}\|_2} = \|W\|_2 \|V\|_2 \sigma_k(M_t). \end{aligned}$$

For an upper bound on the 2-norm of $V = V_{-\theta,t}^{y,z}$, note that, since this matrix is block diagonal,

$$\|V\|_2 = \max_{0 \leq k \leq t} \|U_{\theta,k}\|_2.$$

Similarly, we get an upper bound for $\|W\|_2$ by taking the max over $0 \leq k \leq t - \min(d_1, d_2, d_3)$, so in particular $\|W\|_2 \leq \|V\|_2$. Now we only need to get an upper bound for $\|U_{\theta,k}\|_2$.

Since the entries in row i of $U_{\theta,k}$ are the coefficients of (14) as a polynomial in y and z , we get the following upper bound for the 1-norm of the row

$$\sum_{j=0}^k \binom{k}{j} |\cos^{k-j} \theta \sin^j \theta| = (|\cos \theta| + |\sin \theta|)^k$$

and either the first or the last row have a 1-norm equal to this. The ∞ -norm of a matrix is equal to the maximum 1-norm of a row, so we have

$$\|U_{\theta,k}\|_\infty = (|\cos \theta| + |\sin \theta|)^k,$$

and therefore

$$\|U_{\theta,k}\|_2 \leq \sqrt{k+1} \|U_{\theta,k}\|_\infty \leq \sqrt{k+1} 2^{k/2}.$$

So we have $\|W\|_2 \leq \|V\|_2 \leq \sqrt{t+1} 2^{t/2}$ and hence we showed that

$$\sigma_k(M_t(F'_1, F'_2, F'_3)) \leq (t+1)2^t \sigma_k(M_t(F_1, F_2, F_3)).$$

Rotating each complex coordinate so the root will have real coordinates does not change the singular values of $M_t(F_1, F_2, F_3)$. A rotation around a coordinate axis changes them by no more than a factor of $(t+1)2^t$, and we use two such rotations. Thus, we can change coordinates so that the root is $(0, 0, 1)$ and so that the change in singular values of $M_t(F_1, F_2, F_3)$ is bounded by a factor of $(t+1)^2 2^{2t}$.

The other effect of the rotation is the distribution of the parameters u, v appearing in the previous section. Note that u and v depend on α, β , and the eigenvalue (s, t) :

$$(16) \quad \begin{bmatrix} v \\ -u \\ 0 \end{bmatrix} = \bar{Q}(s\alpha - t\beta),$$

where \bar{Q} is the complex conjugate of the matrix for the coordinate rotation, i.e., the composition of the rotations and scalings introduced in this section. Since $\|\alpha\|_2 = \|\beta\|_2 = 1$, $\alpha^* \beta = 0$, and $|s|^2 + |t|^2 = 1$, and because Q is unitary, the coefficients u and v satisfy $|u|^2 + |v|^2 = 1$.

In the algorithm we choose α and β randomly from a uniform distribution over $\{\mathbf{z} \in \mathbb{C}^3 : \|\mathbf{z}\|_2 = 1\}$ and so that they are orthogonal. We want to argue that our random choice of sweepline causes (u, v) to be uniformly distributed over the set $\{\mathbf{w} \in \mathbb{C}^2 : \|\mathbf{w}\|_2 = 1\}$. The uniform distribution on the unit sphere is uniquely characterized by its invariance under arbitrary unitary transformation. Therefore, we need to check whether $[u; v]$ and $[\tilde{u}; \tilde{v}] = Z[u; v]$ have the same distribution for an arbitrary unitary Z . Observe that

$$\begin{bmatrix} \tilde{v} \\ -\tilde{u} \\ 0 \end{bmatrix} = \tilde{Z} \begin{bmatrix} v \\ -u \\ 0 \end{bmatrix},$$

where $\tilde{Z} = [Z(1, 2), -Z(2, 1), 0; -Z(1, 1), Z(1, 2), 0; 0, 0, 1]$, a unitary matrix. Combining this with (16) yields

$$\begin{bmatrix} \tilde{v} \\ -\tilde{u} \\ 0 \end{bmatrix} = \tilde{Z}\bar{Q}(s\alpha - t\beta) = \bar{Q}(s(\bar{Q}^* \tilde{Z} \bar{Q} \alpha) - t(\bar{Q}^* \tilde{Z} \bar{Q} \beta)).$$

Since $\bar{Q}^* \tilde{Z} \bar{Q}$ is unitary, the distribution of (α, β) is invariant under this transformation. Therefore, comparing the previous equation to (16) shows that $[u; v]$ and $[\tilde{u}; \tilde{v}]$ have the same distribution; hence, this distribution is uniform.

8. EIGENVECTOR SENSITIVITY

Now we are ready to analyze the accuracy of the root-finding algorithm. Our analysis will focus on the conditioning of the eigenvectors of the generalized eigenvalue problem $sA\mathbf{x} = tB\mathbf{x}$. The idea is that since the QZ-algorithm is backwards stable, it computes well-conditioned eigenvectors accurately. Note, however, that we are actually solving the reduced problem (9), but we will argue in subsection 8.6 that its conditioning is no worse than that of the bigger eigenvalue problem.

Suppose \mathbf{x} is the exact eigenvector corresponding to the root (x_*, y_*, z_*) , i.e.,

$$\mathbf{x} = [x_*^d, x_*^{d-1}y_*, x_*^{d-1}z_*, \dots, z_*^d]^T.$$

If this eigenvector is well conditioned, then the computed eigenvector $\hat{\mathbf{x}}$ will be close to \mathbf{x} (if they are scaled properly). In the algorithm we compute the coordinates of the root as follows. First we pick the biggest (in absolute value) of the components of $\hat{\mathbf{x}}$ corresponding to the monomials x^d, y^d, z^d . If that is w^d , where $w \in \{x, y, z\}$, then we let $\hat{x}_*, \hat{y}_*, \hat{z}_*$ be the values of the components of $\hat{\mathbf{x}}$ corresponding to the monomials $w^{d-1}x, w^{d-1}y, w^{d-1}z$.

The point here is that if $\hat{\mathbf{x}}$ were exact, we could pick components corresponding to any set of monomials of the type $\{rx, ry, rz\}$, where r is a monomial of degree $d-1$, as the computed root. But $\hat{\mathbf{x}}$ has errors and the above choice of a set of monomials picks the set of this type corresponding to the biggest components and therefore with the smallest relative error. More precisely, w^d will be the largest entry of the eigenvector. So the relative error in the selected subvector $\{w^{d-1}x, w^{d-1}y, w^{d-1}z\}$ is at most $\sqrt{\binom{d+2}{2}}$ times the relative error in $\hat{\mathbf{x}}$ (if the errors are measured in 2-norm).

8.1. General theory. The eigenvalue (s, t) of (11) may be regarded as lying in the projective space \mathbb{P}^1 . We define a metric for \mathbb{P}^1 by

$$(17) \quad \rho((s_1, t_1), (s_2, t_2)) = \frac{|s_1t_2 - t_1s_2|}{\sqrt{|s_1|^2 + |t_1|^2} \sqrt{|s_2|^2 + |t_2|^2}}.$$

This equals the chordal metric,

$$\text{chord}(\lambda_1, \lambda_2) = \frac{|\lambda_1 - \lambda_2|}{\sqrt{1 + |\lambda_1|^2} \sqrt{1 + |\lambda_2|^2}},$$

for the eigenvalues $\lambda_1 = t_1/s_1$ and $\lambda_2 = t_2/s_2$. Stewart [27] used this metric to estimate eigenvalue sensitivity, showing that if λ_ϵ is the eigenvalue of the perturbed pencil $(A + E) - \lambda(B + F)$ with $\|E\|_F \leq \epsilon, \|F\|_F \leq \epsilon$, then

$$\text{chord}(\lambda_\epsilon, \lambda) \leq \frac{\epsilon}{\sqrt{|\mathbf{y}^*A\mathbf{x}|^2 + |\mathbf{y}^*B\mathbf{x}|^2}} + O(\epsilon^2),$$

where \mathbf{y} and \mathbf{x} are the left and right eigenvectors of the unit 2-norm corresponding to λ , i.e.,

$$\mathbf{y}^*A = \lambda\mathbf{y}^*B, \quad A\mathbf{x} = \lambda B\mathbf{x}, \quad \|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = 1.$$

Suppose (s, t) is a simple eigenvalue and \mathbf{x} is a corresponding eigenvector (with $\|\mathbf{x}\|_2 = 1$) for the eigenvalue problem (11), and that for a small perturbation we have

$$(18) \quad (s + \Delta s)(A + E)(\mathbf{x} + \Delta\mathbf{x}) = (t + \Delta t)(B + F)(\mathbf{x} + \Delta\mathbf{x}).$$

Let δ be the second smallest singular value of the matrix $sA - tB$ (the smallest is zero). Theorem 8.6.5 of [11] implies that if

$$\|\Delta sA - \Delta tB + sE - tF\|_F \leq \frac{\delta}{4},$$

then

$$\|\Delta \mathbf{x}\|_2 \leq 4 \frac{\|\Delta sA - \Delta tB + sE - tF\|_F}{\delta}.$$

Since $|sE_{ij} - tF_{ij}|^2 \leq (|s|^2 + |t|^2)(|E_{ij}|^2 + |F_{ij}|^2)$, we have

$$\begin{aligned} \|\Delta sA - \Delta tB + sE - tF\|_F &\leq \sqrt{|\Delta s|^2 + |\Delta t|^2} \sqrt{\|A\|_F^2 + \|B\|_F^2} \\ &\quad + \sqrt{|s|^2 + |t|^2} \sqrt{\|E\|_F^2 + \|F\|_F^2}. \end{aligned}$$

Note that the choice of Δs and Δt in (18) is not unique, because we can multiply by a nonzero scalar. In other words, we can choose a representative in \mathbb{C}^2 for the perturbed eigenvalue in \mathbb{P}^1 . Let (\hat{s}, \hat{t}) be one such representative. We want to choose Δs and Δt that satisfy $\hat{s}(t + \Delta t) = \hat{t}(s + \Delta s)$ and minimize $\sqrt{|\Delta s|^2 + |\Delta t|^2}$. The solution of this least squares problem yields Δs and Δt such that

$$\sqrt{|\Delta s|^2 + |\Delta t|^2} = \frac{|s\hat{t} - t\hat{s}|}{\sqrt{|\hat{s}|^2 + |\hat{t}|^2}}$$

and then

$$\begin{aligned} &\|\Delta sA - \Delta tB + sE - tF\|_F \\ &\leq \sqrt{|s|^2 + |t|^2} \left(\sqrt{\|E\|_F^2 + \|F\|_F^2} + \rho((s, t), (\hat{s}, \hat{t})) \sqrt{\|A\|_F^2 + \|B\|_F^2} \right) \\ &\leq \sqrt{|s|^2 + |t|^2} \left(1 + \frac{\sqrt{\|A\|_F^2 + \|B\|_F^2}}{\sqrt{|\mathbf{y}^* A \mathbf{x}|^2 + |\mathbf{y}^* B \mathbf{x}|^2}} \right) \sqrt{\|E\|_F^2 + \|F\|_F^2} \\ &\quad + O(\|E\|_F^2 + \|F\|_F^2), \end{aligned}$$

where \mathbf{y} is the left eigenvector and as before we are assuming $\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = 1$.

Theorem 8. *Let \mathbf{x} be an eigenvector of the pencil (A, B) corresponding to a simple eigenvalue (s, t) . Let \mathbf{y} be the corresponding left eigenvector and assume $\|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = 1$. Let δ be the second smallest singular value of $sA - tB$. Then the perturbed pencil $(A + E, B + F)$ has an eigenvector $\mathbf{x} + \Delta \mathbf{x}$ with*

$$\begin{aligned} \|\Delta \mathbf{x}\|_2 &\leq \frac{4\sqrt{|s|^2 + |t|^2}}{\delta} \left(1 + \frac{\sqrt{\|A\|_F^2 + \|B\|_F^2}}{\sqrt{|\mathbf{y}^* A \mathbf{x}|^2 + |\mathbf{y}^* B \mathbf{x}|^2}} \right) \sqrt{\|E\|_F^2 + \|F\|_F^2} \\ &\quad + O(\|E\|_F^2 + \|F\|_F^2), \end{aligned}$$

assuming the perturbation is small enough.

We are viewing (s, t) as a point in \mathbb{P}^1 and we can choose a representative in \mathbb{C}^2 such that $|s|^2 + |t|^2 = 1$. For the rest of our analysis we are assuming this has been done. Note that δ depends on s and t and that rescaling (s, t) does not change the perturbation estimate above.

8.2. Eigenvalue conditioning in terms of the polynomials. So far the analysis is valid for any generalized eigenvalue problem, as long as the eigenvalue in question is simple. Now we will specialize this result for the matrices A and B in our algorithm. Recall that

$$\begin{aligned} F_3(x, y, z) &= s(\alpha_1x + \alpha_2y + \alpha_3z) - t(\beta_1x + \beta_2y + \beta_3z) \\ &= sL_\alpha(x, y, z) - tL_\beta(x, y, z). \end{aligned}$$

We normalized the coefficients of the polynomials $F_1, F_2, L_\alpha, L_\beta$ so that the coefficient vectors had a 2-norm equal to 1. Then

$$\|A\|_F = \sqrt{m} \quad \text{and} \quad \|B\|_F = \sqrt{d_1d_2},$$

where $m = \binom{d+2}{2}$ is the size of the matrix pencil.

The right eigenvector is

$$\mathbf{x} = [x_*^d, x_*^{d-1}y_*, x_*^{d-1}z_*, \dots, z_*^d]^T,$$

where (x_*, y_*, z_*) is a root of the homogeneous polynomials F_1, F_2 , and F_3 . The left eigenvector \mathbf{y} consists of the complex conjugates of the coefficients of the homogeneous polynomials P_1, P_2, P_3 , of degrees $d_2 - 1, d_1 - 1, d_1 + d_2 - 2$, respectively, not all zero, satisfying

$$(19) \quad P_1F_1 + P_2F_2 + P_3F_3 = 0,$$

where we have dropped the appropriate monomials from P_3 (i.e., P_3 does not have the monomials that correspond to the rows we dropped). We assume that P_1, P_2, P_3 and (x_*, y_*, z_*) have been normalized so that the 2-norm of both left and right eigenvectors is 1.

Since (x_*, y_*, z_*) is a root of F_1 and F_2 we get

$$\mathbf{y}^*A\mathbf{x} = P_3(x_*, y_*, z_*)L_\alpha(x_*, y_*, z_*)$$

and

$$\mathbf{y}^*B\mathbf{x} = P_3(x_*, y_*, z_*)L_\beta(x_*, y_*, z_*).$$

So we have the conditioning of the eigenvalue (s, t) in terms of the root and the polynomials

$$\frac{\sqrt{\|A\|_F^2 + \|B\|_F^2}}{\sqrt{|\mathbf{y}^*A\mathbf{x}|^2 + |\mathbf{y}^*B\mathbf{x}|^2}} = \frac{\sqrt{m + d_1d_2}}{|P_3(x_*, y_*, z_*)|\sqrt{|L_\alpha(x_*, y_*, z_*)|^2 + |L_\beta(x_*, y_*, z_*)|^2}}.$$

Let $\mathbf{v} = [x_*, y_*, z_*]^*$. Recall that in the algorithm the random vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are chosen to be orthogonal and of unit 2-norm. Then

$$(20) \quad \sqrt{|L_\alpha(x_*, y_*, z_*)|^2 + |L_\beta(x_*, y_*, z_*)|^2} = \sqrt{|\mathbf{v}^*\boldsymbol{\alpha}|^2 + |\mathbf{v}^*\boldsymbol{\beta}|^2}$$

measures the length of the projection of the vector \mathbf{v} onto the plane spanned by $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. This quantity is small only if \mathbf{v} is close to being orthogonal to the plane, and since the distribution of the direction normal to the random plane is uniform, there is a low probability if this happening.

8.3. **Estimating** $P_3(x_*, y_*, z_*)$. Differentiating equation (19) with respect to x gives

$$\frac{\partial P_1}{\partial x} F_1 + P_1 \frac{\partial F_1}{\partial x} + \frac{\partial P_2}{\partial x} F_2 + P_2 \frac{\partial F_2}{\partial x} + \frac{\partial P_3}{\partial x} F_3 + P_3(s\alpha_1 - t\beta_1) = 0.$$

Evaluate this equation and the two similar equations we get by differentiating with respect to y and z at the root (x_*, y_*, z_*) , and we get

$$(21) \quad P_3(x_*, y_*, z_*)(s\alpha - t\beta) = -[P_1(x_*, y_*, z_*), P_2(x_*, y_*, z_*)]J(x_*, y_*, z_*),$$

where J is the Jacobian

$$J = \begin{bmatrix} \frac{\partial F_1}{\partial x} & \frac{\partial F_1}{\partial y} & \frac{\partial F_1}{\partial z} \\ \frac{\partial F_2}{\partial x} & \frac{\partial F_2}{\partial y} & \frac{\partial F_2}{\partial z} \end{bmatrix}.$$

This shows that the conditioning of the eigenvalue is, as we would expect, dependent on the Jacobian at the root.

Now rotate the coordinates, as we did in Section 6, so that the root will be $(0, 0, 1)$. Let F_i^0 and P_i^0 be the polynomials F_i and P_i in the new coordinates, so that

$$F_i^0(0, 0, 1) = 0 \quad \text{and} \quad P_i^0(0, 0, 1) = P_i(x_*, y_*, z_*), \quad i = 1, 2, 3.$$

Note that $P_i^0(0, 0, 1)$ is the coefficient of z^{d-d_i} in P_i^0 . So in the transformed coordinates the values $P_i(x_*, y_*, z_*)$ have been isolated as coefficients and this will help us in obtaining a bound for these values.

As in Section 6, we write

$$F_3^0(x, y, z) = vx - uy.$$

Let N_d be the submatrix of $M_d(F_1^0, F_2^0, F_3^0)$, where we delete the rows corresponding to $z^{d-d_i}F_i^0$, $i = 1, 2, 3$, and the columns corresponding to xz^{d-1} , yz^{d-1} , z^d . Recall that the rows of $M_d = M_d(F_1, F_2, F_3)$ are related by $H_d M_d = 0$, where $H_d = H_d(F_1, F_2, F_3)$ is given by equation (3). After rotating coordinates we get that

$$H_d(F_1^0, F_2^0, F_3^0)M_d(F_1^0, F_2^0, F_3^0) = 0.$$

Notice that the columns of $H_d(F_1^0, F_2^0, F_3^0)$ corresponding to the rows of the matrix $M_d(F_1^0, F_2^0, F_3^0)$ we dropped when constructing N_d are zero. So $H_d^0 N_d = 0$, where

$$H_d^0 = H_d^0(F_1^0, F_2^0, F_3^0) = \begin{bmatrix} -M_{d-d_1}^0(F_3^0) & 0 & M_{d-1}^0(F_1^0) \\ 0 & -M_{d-d_2}^0(F_3^0) & M_{d-1}^0(F_2^0) \end{bmatrix}.$$

The 0 superscript in $M_k^0(\cdot)$ means that we drop the last column (corresponding to z^k) from the matrix $M_k(\cdot)$.

Example 7. For our continuing example, the matrix N_d is

$$\begin{matrix}
 & x^4 & x^3y & x^3z & x^2y^2 & x^2yz & x^2z^2 & xy^3 & xy^2z & xyz^2 & y^4 & y^3z & y^2z^2 \\
 x & a'_1 & a'_2 & a'_3 & a'_4 & a'_5 & a'_6 & a'_7 & a'_8 & a'_9 & & & \\
 y & & a'_1 & & a'_2 & a'_3 & & a'_4 & a'_5 & a'_6 & a'_7 & a'_8 & a'_9 \\
 x^2 & b'_1 & b'_2 & b'_3 & b'_4 & b'_5 & & & & & & & \\
 xy & & b'_1 & & b'_2 & b'_3 & & b'_4 & b'_5 & & & & \\
 xz & & & b'_1 & & b'_2 & b'_3 & & b'_4 & b'_5 & & & \\
 y^2 & & & & b'_1 & & & b'_2 & b'_3 & & b'_4 & b'_5 & \\
 yz & & & & & b'_1 & & & b'_2 & b'_3 & & b'_4 & b'_5 \\
 x^3 & v & -u & & & & & & & & & & \\
 x^2y & & v & & -u & & & & & & & & \\
 x^2z & & & v & & -u & & & & & & & \\
 xy^2 & & & & v & & & -u & & & & & \\
 xyz & & & & & v & & & -u & & & & \\
 xz^2 & & & & & & v & & & -u & & & \\
 y^3 & & & & & & & v & & & -u & & \\
 y^2z & & & & & & & & v & & & -u & \\
 yz^2 & & & & & & & & & v & & & -u
 \end{matrix}$$

and the matrix $H_d^0 = H_d^0(F_1^0, F_2^0, F_3^0)$ is

$$\begin{matrix}
 & x & y & x^2 & xy & xz & y^2 & yz & x^3 & x^2y & x^2z & xy^2 & xyz & xz^2 & y^3 & y^2z & yz^2 \\
 1 & -v & u & & & & & & a'_1 & a'_2 & a'_3 & a'_4 & a'_5 & a'_6 & a'_7 & a'_8 & a'_9 \\
 x & & & -v & u & & & & b'_1 & b'_2 & b'_3 & b'_4 & b'_5 & & & & \\
 y & & & & -v & u & u & & & b'_1 & & b'_2 & b'_3 & & b'_4 & b'_5 & \\
 z & & & & & -v & u & & & & b'_1 & & b'_2 & b'_3 & & b'_4 & b'_5
 \end{matrix}$$

Proposition 9. We have a lower bound for $|P_3(x_*, y_*, z_*)|$ in terms of the smallest singular values of N_d and the Jacobian

$$|P_3(x_*, y_*, z_*)| \geq \frac{\sigma_{\min}(N_d) \cdot \sigma_{\min}(M_{d-1}(F_1, F_2)) \cdot \sigma_{\min}(J(x_*, y_*, z_*))}{c(d_1, d_2)},$$

where $c(d_1, d_2)$ is a function of d_1 and d_2 .

Proof. Before starting the proof, recall the following facts. Take any $A \in \mathbb{C}^{m \times n}$ and $\mathbf{x} \in \mathbb{C}^n$. If $m \geq n$, then

$$(22) \quad \|A\mathbf{x}\| \geq \sigma_{\min}(A) \cdot \|\mathbf{x}\|,$$

where σ_{\min} refers to the n th (last) singular value. On the other hand, if $m < n$, let H be an $n \times (n - m)$ matrix of full column rank such that $AH = 0$. Then

$$(23) \quad \|A\mathbf{x}\| \geq \sigma_{\min}(A) \cdot \text{dist}(\text{Range}(H), \mathbf{x}),$$

where σ_{\min} now refers to the m th (last) singular value. Both of these facts are proved using the singular value decomposition.

Since $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are orthogonal and have a unit 2-norm,

$$\|s\boldsymbol{\alpha} - t\boldsymbol{\beta}\|_2 = \sqrt{|s|^2 + |t|^2} = 1.$$

So by (21),

$$\begin{aligned}
 |P_3(x_*, y_*, z_*)| &= \|[P_1(x_*, y_*, z_*), P_2(x_*, y_*, z_*)]J(x_*, y_*, z_*)\|_2 \\
 &\geq \|[P_1(x_*, y_*, z_*), P_2(x_*, y_*, z_*)]\|_2 \cdot \sigma_{\min}(J(x_*, y_*, z_*)),
 \end{aligned}$$

where the inequality follows from (22). This implies that

$$\begin{aligned}
& |P_3(x_*, y_*, z_*)| \\
(24) \quad & \geq \frac{\|[P_1(x_*, y_*, z_*), P_2(x_*, y_*, z_*), P_3(x_*, y_*, z_*)]\|_2 \cdot \sigma_{\min}(J(x_*, y_*, z_*))}{\sqrt{1 + \sigma_{\min}(J(x_*, y_*, z_*))^2}} \\
& \geq \frac{\|[P_1(x_*, y_*, z_*), P_2(x_*, y_*, z_*), P_3(x_*, y_*, z_*)]\|_2 \cdot \sigma_{\min}(J(x_*, y_*, z_*))}{c_1(d_1, d_2)}.
\end{aligned}$$

The second inequality is derived from the first by noting that $\sigma_{\min}(J(x_*, y_*, z_*)) \leq \|J(x_*, y_*, z_*)\|_2 \leq c_2(d_1, d_2)$ by our assumption that the polynomial coefficients are normalized.

We want to derive a lower bound on

$$\|[P_1(x_*, y_*, z_*), P_2(x_*, y_*, z_*), P_3(x_*, y_*, z_*)]\|_2.$$

As above, let \mathbf{y} be the unit-length left eigenvector of $sA - tB$ corresponding to the eigenvalue (s, t) under consideration. Let $\hat{\mathbf{y}}$ be the extension of \mathbf{y} by inserting zeros in the positions of the rows deleted in the algorithm, so that $\hat{\mathbf{y}}^* M_d(F_1, F_2, F_3) = \mathbf{0}$. Then the entries $\hat{\mathbf{y}}^*$ are equal to the coefficients of P_1, P_2, P_3 (none omitted).

The vector $\hat{\mathbf{y}}^*$ is not the only left nullvector of $M_d = M_d(F_1, F_2, F_3)$. Recall from Section 4 that $H_d M_d = 0$, where $H_d = H_d(F_1, F_2, F_3)$. We first wish to argue that $\hat{\mathbf{y}}^*$ is not close to the rowspan of H_d , in other words, that \mathbf{y}^* is not one of the “generic” left null vectors of M_d . Select an arbitrary vector in the rowspan of H_d , say $\mathbf{v}^* H_d$. Partition H_d as $[H, H']$, where H are the dropped (basis) columns of H_d selected by our algorithm. Note that $\hat{\mathbf{y}}^*$ is all zeros in the positions indexed by H by construction of $\hat{\mathbf{y}}^*$. Thus,

$$(25) \quad \|\hat{\mathbf{y}}^* - \mathbf{v}^* H_d\|_2^2 = \|\mathbf{y}^* - \mathbf{v}^* H'\|_2^2 + \|\mathbf{v}^* H\|_2^2.$$

Take two cases. In the first case, $\|\mathbf{v}^* H'\|_2 \leq 0.5$, in which case the first term of (25) is at least $(0.5)^2$ (since \mathbf{y} is a unit vector). In the second case, $\|\mathbf{v}^* H'\|_2 \geq 0.5$, so $\|\mathbf{v}\|_2 \geq 0.5/\|H'\|_2 \geq c_3(d_1, d_2)$ because of the normalization of the coefficients. Thus, the second term of (25) is at least $(c_3(d_1, d_2)\sigma_{\min}(H))^2$. By (6), this quantity is at least $(c_4(d_1, d_2)\sigma_{\min}(M_{d-1}(F_1, F_2)))^2$. Thus, in either case,

$$(26) \quad \text{dist}(\hat{\mathbf{y}}^*, \text{Rowspan}(H_d(F_1, F_2, F_3))) \geq c_4(d_1, d_2)\sigma_{\min}(M_{d-1}(F_1, F_2)).$$

We showed in Section 7 that

$$M_d(F_1^0, F_2^0, F_3^0) = W M_d(F_1, F_2, F_3) V,$$

where W and V are monomial change-of-basis matrices whose norms (and the norms of their inverses) are bounded by $(d+1)2^d$. We also have

$$H_d(F_1^0, F_2^0, F_3^0) = U H_d(F_1, F_2, F_3) W^{-1},$$

where U is the correct change-of-basis matrix for the monomials associated with the rows of H_d . Let $\hat{\mathbf{w}}^* = \mathbf{y}^* W^{-1}$, so that $\hat{\mathbf{w}}^* M_d' = \mathbf{0}$, where $M_d' = M_d(F_1^0, F_2^0, F_3^0)$. By applying the square matrix W^{-1} to the quantities in (26), we conclude that

$$(27) \quad \text{dist}(\hat{\mathbf{w}}^*, \text{Rowspan}(U H_d(F_1, F_2, F_3) W^{-1})) \geq c_5(d_1, d_2)\sigma_{\min}(M_{d-1}(F_1, F_2)).$$

Here, $c_5(d_1, d_2)$ is another function of d_1, d_2 that accounts for the product of $c_4(d_1, d_2)$ appearing in (27) and the norm of W^{-1} , the change-of-basis matrix. (Note that, since U is nonsingular, it does not affect the rowspan.)

Next, observe that the columns of $H_d(F_1^0, F_2^0, F_3^0)$ associated with the monomials z^{d-d_i} , for $i = 1, 2, 3$, are all zeros, because the polynomials F_1^0, F_2^0, F_3^0 have $(0, 0, 1)$ as a root by construction. Let I be the indices of these three distinguished positions, and let \bar{I} be the complementary indices. We can then rewrite (27) as

$$(28) \quad \begin{aligned} \|\hat{\mathbf{w}}(I)\|_2^2 + \text{dist}(\hat{\mathbf{w}}(\bar{I})^*, \text{Rowspan}(H_d(F_1^0, F_2^0, F_3^0)(:, \bar{I})))^2 \\ \geq c_5(d_1, d_2)^2 \sigma_{\min}(M_{d-1}(F_1, F_2))^2. \end{aligned}$$

Here, the notation $X(:, \bar{I})$, borrowed from Matlab, refers to the submatrix with columns indexed by \bar{I} . It is not hard to see that $H_d(F_1^0, F_2^0, F_3^0)(:, \bar{I})$ appearing in (28) is identical to H_d^0 introduced above. Also, one checks that $\hat{\mathbf{w}}(I)^*$ is exactly equal to the vector $[P_1(x_*, y_*, z_*), P_2(x_*, y_*, z_*), P_3(x_*, y_*, z_*)]$ under consideration in this section because $\hat{\mathbf{y}}$ contains the coefficients of P_1, P_2, P_3 , and W maps certain vectors of monomials in x_*, y_*, z_* into columns of the identity matrix.

Now, we take two cases. The first case is that the second term on the left-hand side of (28) is less than half the right-hand side. In this case, we are finished proving the proposition, because we conclude that

$$\|\hat{\mathbf{w}}(I)\| \geq c_5(d_1, d_2) \sigma_{\min}(M_{d-1}(F_1, F_2)) / \sqrt{2},$$

which we can combine with inequality (24) to prove the proposition. In fact, this proves a stronger version of the proposition since there is no dependence on $\sigma_{\min}(N_d)$ in this case.

The other case is that the second term on the left-hand side of (28) is at least half the right-hand side, i.e.,

$$(29) \quad \text{dist}(\hat{\mathbf{w}}(\bar{I})^*, \text{Rowspan}(H_d^0)) \geq c_5(d_1, d_2) \sigma_{\min}(M_{d-1}(F_1, F_2)) / \sqrt{2}.$$

Applying (23) to (29) (since $H_d^0 N_d = 0$) yields

$$(30) \quad \|\hat{\mathbf{w}}(\bar{I})^* N_d\| \geq c_5(d_1, d_2) \sigma_{\min}(N_d) \sigma_{\min}(M_{d-1}(F_1, F_2)) / \sqrt{2}.$$

Recall that N_d is equal to $M'_d(\bar{I}, :)$ after deleting three zero columns. Thus,

$$\|\hat{\mathbf{w}}(\bar{I})^* N_d\| = \|\hat{\mathbf{w}}(\bar{I})^* M'_d(\bar{I}, :)\|.$$

From the equations $\mathbf{0} = \hat{\mathbf{w}}^* M'_d = \hat{\mathbf{w}}(I) M'_d(I, :) + \hat{\mathbf{w}}(\bar{I}) M'_d(\bar{I}, :)$, we conclude that

$$\|\hat{\mathbf{w}}(I)^* M'_d(I, :)\| = \|\hat{\mathbf{w}}(\bar{I})^* M'_d(\bar{I}, :)\|.$$

Chain the two previous equations to substitute for the left-hand side of (30) to obtain

$$\|\hat{\mathbf{w}}(I)^* M'_d(I, :)\| \geq c_5(d_1, d_2) \sigma_{\min}(N_d) \sigma_{\min}(M_{d-1}(F_1, F_2)) / \sqrt{2},$$

so

$$\begin{aligned} \|\hat{\mathbf{w}}(I)\| &\geq \frac{c_5(d_1, d_2) \sigma_{\min}(N_d) \sigma_{\min}(M_{d-1}(F_1, F_2))}{\sqrt{2} \cdot \|M'_d(I, :)\|} \\ &\geq c_6(d_1, d_2) \sigma_{\min}(N_d) \sigma_{\min}(M_{d-1}(F_1, F_2)), \end{aligned}$$

since $\|M'_d\|$ is bounded above in terms of d_1, d_2 because of problem normalization. Thus, as in the first case, we have obtained a lower bound on $\|\hat{\mathbf{w}}(I)\|$. Again, as in the first case, we can combine this lower bound with (24) to establish the proposition. \square

The analysis in Section 6 was based on R_0 , which divides any major subdeterminant of the matrix M_{d-1}^0 . We can find M_{d-1}^0 as a submatrix in N_d : If we reordered the rows and the columns, we could write

$$N_d = \left[\begin{array}{c|c} M_{d-1}^0 & \mathbf{0} \\ \hline * & \end{array} \right].$$

The next theorem shows that R_0 is also connected to the matrix N_d .

Theorem 10. *Suppose we drop any $\binom{d_1}{2} + \binom{d_2}{2}$ rows from the matrix N_d . Then the determinant D of the resulting square matrix is a multiple of R_0 ; i.e., $D = S_0 R_0$. The extraneous factor S_0 is, up to a sign, the subdeterminant of H_d^0 where we take the columns corresponding to the rows we dropped from N_d .*

Proof. We know that $M_d(F_1^0, F_2^0, F_3^0)$ has $\binom{d_1}{2} + \binom{d_2}{2}$ more rows than columns, and to get N_d we delete three columns and three rows, so the same is true for N_d . The rows of N_d are related by $H_d^0 N_d = 0$ and the matrix H_d^0 has $\binom{d_1}{2} + \binom{d_2}{2}$ rows that are linearly independent for symbolic coefficients. So by Theorem 2 there exists R'_0 such that $R'_0 = \pm D/S_0$ for any choice of determinants D and S_0 as in the statement of this theorem with $S_0 \neq 0$.

In the same way as in the proof of Theorem 5 we can show that R'_0 is an integer polynomial in the coefficients of F_1^0, F_2^0, F_3^0 and that it is homogeneous of the same degree as R_0 in the coefficients of F_i^0 .

Let N' be the submatrix of N_d we get by deleting the F_1^0 - and F_2^0 -rows whose monomials are divisible by y . Then $\det(N') = \pm u^k R'_0$, where $k = \binom{d_1}{2} + \binom{d_2}{2}$. But

$$N' = \left[\begin{array}{c|c} M' & \mathbf{0} \\ \hline * & \end{array} \right],$$

where M' is the submatrix of M_{d-1}^0 we get by again deleting the F_1^0 - and F_2^0 -rows whose monomials depend on y . So $\det(M')$ divides $\det(N')$ and therefore R_0 must divide $\det(N')$. If we had deleted rows divisible by x instead of y we would similarly have seen that R_0 divides $v^k R'_0$. So R_0 must divide R'_0 and since they have the same degree, $R'_0 = R_0$ (if we choose the sign correctly). \square

Now we can use this theorem to relate $\sigma_{\min}(N_d)$ to the quantity R_0 :

Proposition 11.

$$\sigma_{\min}(N_d) \geq \frac{|R_0|}{c(d_1, d_2)},$$

where $c(d_1, d_2)$ is some function of d_1 and d_2 .

Proof. We know $|u|^2 + |v|^2 = 1$, so either $|u| \geq 1/\sqrt{2}$ or $|v| \geq 1/\sqrt{2}$. In the former case, let N' be the submatrix of N_d we get by deleting the F_1^0 - and F_2^0 -rows whose monomials are divisible by y , so that $\det(N') = \pm u^k R_0$. In the latter case, delete rows divisible x instead of y , so that $\det(N') = \pm v^k R_0$. In either case we get a matrix N' such that

$$|\det(N')| \geq 2^{-k/2} |R_0|.$$

Adding rows to a matrix makes the singular values bigger:

$$(31) \quad X = \begin{bmatrix} Y \\ Z \end{bmatrix} \Rightarrow \sigma_j(X) \geq \sigma_j(Y),$$

where $\sigma_j(\cdot)$ denotes the j th biggest singular value of a matrix. (This follows from Theorem 8.1.5 of [11].) So,

$$\sigma_{\min}(N_d) = \sigma_{m-3}(N_d) \geq \sigma_{m-3}(N'),$$

where $m = \binom{d+2}{2}$.

The absolute value of the determinant of a square matrix is the product of its singular values, so

$$|\det(N')| = \sigma_1(N') \cdots \sigma_{m-3}(N') \leq (\sigma_1(N'))^{m-4} \sigma_{m-3}(N').$$

The biggest singular value is the 2-norm of the matrix,

$$\sigma_1(N') = \|N'\|_2 \leq \|N'\|_F \leq c_1(d_1, d_2),$$

so

$$\sigma_{\min}(N_d) \geq \sigma_{m-3}(N') \geq \frac{|R_0|}{2^{k/2} c_1(d_1, d_2)^{m-4}} = \frac{|R_0|}{c_2(d_1, d_2)},$$

where $c_i(d_1, d_2)$ is a function of d_1 and d_2 . □

8.4. The second smallest singular value. As before, let (s, t) be a simple eigenvalue (with $|s|^2 + |t|^2 = 1$) for the eigenvalue problem (11) and \mathbf{x} be the corresponding eigenvector. Let δ be the second smallest singular value for the matrix $M = sA - tB$, (i.e., the smallest nonzero singular value). In this section we argue that there is a lower bound for δ in terms of the choice of the swepline and the condition numbers of the top part of A and the Jacobian of F_1, F_2 at the root.

Proposition 12. *Let δ be as defined above. Then*

$$\delta \geq \frac{|R_0| \cdot \sigma_{\min}(M_{d-1}(F_1, F_2)) \cdot [\sigma_{\min}(J(x_*, y_*, z_*))]^2}{c(d_1, d_2)},$$

where $c(d_1, d_2)$ is an expression in d_1 and d_2 .

Proof. First, we estimate how much the second smallest singular value increases when we add back to the matrix M the rows we dropped in step 3 of the algorithm. Let \mathbf{v} be the right singular vector corresponding to δ , so that $\|M\mathbf{v}\|_2 = \delta$. Let $\tilde{M} = sA_d - tB_d$ be the rectangular matrix with no rows being dropped and let $\tilde{\delta}$ be its second smallest singular value. Note that $\tilde{\delta} \geq \delta$ by (31), but we need an inequality in the opposite direction.

By Corollary 3 we know the rows of \tilde{M} are related by $\tilde{H}\tilde{M} = 0$, where

$$\tilde{H} = \begin{bmatrix} -M_{d-d_1}(F_3) & 0 & M_{d-1}(F_1) \\ 0 & -M_{d-d_2}(F_3) & M_{d-1}(F_2) \end{bmatrix}.$$

Let P be a permutation matrix such that

$$\tilde{H}P^T = [H', H], \quad P\tilde{M} = \begin{bmatrix} M \\ M' \end{bmatrix}.$$

Assume $M_{d-1}(F_1, F_2)$ is nonsingular. We chose rows to drop so that $S = \det(H) \neq 0$, so H is nonsingular, and we have

$$\tilde{H}\tilde{M} = H'M + HM' = 0 \quad \Rightarrow \quad M' = -H^{-1}H'M.$$

Furthermore, (6) gives us a lower bound on the smallest singular value of H ,

$$(32) \quad \sigma_h(H) \geq \frac{\sigma_h(M_{d-1}(F_1, F_2))}{\sqrt{d_1 d_2} 2^h},$$

where $h = \binom{d_1}{2} + \binom{d_2}{2}$.

The eigenvector \mathbf{x} is also the right singular vector for the smallest singular value (which is zero) for both M and \tilde{M} . This means that \mathbf{v} is orthogonal to \mathbf{x} and therefore $\|\tilde{M}\mathbf{v}\|_2 \geq \tilde{\delta}$. So we have

$$\tilde{\delta}^2 \leq \|\tilde{M}\mathbf{v}\|_2^2 = \|M\mathbf{v}\|_2^2 + \|M'\mathbf{v}\|_2^2 \leq (1 + \|H^{-1}\|_2^2 \cdot \|H'\|_2^2) \cdot \|M\mathbf{v}\|_2^2.$$

We know that $\|H^{-1}\|_2 = 1/\sigma_h(H)$ and $\|M\mathbf{v}\|_2 = \delta$, so

$$\tilde{\delta} \leq \frac{\sqrt{\sigma_h(H)^2 + \|H'\|_2^2}}{\sigma_h(H)} \delta.$$

Because of the normalization of the coefficients, the square root can be bounded in terms of h . So by the above inequality and from (32), we have

$$\tilde{\delta} \leq \frac{c_1(d_1, d_2)}{\sigma_h(M_{d-1}(F_1, F_2))} \delta,$$

where $c_1(d_1, d_2)$ is a function of d_1 and d_2 .

Now rotate the coordinates so that the root will be $(0, 0, 1)$. As in Section 6 let F_1^0, F_2^0, F_3^0 be the polynomials F_1, F_2, F_3 in the new coordinates, and let M_d^0 be the matrix $M_d(F_1^0, F_2^0, F_3^0)$ with the last column being deleted. It follows from Section 7 that

$$\sigma_{\min}(M_d^0) \leq (d + 1)^2 2^{2d} \tilde{\delta};$$

hence,

$$\sigma_{\min}(M_d^0) \leq \frac{c_2(d_1, d_2)}{\sigma_h(M_{d-1}(F_1, F_2))} \delta,$$

where $c_2(d_1, d_2)$ is an expression in d_1 and d_2 .

If we reorder the rows and the columns, we can write

$$M_d^0 = \left[\begin{array}{c|c} N_d & 0 \\ \hline * & W \end{array} \right],$$

where

$$W = \left[\begin{array}{cc} J^0 \\ v & -u \end{array} \right] = \left[\begin{array}{cc} \frac{\partial F_1^0}{\partial x} & \frac{\partial F_1^0}{\partial y} \\ \frac{\partial F_2^0}{\partial x} & \frac{\partial F_2^0}{\partial y} \\ v & -u \end{array} \right],$$

with the derivatives evaluated at the root $(0, 0, 1)$.

Let N' be the submatrix of N_d constructed in the proof of Proposition 11, so that $|\det(N')| \geq 2^{-k/2}|R_0|$. Then we have an $(m - 1) \times (m - 1)$ submatrix of M_d^0 ,

$$N'' = \left[\begin{array}{c|c} N' & 0 \\ \hline * & J^0 \end{array} \right],$$

where $m = \binom{d+2}{2}$. By (31),

$$\sigma_{\min}(M_d^0) = \sigma_{m-1}(M_d^0) \geq \sigma_{m-1}(N'').$$

Also

$$|\det(N') \det(J^0)| = |\det(N'')| \leq \|N''\|_2^{m-2} \sigma_{m-1}(N''),$$

and we have a bound on $\|N''\|_2$ in terms of d_1 and d_2 , so

$$\sigma_{\min}(M_d^0) \geq \sigma_{m-1}(N'') \geq \frac{|R_0 \det(J^0)|}{c_3(d_1, d_2)} \geq \frac{|R_0| \cdot (\sigma_{\min}(J^0))^2}{c_3(d_1, d_2)},$$

where $c_3(d_1, d_2)$ is some function of d_1 and d_2 . Finally, note that the rotation of coordinates does not affect the singular values of the Jacobian. \square

8.5. **Main result.** Summing up, we have shown that if the matrix pencil $sA - tB$ is perturbed by $sE - tF$ and the perturbation is sufficiently small, then the eigenvector corresponding to the root (x_*, y_*, z_*) is perturbed by $\Delta \mathbf{x}$, where

$$\|\Delta \mathbf{x}\|_2 \leq \frac{4}{\delta} \left(1 + \frac{\sqrt{\|A\|_F^2 + \|B\|_F^2}}{\sqrt{|\mathbf{y}^* A \mathbf{x}|^2 + |\mathbf{y}^* B \mathbf{x}|^2}} \right) \sqrt{\|E\|_F^2 + \|F\|_F^2}.$$

We have a lower bound for δ ,

$$\delta \geq \frac{|R_0| \cdot \sigma_{\min}(M_{d-1}(F_1, F_2)) \cdot [\sigma_{\min}(J(x_*, y_*, z_*))]^2}{c(d_1, d_2)}.$$

The quantity

$$\frac{\sqrt{\|A\|_F^2 + \|B\|_F^2}}{\sqrt{|\mathbf{y}^* A \mathbf{x}|^2 + |\mathbf{y}^* B \mathbf{x}|^2}} = \frac{\sqrt{m + d_1 d_2}}{|P_3(x_*, y_*, z_*)| \sqrt{|L_\alpha(x_*, y_*, z_*)|^2 + |L_\beta(x_*, y_*, z_*)|^2}}$$

is the condition number for the eigenvalue, and we have shown that

$$|P_3(x_*, y_*, z_*)| \geq \frac{|R_0| \cdot \sigma_{\min}(M_{d-1}(F_1, F_2)) \cdot \sigma_{\min}(J(x_*, y_*, z_*))}{c(d_1, d_2)}.$$

In both occurrences $c(d_1, d_2)$ denote some function of d_1 and d_2 , but a different one for each case.

It remains to get a lower bound for $|R_0|$. We know R_0 is a homogeneous polynomial in u and v of degree $k = d_1 d_2 - 1$, so we can write

$$R_0(u, v) = \sum_{j=0}^k \hat{R}_j u^{k-j} v^j.$$

The idea is to show that for most (u, v) the value of R_0 is not too small, so that there is a low probability of hitting a very small value. Recall that the distribution of (u, v) is uniform, so the expected value of any function of (u, v) is the integral of this function divided by the surface area $2\pi^2$.

Theorem 13. *The expected value of $|R_0|^2$ is*

$$\frac{1}{2\pi^2} \int_{\mathcal{S}} |R_0|^2 = \frac{1}{k+1} \sum_{j=0}^k |\hat{R}_j|^2 \binom{k}{j}^{-1},$$

where $\mathcal{S} = \{(u, v) \in \mathbb{C}^2 : |u|^2 + |v|^2 = 1\}$, and, as in the above discussion, \hat{R}_j 's denote the coefficients of $R_0(u, v)$.

Proof. We can use Parseval's equality for the semidiscrete Fourier transform

$$\int_0^{2\pi} \left| \sum_j v_j e^{-ij\theta} \right|^2 d\theta = 2\pi \sum_j |v_j|^2$$

to compute the average of $|R_0|^2$ over \mathcal{S} . For the integration we parameterize \mathcal{S} by

$$(u, v) = (e^{i\theta} \cos \psi, e^{i\phi} \sin \psi), \quad \theta, \phi \in [0, 2\pi], \quad \psi \in [0, \pi/2].$$

By identifying \mathbb{C}^2 with \mathbb{R}^4 , it is easily verified that the volume element in these coordinates is $\cos \psi \sin \psi d\theta d\phi d\psi$. Thus, the average is

$$\begin{aligned} & \frac{1}{2\pi^2} \int_0^{\pi/2} \int_0^{2\pi} \int_0^{2\pi} |R_0(e^{i\theta} \cos \psi, e^{i\phi} \sin \psi)|^2 \cos \psi \sin \psi d\theta d\phi d\psi \\ &= \frac{1}{4\pi^2} \int_0^{\pi/2} \sin 2\psi \int_0^{2\pi} \int_0^{2\pi} \left| \sum_{j=0}^k \hat{R}_j \cos^{k-j} \psi \sin^j \psi e^{i(k-j)\theta} e^{ij\phi} \right|^2 d\phi d\theta d\psi \\ &= \frac{1}{2\pi} \int_0^{\pi/2} \sin 2\psi \int_0^{2\pi} \sum_{j=0}^k \left| \hat{R}_j \cos^{k-j} \psi \sin^j \psi e^{i(k-j)\theta} \right|^2 d\theta d\psi \\ &= \int_0^{\pi/2} \sin 2\psi \sum_{j=0}^k |\hat{R}_j|^2 \cos^{2(k-j)} \psi \sin^{2j} \psi d\psi \\ &= 2 \sum_{j=0}^k |\hat{R}_j|^2 \int_0^{\pi/2} \cos^{2k-2j+1} \psi \sin^{2j+1} \psi d\psi = \sum_{j=0}^k |\hat{R}_j|^2 \frac{1}{(k+1) \binom{k}{j}}, \end{aligned}$$

where the last equality follows from the properties of the Beta-function (see [12]). □

The intuition behind the next theorem is that on the unit sphere, a low-order polynomial can have values much smaller than its mean on only a very small subset.

Theorem 14. *The probability of picking a sweepline so that $|R_0(u, v)| < \varepsilon \max_j |\hat{R}_j|$ is less than*

$$2k \sin^{-1} \left(\sqrt{2}(\varepsilon \sqrt{k+1})^{1/k} \right),$$

where $k = \deg_{u,v}(R_0)$.

Proof. We can write

$$R_0(u, v) = K \prod_{j=1}^k (\eta_j u - \zeta_j v),$$

where (ζ_j, η_j) are the roots of R_0 , with $|\zeta_j|^2 + |\eta_j|^2 = 1$, and K is a constant. For $(u, v) \in \mathcal{S}$,

$$|R_0(u, v)| \leq |K| \prod_{j=1}^k |\eta_j u - \zeta_j v| \leq |K|,$$

because $|\eta_j u - \zeta_j v| \leq 1$. We can certainly pick (u, v) so that $|R_0(u, v)|^2$ is at least as big as the mean derived in Theorem 13, so

$$|K| \geq \frac{\max_j |\hat{R}_j|}{\sqrt{(k+1)2^k}}.$$

Suppose we are given $\varepsilon > 0$ and $(u, v) \in \mathcal{S}$ such that $|R_0(u, v)| < \varepsilon \max_j |\hat{R}_j|$. Let

$$\gamma = \min_j |\eta_j u - \zeta_j v|.$$

Then $|R_0(u, v)| \geq |K| \gamma^k$, so

$$\gamma \leq \left(\frac{|R_0(u, v)|}{|K|} \right)^{1/k} < \sqrt{2} \left(\varepsilon \sqrt{k+1} \right)^{1/k}.$$

So if we define $\delta = \sqrt{2} (\varepsilon \sqrt{k+1})^{1/k}$, then

$$|R_0(u, v)| < \varepsilon \max_j |\hat{R}_j| \Rightarrow |\eta_j u - \zeta_j v| < \delta, \text{ for at least one } j.$$

This shows that if $|R_0(u, v)|$ is small compared to the biggest coefficient, then (u, v) is close to a root of R_0 in the metric we defined in (17).

If we were dealing with real values, then $|\eta_j u - \zeta_j v|$ would be the sine of the angle between (u, v) and (ζ_j, η_j) . But we have complex coordinates. Let (θ, ϕ, ψ) and $(\theta_j, \phi_j, \psi_j)$ be (u, v) and (ζ_j, η_j) written in the coordinates used in the integration above. Then, since $0 \leq \psi, \psi_j \leq \pi/2$,

$$\begin{aligned} |\eta_j u - \zeta_j v| &= |e^{i\theta} \cos \psi e^{i\phi_j} \sin \psi_j - e^{i\phi} \sin \psi e^{i\theta_j} \cos \psi_j| \\ &= |\cos \psi \sin \psi_j - e^{i(\theta - \theta_j + \phi - \phi_j)} \sin \psi \cos \psi_j| \\ &\geq |\cos \psi \sin \psi_j - \sin \psi \cos \psi_j| = \sin |\psi - \psi_j|. \end{aligned}$$

This shows that the probability of being within δ from the root (ζ_j, η_j) in our metric is less than or equal to

$$\frac{(2\pi)^2}{2\pi^2} \int_{\psi_j - \sin^{-1} \delta}^{\psi_j + \sin^{-1} \delta} |\cos \psi \sin \psi| d\psi = \int_{\psi_j - \sin^{-1} \delta}^{\psi_j + \sin^{-1} \delta} |\sin 2\psi| d\psi \leq 2 \sin^{-1} \delta,$$

and therefore the probability of being within δ from a root is $\leq 2k \sin^{-1} \delta$. \square

Theorem 15. *Let (x_*, y_*, z_*) be a root of the polynomials F_1 and F_2 , and let \mathbf{x} be the eigenvector of*

$$sA\mathbf{x} = tB\mathbf{x}$$

corresponding to the root; i.e.,

$$\mathbf{x} = [x_*^d, x_*^{d-1} y_*, x_*^{d-1} z_*, \dots, z_*^d]^T.$$

There exists a function $\mathcal{F} : [0, 1] \times \mathbb{R}_+^2 \times \mathbb{Z}^2 \rightarrow \mathbb{R}_+$, such that with probability τ the sweepline is chosen so that the condition number of the eigenvector is at most

$$\mathcal{F}(\tau, \kappa(A_1), \sigma_{\min}(J(x_*, y_*, z_*)), d_1, d_2),$$

where $\kappa(A_1)$ is the condition number of the top part of A and $\sigma_{\min}(J(x_, y_*, z_*))$ is the smallest singular value of the Jacobian at the root.*

Remark 1. This is the main result of the paper. As mentioned before, the probability space in the theorem is over choices of randomized sweepline and not over problem data. Note that the eigenvector condition number is directly proportional to the accuracy of the computed root (x_*, y_*, z_*) in the presence of roundoff error, as detailed at the beginning of Section 8.

Remark 2. The quantity $1/\kappa(A_1)$, that is, the reciprocal condition number of the top part of A ($= M_d(F_1, F_2)$), is the relative distance to singularity for A_1 , i.e., the minimum value of $\|E\|/\|A_1\|$ over matrices E such that $A_1 + E$ is rank deficient. One might wonder also about $1/\tilde{\kappa}(A_1)$, which is the same minimum except that E is restricted to matrices that preserve the structure of A_1 (i.e., E has the form $M_d(e_1, e_2)$ for two polynomials e_1, e_2). This corresponds to the problem of the smallest perturbation to the original polynomial system to make it singular. The general problem of condition number with respect to structured perturbations has received some attention in the literature [25] and is much more difficult to compute than the unstructured distance. Clearly $\tilde{\kappa}(A_1) \leq \kappa(A_1)$. It would be interesting to derive a bound in the opposite direction.

Proof. Recall that R_0 is a polynomial in u, v and in the coefficients of F_1^0, F_2^0 . Regarding R_0 as a polynomial in u, v alone (as we have for this section of the paper), its coefficients \hat{R}_j are polynomials in the coefficients of the polynomials F_1^0 and F_2^0 ; i.e.,

$$\hat{R}_j \in \mathbb{C}[a'_1, \dots, a'_\mu, b'_1, \dots, b'_\nu],$$

where $\mu = \binom{d_1+2}{2} - 1$ and $\nu = \binom{d_2+2}{2} - 1$. The Jacobian at the root $(0, 0, 1)$ is

$$J^0 = \begin{bmatrix} a'_{j_1} & a'_{j_2} \\ b'_{j_3} & b'_{j_4} \end{bmatrix},$$

for some indices j_i . We know from Theorem 7 that if $\hat{R}_j = 0$ for all j , then $J^0 = 0$ or the polynomials F_1^0 and F_2^0 have a factor in common. The latter is equivalent to $M_d(F_1^0, F_2^0)$ not having full row rank. Let H be an $h \times h$ submatrix of $M_d(F_1^0, F_2^0)$, where $h = \binom{d_1+1}{2} + \binom{d_2+1}{2}$ is the number of rows of $M_d(F_1^0, F_2^0)$. Then the condition that F_1^0 and F_2^0 have a common factor is equivalent to stating that all such matrices H have a zero determinant.

Then we know by Theorem 7 that if $\hat{R}_j = 0$ for all j , then for example $a'_{j_1} \det(H) = 0$. Note that $a'_{j_1} \det(H)$, like \hat{R}_j , is a polynomial in $\mathbb{C}[a'_1, \dots, a'_\mu, b'_1, \dots, b'_\nu]$. By Hilbert's Nullstellensatz (strong form) there exists an integer $r \geq 1$ such that $(a'_{j_1} \det(H))^r$ is in the ideal generated by the coefficients of R_0 , i.e.,

$$(a'_{j_1} \det(H))^r = \sum_{j=0}^k \hat{S}_j \hat{R}_j,$$

for some $\hat{S}_j \in \mathbb{C}[a'_1, \dots, a'_\mu, b'_1, \dots, b'_\nu]$.

Note that the polynomials \hat{S}_j are universal; their coefficients depend only on d_1, d_2 and not on any of the other problem data. For the case $d_1 = d_2 = 2$, we have explicitly worked out by hand many of the polynomials \hat{S}_j . All the polynomials we worked out have small-integer coefficients. We were not able to extend this hand analysis to the case $d_1 = 2, d_2 = 3$. It should be possible in principle for a particular choice of d_1, d_2 to use an effective version of Hilbert's Nullstellensatz (e.g., a Gröbner basis algorithm) to find all the \hat{S}_j 's. But we were unable to complete this computation even when $d_1 = 2, d_2 = 3$ because memory was exhausted by the Gröbner basis algorithm. Thus, our eigenvector condition bound depends on (d_1, d_2) in a manner that we cannot state in closed form.

Given the degrees d_1 and d_2 , the algebraic makeup of the problem is completely determined, that is, we can write out what the polynomials and matrices are as symbolic objects. This means that r is also determined; it is some function of d_1 and d_2 . Although we do not know r in closed form, it is possible to obtain an upper bound on it using estimates for the Nullstellensatz. For example, it is known [17] that the polynomial multipliers in the weak form of the Nullstellensatz have degree at most $4nd^n$, where n is the number of variables and d is the maximum degree among the polynomials. The Rabinowitsch proof of the strong form starting from the weak form implies that the exponent r is therefore bounded by $4(n+1)(d+1)^{(n+1)}$, where d is the maximum degree among the R_j 's and factors like $a'_{j_1} \det(H)$ and n are the number of variables. The degree bound d is at most $h+1$ (where h is the number of rows in the top part). The number of variables n is the total number of coefficients, which is $(d_1+1)(d_1+2)/2 + (d_2+1)(d_2+2)/2$.

Therefore, the quantity r is at most $2^{O((d_1^2+d_2^2)\log(d_1+d_2))}$. According to Sombra [26], sharper bounds are possible for the several quantities described above, but these improvements do not decrease the estimate that $r \leq 2^{O((d_1^2+d_2^2)\log(d_1+d_2))}$.

Furthermore, the values of the polynomials \hat{S}_j when the specific problem data (that is, $a'_1, \dots, a'_\mu, b'_1, \dots, b'_\nu$) is substituted is bounded above by some function $c(d_1, d_2)$ depending on d_1, d_2 only. This is because the coefficients of the \hat{S}_j 's depend only on d_1, d_2 , and the problem data (that is, $a'_1, \dots, a'_\mu, b'_1, \dots, b'_\nu$) is bounded above since the coefficients of F_1, F_2 are assumed to be normalized. So there exists some function $c(d_1, d_2)$ such that

$$|a'_{j_1} \det(H)|^r \leq c(d_1, d_2) \max_j |\hat{R}_j|.$$

Again, the result of [17] could be used to get a very large upper bound on the factor $c(d_1, d_2)$ appearing here, but we omit this analysis. We can assume without loss of generality that $|a'_{j_1}| \geq |a'_{j_2}|$. Then $|a'_{j_1}| \geq \sigma'_{\min}(J^0)/\sqrt{2}$. If we choose the submatrix H by doing a QR-factorization with column pivoting, then (6) gives

$$\sigma_h(H) \geq \frac{\sigma_h(M_d(F_1^0, F_2^0))}{2^h \sqrt{d_1 d_2 - 1}}.$$

So we have

$$\begin{aligned} \max_j |\hat{R}_j| &\geq \frac{|a'_{j_1} \det(H)|^r}{c(d_1, d_2)} \geq \frac{(\sigma_{\min}(J^0) \cdot [\sigma_{\min}(H)]^h)^r}{2^{r/2} c(d_1, d_2)} \\ &\geq \frac{(\sigma_{\min}(J^0) \cdot [\sigma_{\min}(M_d(F_1^0, F_2^0))]^h)^r}{c'(d_1, d_2)} \\ &\geq \frac{(\sigma_{\min}(J(x_*, y_*, z_*)) \cdot [\sigma_{\min}(M_d(F_1, F_2))]^h)^r}{c''(d_1, d_2)}. \end{aligned}$$

Then by Theorem 14, with probability at least $\tau/2$,

$$|R_0| \geq \frac{\varepsilon (\sigma_{\min}(J(x_*, y_*, z_*)) \cdot [\sigma_{\min}(M_d(F_1, F_2))]^h)^r}{c''(d_1, d_2)},$$

where

$$\varepsilon = \frac{1}{\sqrt{2^k(k+1)}} \sin^k \left(\frac{1 - \tau/2}{2k} \right).$$

It follows from (20) that we can find a function $\gamma(\tau)$, so that with probability $\tau/2$,

$$\sqrt{|L_\alpha(x_*, y_*, z_*)|^2 + |L_\beta(x_*, y_*, z_*)|^2} \geq \gamma(\tau).$$

So for a given τ we have shown that we can obtain bounds for all the quantities in the eigenvector condition number formula that hold with probability τ .

The bounds that we have derived are all in terms of the smallest singular value of $A_1 = M_d(F_1, F_2)$. But we can convert them into bounds in terms of the condition number $\kappa(A_1)$, because

$$\kappa(A_1) = \|A_1\|_2 \|A^{-1}\|_2 \geq \frac{\|A_1\|_F}{\sqrt{h} \sigma_{\min}(A_1)} = \frac{1}{\sigma_{\min}(A_1)}. \quad \square$$

8.6. Conditioning of the reduced eigenvalue problem. The analysis has so far focused on conditioning of an eigenvector of the generalized eigenvalue problem $sA\mathbf{x} = tB\mathbf{x}$, which we can write as

$$s \begin{bmatrix} A_1 \\ A_2 \end{bmatrix} \mathbf{x} = t \begin{bmatrix} 0 \\ B_2 \end{bmatrix} \mathbf{x},$$

separating the top and bottom parts of the matrices as we did in Section 5. But in the algorithm we actually use the QZ -algorithm to compute the eigenvectors for the reduced problem

$$sA_2Z\mathbf{y} = tB_2Z\mathbf{y},$$

where the columns of Z are an orthonormal basis for the nullspace of A_1 . We then compute \mathbf{x} by multiplying \mathbf{y} by Z .

Suppose we perturb the reduced problem

$$(s + \Delta s)(A_2Z + \tilde{E})(\mathbf{y} + \Delta\mathbf{y}) = (t + \Delta t)(B_2Z + \tilde{F})(\mathbf{y} + \Delta\mathbf{y}).$$

Then there is a corresponding perturbation for the bigger problem

$$(s + \Delta s)(A + E)(\mathbf{x} + \Delta\mathbf{x}) = (t + \Delta t)(B + F)(\mathbf{x} + \Delta\mathbf{x}),$$

where

$$E = \begin{bmatrix} 0 \\ \tilde{E}Z^* \end{bmatrix}, \quad F = \begin{bmatrix} 0 \\ \tilde{F}Z^* \end{bmatrix}, \quad \mathbf{x} = Z\mathbf{y}, \quad \Delta\mathbf{x} = Z\Delta\mathbf{y}.$$

But since $Z^*Z = I$,

$$\|\Delta\mathbf{y}\|_2 = \|\Delta\mathbf{x}\|_2 \quad \text{and} \quad \|E\|_F^2 + \|F\|_F^2 = \|\tilde{E}\|_F^2 + \|\tilde{F}\|_F^2;$$

i.e., the perturbations are equal in size.

So if the computed eigenvector $\hat{\mathbf{y}}$ is exact for a small perturbation of the reduced generalized eigenvalue problem, then the computed vector $\hat{\mathbf{x}}$ is exact for a small perturbation of the bigger generalized eigenvalue problem. If we want to be absolutely precise, then we have to note that there is an error in the computation of Z and in the multiplication of A_2 and B_2 by Z . But these errors are small and can be absorbed in \tilde{E} and \tilde{F} , so we can assume Z is exact in the analysis above. There is also an error when we multiply $\hat{\mathbf{y}}$ by the computed Z , but again this is a small error.

9. CLOSING REMARKS

Our analysis shows that Macaulay's algorithm, modified with our choice of dropped rows, is conditionally stable, i.e., with high probability for a well conditioned problem instance, it returns a good answer.

The bound for the eigenvector condition number given in Theorem 15 is too weak to be useful. It is very crude because of several contributing factors. The resultant theory is based on determinants, and converting between condition numbers and determinants does not produce good bounds. Even though the coordinate transformation used in the analysis was unitary, it caused the resultant matrices to be changed in a nonunitary way. But the biggest weakness in the bound is that we could not establish an explicit relation between the coefficients of R_0 and the subdeterminants of the top part of A , but had to settle for establishing a relation through Hilbert's Nullstellensatz that is not in closed form.

We have implemented the algorithm described herein as the solver for operations with curved parametric surface patches in the QMG mesh generation algorithm

[23, 31]. Our experience with this algorithm indicates that for well-conditioned problems it returns a highly accurate answer—far more accurate than the bound we have established. The algorithm accurately solves thousands of polynomial systems during a typical run of the mesh generator. We have tried degrees up to 6.

Even though our bound seems to be unrealistically weak, our result gives an idea of what conditions need to be satisfied so that the algorithm will give accurate roots. Those are:

- (1) The top part of the matrix A has to be well conditioned, which means the polynomials F_1 and F_2 are not close to having a common factor.
- (2) The Jacobian at the root cannot be too small (relative to the coefficients of the polynomials).
- (3) The sweepline has to be chosen to avoid being close to going through two or more roots simultaneously. If the other two conditions are being met, then the set of bad choices should be small, so there is a high probability of picking a good sweepline.

A more desirable error analysis would be to show that our modified algorithm is *backwards stable*, meaning that the computed roots are the exact roots of a nearby polynomial system. Unfortunately, our experiments with QMG indicate that it is not backwards stable. In our experiments, the computed solution for an ill-conditioned problem has a large residual. A large residual means the algorithm is not backwards stable.

ACKNOWLEDGMENT

We thank the anonymous referee for suggesting the simpler proofs of Theorems 5 and 7 presented here and for several other suggestions that improved the quality and clarity of the paper.

REFERENCES

1. W. Auzinger and H. J. Stetter, *An elimination algorithm for the computation of all zeros of a system of multivariate polynomial equation*, Numerical mathematics, Singapore 1988, ISNM vol. 86, Birkhäuser, 1988, pp. 11–30. MR91g:65112
2. L. Busé, M. Elkadi, and B. Mourrain, *Resultant over the residual of a complete intersection*, Journal of Pure and Applied Algebra **164** (2001), 35–57. MR2002h:13042
3. Arthur Cayley, *On the theory of elimination*, Cambridge and Dublin Math. J. **III** (1848), 116–120, Can also be found as Appendix B of [10].
4. M. Chardin, *Multivariate subresultants*, Journal of Pure and Applied Algebra **101** (1995), 129–138. MR96j:12002
5. Robert M. Corless, Patrizia M. Gianni, and Barry M. Trager, *A reordered Schur factorization method for zero-dimensional polynomial systems with multiple roots*, ISSAC '97. Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation (Wolfgang W. Küchlin, ed.), ACM Press, 1997, pp. 133–140.
6. David Cox, John Little, and Donal O'Shea, *Using algebraic geometry*, Springer-Verlag, 1998. MR99h:13033
7. J. Dennis and R. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, SIAM, Philadelphia, 1996. MR96i:90002
8. Alan Edelman and H. Murakami, *Polynomial roots from companion matrix eigenvalues*, Math. Comp. **64** (1995), no. 210, 763–776. MR95f:65075
9. Ioannis Z. Emiris and Bernard Mourrain, *Matrices in elimination theory*, J. Symbolic Comput. **28** (1999), no. 1–2, 3–44. MR2000h:68249
10. I. M. Gelfand, M. M. Kapranov, and A. V. Zelevinsky, *Discriminants, resultants, and multi-dimensional determinants*, Birkhäuser, 1994. MR95e:14045

11. Gene H. Golub and Charles F. Van Loan, *Matrix computations*, 3rd ed., John Hopkins University Press, 1996. MR97g:65006
12. I. S. Gradshteyn and I. M. Ryzhik, *Table of integrals, series, and products*, 5th ed., Academic Press, 1994. MR97c:00014
13. Ming Gu and Stanley C. Eisenstat, *Efficient algorithms for computing a strong rank-revealing QR factorization*, SIAM J. Sci. Comput. **17** (1996), no. 4, 848–869. MR97h:65053
14. Thomas Hawkins, *Another look at Cayley and the theory of matrices*, Arch. Internat. Historie Sci. **27** (1977), no. 100, 82–112. MR58:10105
15. G. Jónsson, *Eigenvalue methods for accurate solution of polynomial equations*, Ph.D. thesis, Center for Applied Mathematics, Cornell University, Ithaca, NY, 2001.
16. G. F. Jónsson and S. A. Vavasis, *Solving polynomials with small leading coefficients*, to appear in SIAM J. Matrix Anal. Appl.
17. T. Krick, L. Pardo, and M. Sombra, *Sharp estimates for the arithmetic nullstellensatz*, Duke Math. J. **109** (2001), 521–598. MR2002h:11060
18. T. Y. Li, *Numerical solutions of multivariate polynomial systems by homotopy continuation methods*, Acta Numerica **6** (1997), 399–436. MR2000i:65084
19. F. S. Macaulay, *On some formulæ in elimination*, Proc. London Math. Soc. **35** (1902), 3–27.
20. ———, *The algebraic theory of modular systems*, Cambridge University Press, 1916.
21. Dinesh Manocha and James Demmel, *Algorithms for intersecting parametric and algebraic curves I: Simple intersections*, ACM Trans. Graphics **13** (1994), no. 1, 73–100.
22. Dinesh Manocha and Shankar Krishnan, *Solving algebraic systems using matrix computations*, SIGSAM Bulletin **30** (1996), no. 4, 4–21.
23. Scott A. Mitchell and Stephen A. Vavasis, *Quality mesh generation in higher dimensions*, SIAM J. Computing **29** (2000), no. 4, 1334–1370. MR2000m:65146
24. Bernard Mourrain and Victor Y. Pan, *Multivariate polynomials, duality, and structured matrices*, J. Complexity **16** (2000), no. 1, 110–180. MR2002c:15048
25. J. Rosen, H. Park, and J. Glick, *Total least norm formulation and solution for structured problems*, SIAM J. Matrix Anal. App. **17** (1996), 110–126. MR96m:65046
26. M. Sombra, *Private email communication*, 2003.
27. G. W. Stewart, *Perturbation theory for the generalized eigenvalue problem*, Recent Advances in Numerical Analysis (Carl de Boor and Gene H. Golub, eds.), Academic Press, 1978, pp. 193–206. MR80c:65092
28. Kim-Chuan Toh and Lloyd N. Trefethen, *Pseudozeros of polynomials and pseudospectra of companion matrices*, Numer. Math. **68** (1994), 403–425. MR95m:65085
29. B. L. van der Waerden, *Modern algebra*, vol. II, Frederick Ungar Publishing Co., 1950, English translation of *Moderne Algebra*, Springer-Verlag, Berlin, 1931.
30. Paul Van Dooren and Patrick Dewilde, *The eigenstructure of an arbitrary polynomial matrix: Computational aspects*, Lin. Alg. Appl. **50** (1983), 545–579. MR84j:15009
31. Stephen A. Vavasis, *QMG: Software for finite-element mesh generation*, See <http://www.cs.cornell.edu/home/vavasis/qmg-home.html>, 1996.
32. Layne T. Watson, Maria Sosonkina, Robert C. Melville, Alexander P. Morgan, and Homer F. Walker, *Algorithm 777. HOMPACK90: A suite of Fortran 90 codes for globally convergent homotopy algorithms*, ACM Trans. Math. Software **23** (1997), no. 4, 514–549.

CENTER FOR APPLIED MATHEMATICS, RHODES HALL, CORNELL UNIVERSITY, ITHACA, NEW YORK 14853

Current address: deCODE Genetics, Lynghals 1, IS-110 Reykjavik, Iceland

E-mail address: gfj@decode.is

DEPARTMENT OF COMPUTER SCIENCE, UPSON HALL, CORNELL UNIVERSITY, ITHACA, NEW YORK 14853

E-mail address: vavasis@cs.cornell.edu