The Method of Gauss-Newton to Compute Power Series Solutions of Polynomial Homotopies*

Nathan Bliss Jan Verschelde

University of Illinois at Chicago
Department of Mathematics, Statistics, and Computer Science
851 S. Morgan Street (m/c 249), Chicago, IL 60607-7045, USA
{nbliss2,janv}@uic.edu

Abstract

We consider the extension of the method of Gauss-Newton from complex floating-point arithmetic to the field of truncated power series with complex floating-point coefficients. With linearization we formulate a linear system where the coefficient matrix is a series with matrix coefficients, and provide a characterization for when the matrix series is regular based on the algebraic variety of an augmented system. The structure of the linear system leads to a block triangular system. In the regular case, solving the linear system is equivalent to solving a Hermite interpolation problem. In general, we solve a Hermite-Laurent interpolation problem, via a lower triangular echelon form on the coefficient matrix. We show that this solution has cost cubic in the problem size. With a few illustrative examples, we demonstrate the application to polynomial homotopy continuation.
Key words and phrases. Linearization, Gauss-Newton, Hermite interpolation, polynomial homotopy, power series.

1 Introduction

1.1 Preliminaries

A polynomial homotopy is a family of polynomial systems which depend on one parameter. Numerical continuation methods to track solution paths defined by a homotopy are classical, see e.g.: [3] and [23]. Our goal is to improve the algorithms to track solution paths in two ways:

*This material is based upon work supported by the National Science Foundation under Grant No. 1440534. Date: 30 June 2017.

1. Polynomial homotopies define deformations of polynomial systems starting at generic instances and moving to specific instances. Tracking solution paths that start at singular solutions is not supported by current polynomial homotopy software systems.

2. To predict the next solution along a path, the current path trackers apply extrapolation methods on each coordinate of the solution separately, without taking the interdependencies between the variables into account.

Problem statement. We want to define an efficient, numerically stable, and robust algorithm to compute a power series expansion for a solution curve of a polynomial system. The input is a list of polynomials in several variables and a point on a solution curve which vanishes when evaluated at each polynomial in the list. The output of the algorithm is a tuple of series in a parameter t, where t equals one of the variables in the given list of polynomials.

Background and related work. As pointed out in [7], polynomials, power series, and Toeplitz matrices are closely related. A direct method to solve block banded Toeplitz systems is presented in [10]. The book [6] is a general reference for methods related to approximations and power series. We found inspiration for the relationship between higher-order Newton-Raphson iterations and Hermite interpolation in [20]. The computation of power series is a classical topic in computer algebra [14]. The authors of [4] propose new algorithms to manipulate polynomials by values via Lagrange interpolation.

The Puiseux series field is one of the building blocks of tropical algebraic geometry [22]. In finding the right exponents of the leading powers of the Puiseux series, we rely on tropical methods [9], and in particular on the constructive proof of the fundamental theorem of tropical algebraic geometry [16], see also [19] and [24].

Our contributions. Via linearization, rewriting matrices of series into series with matrix coefficients, we formulate the problem of computing the updates in Newton's method as a block structured linear algebra problem. For matrix series where the leading coefficient is regular, the solution of the block linear system satisfies the Hermite interpolation problem. For general matrix series, where several of the leading matrix coefficients may be rank deficient, Hermite-Laurent interpolation applies. We distinguish the cases using the algebraic variety of an augmented system. To solve the block diagonal linear system, we propose to reduce the coefficient matrix to a lower triangular echelon form, and we provide a brief analysis of its cost.

The source code for the algorithm presented in this paper is archived at github via our accounts `nbliss` and `janverschelde`.

Acknowledgments. We thank the organizers of the ILAS 2016 minisymposium on Multivariate Polynomial Computations and Polynomial Systems, Bernard Mourrain, Vanni Noferini, and Marc Van Barel, for giving the second author the opportunity to present this work. In addition, we are grateful to the anonymous referee who supplied many helpful remarks.

1.2 Motivating Example: Padé Approximant

One motivation for finding a series solution is that once it is obtained, one can directly compute the associated Padé approximant, which often has much better convergence properties. Padé approximants [6] are applied in symbolic deformation algorithms [18]. In this section we reproduce [6, Figure 1.1.1] in the context of polynomial homotopy continuation. Consider the homotopy

$$(1-t)(x^2-1)+t(3x^2-3/2)=0. \tag{1}$$

The function $x(t)=\left(\frac{1+t/2}{1+2t}\right)^{1/2}$ is a solution of this homotopy.

Its second order Taylor series at $t=0$ is $s(t)=1-3t/4+39t^2/32+O(t^2)$. The Padé approximant of degree one in numerator and denominator is $q(t)=\frac{1+7t/8}{1+13t/8}$. In Figure 1 we see that the series approximates the function only in a small interval and then diverges, whereas the Padé approximant is more accurate.

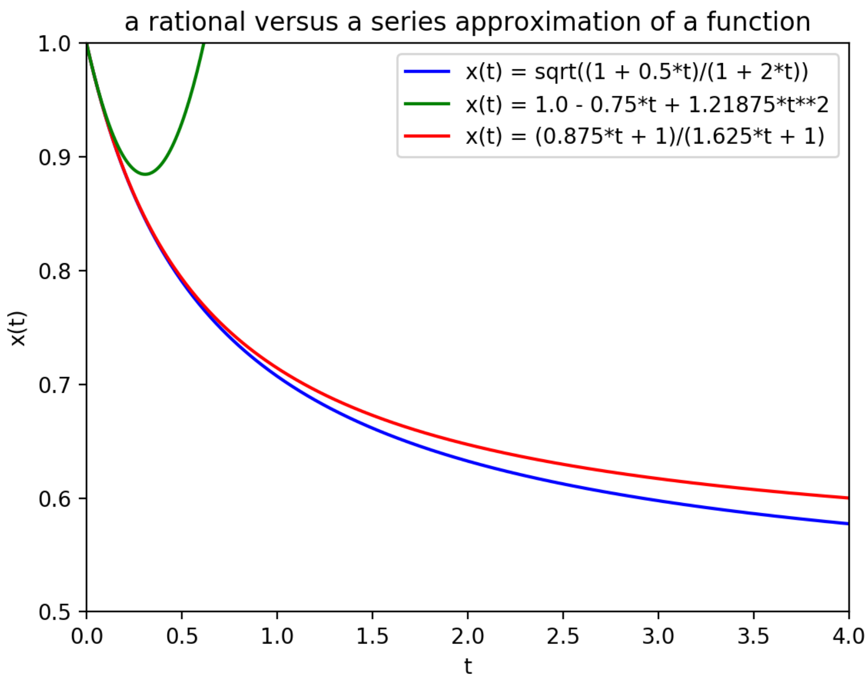

Figure 1: Comparing a Padé approximant to a series approximation.

1.3 Motivating Example: Viviani's Curve

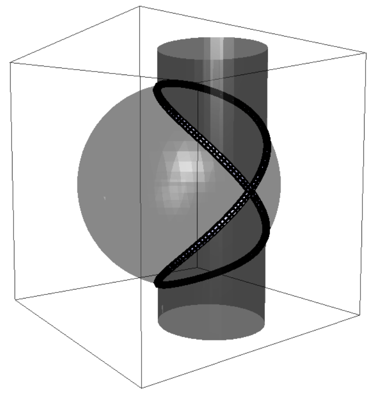

Figure 2: Viviani's Curve

Viviani's curve is defined as the intersection of the sphere $x_1^2+x_2^2+x_3^2-4=0$ and the cylinder $(x_1-1)^2+x_2^2-1=0$, shown in Figure 2. Our methods will allow us to find the Taylor series expansion around any non-isolated point of a 1-dimensional variety, assuming we have suitable starting information. For example, if we expand around the point $\mathbf{p} = (0,0,2)$ of Viviani's curve, we obtain the following series solution for x_1, x_2, x_3:

$$\begin{cases} 2t^2 \\ 2t - t^3 - \frac{1}{4}t^5 - \frac{1}{8}t^7 - \frac{5}{64}t^9 - \frac{7}{128}t^{11} - \frac{21}{512}t^{13} - \frac{33}{1024}t^{15} \\ 2 - t^2 - \frac{1}{4}t^4 - \frac{1}{8}t^6 - \frac{5}{64}t^8 - \frac{7}{128}t^{10} - \frac{21}{512}t^{12} - \frac{33}{1024}t^{14} - \frac{429}{16384}t^{16} \end{cases} \quad (2)$$

This solution is plotted in Figure 3 for a varying number of terms. To check the correctness, we can substitute (2) into the original equations, obtaining $\frac{1573}{8192}t^{18} + O(t^{20})$ and $\frac{429}{4096}t^{18} + O(t^{20})$, respectively. The vanishing of the lower-order terms confirms that we have indeed found an approximate series solution.

2 The Problem and Our Solution

2.1 Problem Setup

Let $\mathbf{f} = (f_1, f_2, \dots, f_m) \subseteq \mathbb{C}[t, x_1, \dots, x_n]$ such that the solution variety $\mathbb{V}(\mathbf{f})$ is 1-dimensional. Define $\tilde{\mathbf{f}} = (\tilde{f}_1, \tilde{f}_2, \dots, \tilde{f}_m)$ to be the image of $\mathbf{f}$ under the natural embedding $\mathbb{C}[t, x_1, \dots, x_n] \hookrightarrow \mathbb{C}((t))[x_1, \dots, x_n]$. We wish to solve the system $\tilde{\mathbf{f}}$, or in other words, compute truncated series solutions of $\mathbf{f}$ with t seen as the series parameter. One could of course use any variable; t is merely chosen for simplicity and without loss of generality. If there is a point $\mathbf{p} = (p_0, \dots, p_n) \in \mathbb{V}(\mathbf{f})$ with $p_0 = 0$, our series solution will correspond to the Taylor expansion of $\mathbf{f}$ around

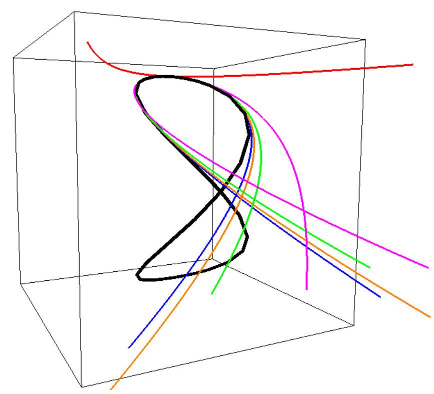

Figure 3: Viviani's curve, with improving series approximations

that point. If no such $\mathbf{p}$ exists, our solution will be a Laurent series expansion around a point at infinity.

From an algebraic perspective, the more natural place to solve $\tilde{\mathbf{f}}$ is over the field of Puiseux series. This field is informally defined as that of fractional power series whose exponents have bounded denominators, and it is algebraically closed by the Newton-Puiseux theorem. In our computations we will be content, however, to work over the ring of formal power series $\mathbb{C}[[t]]$ or the field of formal Laurent series $\mathbb{C}((t))$. This is because a suitable substitution $t \to t^k$ in the original equations, where k is the ramification index of the curve, removes any exponent denominators from all intermediate computations as well as from the resulting solution.

The next necessary ingredient is the Jacobian matrix. Let $J_{\mathbf{f}}$ be the Jacobian of $\mathbf{f}$ with respect to $t, x_1, \ldots, x_n$ and let $J_{\tilde{\mathbf{f}}}$ be the Jacobian of $\tilde{\mathbf{f}}$ with respect to $x_1, \ldots, x_n$. In other words, $J_{(\cdot)}$ as a function returns the Jacobian with respect to the variables in the polynomial ring of its input. The distinction may seem trivial, but it is important to distinguish carefully between the two, as some arguments will require $J_{\mathbf{f}}$, while $J_{\tilde{\mathbf{f}}}$ is the Jacobian used in the Newton iteration.

This brings us to our approach, which is to use Newton iteration on the system $\tilde{\mathbf{f}}$. Namely, given some starting $\mathbf{z} \in \mathbb{C}((t))^n$, we will solve

$$J_{\tilde{\mathbf{f}}}(\mathbf{z})\Delta\mathbf{z} = -\tilde{\mathbf{f}}(\mathbf{z}) \tag{3}$$

for the update $\Delta\mathbf{z}$ to $\mathbf{z}$. This is a system of equations that is linear over $\mathbb{C}((t))$, so the problem is well-posed. From a computational perspective, one could simply overload the operators on (truncated) power series and apply basic linear algebra techniques. Such an approach, however, is computationally expensive; the main point of our paper is that it can be avoided, which we will demonstrate in Section 2.2.

The last item of setup is the choice of the initial $\mathbf{z} \in \mathbb{C}((t))^n$ to begin the Newton iteration. Let $\mathbf{f}_{\text{aug}}$ be the augmented system $(t, f_1, f_2, \ldots, f_m) \subseteq \mathbb{C}[t, x_1, \ldots, x_n]$.

Remark 1. *There are three possible scenarios for $\mathbb{V}(\mathbf{f}_{\text{aug}})$:*

1. $\exists \mathbf{p} \in \mathbb{V}(\mathbf{f}_{\text{aug}})$ *nonsingular,*
2. $\exists \mathbf{p} \in \mathbb{V}(\mathbf{f}_{\text{aug}})$ *singular, or*
3. $\nexists \mathbf{p} \in \mathbb{V}(\mathbf{f}_{\text{aug}})$

In the first case, we can simply use $(t, p_1, p_2, \ldots, p_n)$ to start the Newton iteration. In the second, we must defer to tropical methods in order to obtain the necessary starting $\mathbf{z}$, which will lie in $\mathbb{C}[[t]]^n$. In the final case, we also defer to tropical methods, which provide a starting $\mathbf{z}$ that will be an element of $\mathbb{C}((t))^n$. In this case, a change of coordinates brings the problem back into one of the first two cases depending on whether the point at infinity is singular, and we can apply our method directly. It is important to note that $\mathbf{p}$ may be a nonsingular point of $\mathbb{V}(\mathbf{f})$ but a singular point of $\mathbb{V}(\mathbf{f}_{\text{aug}})$. This is the case for Viviani's curve, which we will revisit in Section 2.2.

2.2 The Linearized Newton Step

Solving the Newton step (3) amounts to solving a linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{4}$$

over the field $\mathbb{C}((t))$. Our first step is linearization — turning the vector of series into a series of vectors, and likewise for the matrix series. In other words, we refactor the problem and think of $\mathbf{x}$ and $\mathbf{b}$ as in $\mathbb{C}^n((t))$ instead of $\mathbb{C}((t))^n$, and $\mathbf{A}$ as in $\mathbb{C}^{n\times n}((t))$ instead of $\mathbb{C}((t))^{n\times n}$.

Suppose that a is the lowest order of a term in $\mathbf{A}$, and b the lowest order of a term in $\mathbf{b}$. Then we can write the linearized

$$\mathbf{A} = A_0 t^a + A_1 t^{a+1} + \ldots, \tag{5}$$

$$\mathbf{b} = \mathbf{b}_0 t^b + \mathbf{b}_1 t^{b+1} + \ldots, \text{ and} \tag{6}$$

$$\mathbf{x} = \mathbf{x}_0 t^{b-a} + \mathbf{x}_1 t^{b-a+1} + \ldots \tag{7}$$

where $A_i \in \mathbb{C}^{n\times n}$ and $\mathbf{b}_i, \mathbf{x}_i \in \mathbb{C}^n$. Expanding and equating powers of t, the linearized version of (4) is therefore equivalent to solving

$$\begin{array}{rcl} A_0\mathbf{x}_0 &=& \mathbf{b}_0 \\ A_0\mathbf{x}_1 &=& \mathbf{b}_1 - A_1\mathbf{x}_0 \\ A_0\mathbf{x}_2 &=& \mathbf{b}_2 - A_1\mathbf{x}_1 - A_2\mathbf{x}_0 \\ &\vdots& \\ A_0\mathbf{x}_d &=& \mathbf{b}_d - A_1\mathbf{x}_{d-1} - A_2\mathbf{x}_{d-2} - \cdots - A_d\mathbf{x}_0 \end{array} \tag{8}$$

for some d. This can be written in block matrix form as

$$\begin{bmatrix} A_0 & & & & \\ A_1 & A_0 & & & \\ A_2 & A_1 & A_0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ A_d & A_{d-1} & A_{d-2} & \cdots & A_0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_d \end{bmatrix} = \begin{bmatrix} \mathbf{b}_0 \\ \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_d \end{bmatrix}. \tag{9}$$

Next we characterize when such a Newton step has a unique solution.

Proposition 2.1. *Let $\mathbf{p} \in \mathbb{V}(\mathbf{f})$, $p_0 = 0$. Then $\mathbf{p}$ is a nonsingular point of $\mathbb{V}(\mathbf{f}_{\text{aug}})$ if and only if, for every step of Newton's method applied to $(t, p_1, \ldots, p_n)$, $a = 0$ and A_0 is full rank.*

Proof. ($\Rightarrow$) $\mathbf{p}$ is a nonsingular point of $\mathbf{f}_{\text{aug}}$ if and only if $J_{\mathbf{f}_{\text{aug}}}(\mathbf{p})$ is full rank. But note that $J_{\mathbf{f}_{\text{aug}}}$ is

$$\begin{bmatrix} 1 & 0 & \cdots & 0 \\ df_1/dt & df_1/dx_1 & \cdots & df_1/dx_n \\ df_2/dt & df_2/dx_1 & \cdots & df_2/dx_n \\ \vdots & \vdots & & \vdots \\ df_m/dt & df_m/dx_1 & \cdots & df_m/dx_n \end{bmatrix}. \tag{10}$$

and $J_{\tilde{\mathbf{f}}}$ is

$$\begin{bmatrix} df_1/dx_1 & \cdots & df_1/dx_n \\ df_2/dx_1 & \cdots & df_2/dx_n \\ \vdots & & \vdots \\ df_m/dx_1 & \cdots & df_m/dx_n \end{bmatrix}. \tag{11}$$

So $J_{\tilde{\mathbf{f}}}$ is full rank at $\mathbf{p}$ if and only if $J_{\mathbf{f}_{\text{aug}}}$ is. Thus it suffices to show that after each Newton step, $a = 0$ and $\mathbf{x}(0) = \mathbf{p}$ remain true, so that $A_0 = J_{\tilde{\mathbf{f}}}(\mathbf{x}(0)) = J_{\tilde{\mathbf{f}}}(\mathbf{p})$ continues to be full rank.

We clearly have $a \geq 0$ at every step, since the Newton iteration cannot introduce negative exponents. At the beginning, $a = 0$ and $\mathbf{x}(0) = \mathbf{p}$ hold trivially. Inducting on the Newton steps, if $a = 0$ and $\mathbf{x}(0) = \mathbf{p}$ at some point in the algorithm, then the next A_0, namely $J_{\tilde{\mathbf{f}}}(\mathbf{x}(0)) = J_{\tilde{\mathbf{f}}}(\mathbf{p})$, is the same matrix as in the last step, hence it is again nonsingular and a is 0. Since $\tilde{\mathbf{f}}(\mathbf{x}(0)) = \tilde{\mathbf{f}}(\mathbf{p}) = 0$, b must be strictly greater than 0. Thus the next Newton update $\Delta\mathbf{x}$ must have positive degree in all components, leaving $\mathbf{x}(0) = \mathbf{p}$ unchanged.

($\Leftarrow$) If $\mathbf{p}$ is a singular point of $\mathbb{V}(\mathbf{f}_{\text{aug}})$, then on the first Newton step $A_0 = J_{\tilde{\mathbf{f}}}(\mathbf{p})$ must drop rank by the same argument above comparing (10) and (11). □

Example 1 (The nonsingular case)**.** Let

$$\mathbf{f} = (2x_1^2 + x_1x_2 - x_3 + 1, x_2^3 - 4x_1^2 + x_1x_3 + 2x_1 - 1), \tag{12}$$

and let $\mathbf{p} = (0,1,1) \in \mathbb{V}(\mathbf{f})$. At $\mathbf{p}$, the augmented Jacobian $J_{\mathbf{f}_{\text{aug}}}$ is

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & -1 \\ 3 & 3 & 0 \end{bmatrix}, \tag{13}$$

so we are in the nonsingular case of Remark 1. The first Newton step, $J_{\tilde{\mathbf{f}}}(\mathbf{z})\Delta\mathbf{z} = -\tilde{\mathbf{f}}(\mathbf{z})$, can be written as

$$\begin{bmatrix} x_1 & -1 \\ 3 & x_1 \end{bmatrix} \Delta\mathbf{z} = -\begin{bmatrix} x_1 + 2x_1^2 \\ 3x_1 - 4x_1^2 \end{bmatrix}. \tag{14}$$

To put in linearized form, we have $a = 0$, $b = 1$,

$$A_0 = \begin{bmatrix} 0 & -1 \\ 3 & 0 \end{bmatrix}, A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{15}$$

$$\mathbf{b}_0 = \begin{bmatrix} -1 \\ -3 \end{bmatrix}, \text{ and } \mathbf{b}_1 = \begin{bmatrix} -2 \\ 4 \end{bmatrix}. \tag{16}$$

Since A_0 is nonsingular, we can solve (8), which yields the next term:

$$\Delta\mathbf{z} = \begin{bmatrix} -x_1 \\ x_1 \end{bmatrix}. \tag{17}$$

After another iteration, our series solution is

$$\begin{bmatrix} 1 - x_1 \\ 1 + x_1 + x_1^2 \end{bmatrix}. \tag{18}$$

In fact this is the entire series solution for $\mathbf{f}$ — substituting (18) into $\mathbf{f}$ causes both polynomials to vanish completely. △

We constructed the example above so its solution is a series with finitely many terms, a polynomial. The solution of (4) can be interpreted as the solution obtained via Hermite interpolation. Observe that for a series

$$\mathbf{x}(t) = \mathbf{x}_0 + \mathbf{x}_1 t + \mathbf{x}_2 t^2 + \mathbf{x}_3 t^3 + \cdots + \mathbf{x}_k t^k + \cdots \tag{19}$$

its Maclaurin expansion is

$$\mathbf{x}(t) = \mathbf{x}(0) + \mathbf{x}'(0)t + \frac{1}{2}\mathbf{x}''(0)t^2 + \frac{1}{3!}\mathbf{x}'''(0)t^3 + \cdots + \frac{1}{k!}\mathbf{x}^{(k)}(0)t^k + \cdots \tag{20}$$

where $\mathbf{x}^{(k)}(0)$ denotes the k-th derivative of $\mathbf{x}(t)$ evaluated at zero. Then:

$$\mathbf{x}_k = \frac{1}{k!}\mathbf{x}^{(k)}(0), \quad k = 0, 1, \ldots \tag{21}$$

Solving (4) up to degree d implies that all derivatives up to degree d of $\mathbf{x}(t)$ at $t = 0$ match the solution. If the solution is a polynomial, then this polynomial will be obtained if (4) is solved up to the degree of the polynomial.

Example 2 (Viviani, continued)**.** In Section 1.3 we introduced the example of Viviani's curve, given by the system

$$\mathbf{f} = (x_1^2 + x_2^2 + x_3^2 - 4, (x_1 - 1)^2 + x_2^2 - 1). \tag{22}$$

For the augmented system $\mathbf{f}_{\text{aug}}$, the Jacobian $J_{\mathbf{f}_{\text{aug}}}$ is

$$\begin{bmatrix} 1 & 0 & 0 \\ 2x_1 & 2x_2 & 2x_3 \\ 2x_1 - 2 & 2x_2 & 0 \end{bmatrix} \tag{23}$$

which at the point $\mathbf{p} = (0, 0, 2)$ becomes

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 4 \\ -2 & 0 & 0 \end{bmatrix}. \tag{24}$$

This matrix drops rank, hence $\mathbf{p}$ is a singular point of $\mathbf{f}_{\text{aug}}$ and we are in the second case of Remark 1. Following Section 2.1, we defer to tropical methods to begin, obtaining the transformation $x_1 \to 2t^2$ and the starting term $\mathbf{z} = (2t, 2)$. Now the first Newton step can be written:

$$\begin{bmatrix} 4t & 4 \\ 4t & 0 \end{bmatrix} \Delta\mathbf{z} = - \begin{bmatrix} 4t^2 + 4t^4 \\ 4t^4 \end{bmatrix}. \tag{25}$$

Note that $J_{\tilde{\mathbf{f}}}(\mathbf{z})$ is now invertible over $\mathbb{C}((t))$. Its inverse begins with negative exponents of t:

$$\begin{bmatrix} 0 & 1/4 \\ 1/4\, t^{-1} & -1/4\, t^{-1} \end{bmatrix}. \tag{26}$$

To linearize, we first observe that $a = 0$ and $b = 2$, so $\mathbf{x}$ will have degree at least $b - a = 2$. The linearized block form of (25) is then

$$\left[\begin{array}{cc|cc|cc} 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 4 & 0 & 0 & 4 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 4 & 0 & 0 & 4 \\ 0 & 0 & 4 & 0 & 0 & 0 \end{array}\right] \mathbf{x} = \begin{bmatrix} -4 \\ 0 \\ 0 \\ 0 \\ -4 \\ -4 \end{bmatrix}. \tag{27}$$

Whether we solve (25) over $\mathbb{C}((t))$ or solve (27) in the least squares sense, we obtain the same Newton update

$$\Delta\mathbf{z} = \begin{bmatrix} -t^3 \\ -t^2 \end{bmatrix}. \tag{28}$$

Substituting $\mathbf{z} + \Delta\mathbf{z} = (2t - t^3, 2 - t^2)$ into (22) produces $(x_1^6 + x_1^4, x_1^6)$, and we have obtained the desired cancellation of lower-order terms. △

The matrix in (27) we call a Hermite-Laurent matrix, because its correspondence with Hermite-Laurent interpolation.

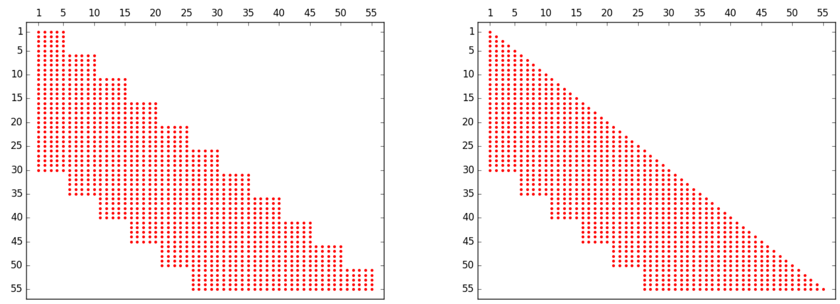

Figure 4: The banded block structure of a generic Hermite-Laurent matrix for $n = 5$ at the left, with at the right its lower triangular echelon form.

2.3 A Lower Triangular Echelon Form

When we are in the regular case of Remark 1 and the condition number of A_0 is low, we can simply solve the staggered system (8). When this is not possible, we are forced to solve (9). Figure 4 shows the structure of the coefficient matrix (9) for the regular case, when A_0 is regular and all block matrices are dense. The essence of this section is that we can use column operations to reduce the block matrix to a lower triangular echelon form as shown at the right of Figure 4, solving (9) in the same time as (8).

The lower triangular echelon form of a matrix is a lower triangular matrix with zero elements above the diagonal. If the matrix is regular, then all diagonal elements are nonzero. For a singular matrix, the zero rows of its echelon form are on top (have the lowest row index) and the zero columns are at the right (have the highest column index). Every nonzero column has one pivot element, which is the nonzero element with the smallest row index in the column. All elements at the right of a pivot are zero. Columns may need to be swapped so that the row indices of the pivots of columns with increasing column indices are sorted in decreasing order.

Example 3. (Viviani, continued). For the matrix series in (27), we have the following reduction:

$$\begin{bmatrix} 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 4 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 4 \\ 0 & 0 & 4 & 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 4 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 4 & 4 & 0 \end{bmatrix}. \qquad (29)$$

Because of the singular matrix coefficients in the series, we find zeros on the diagonal in the echelon form. △

Given a general n-by-m dimensional matrix A, the lower triangular echelon form L can be described by one n-by-n row permutation matrix P which swaps the zero rows of A and a sequence of m column permutation matrices Q_k (of dimension m) and multiplier matrices U_k (also of dimension m). The matrices Q_k define the column swaps to bring the pivots with lowest row indices to the lowest column indices. The matrices U_k contain the multipliers to reduce what is at the right of the pivots to zero. Then the construction of the lower triangular echelon form can be summarized in the following matrix equation:

$$L = PAQ_1U_1Q_2U_2 \cdots Q_mU_m. \tag{30}$$

Similar to solving a linear system with a LU factorization, the multipliers are applied to the solution of the lower triangular system which has L as its coefficient matrix.

3 Some Preliminary Cost Estimates

Working with truncated power series is somewhat similar to working with extended precision arithmetic. In this section we make some observations regarding the cost overhead.

3.1 Cost of one step

First we compare the cost of computing a single Newton step using the various methods introduced. We let d denote the degree of the truncated series in $A(t)$, and n the dimension of the matrix coefficients in $A(t)$ as before.

The staggered system. In the case that $a \geq 0$ and the leading coefficient A_0 of the matrix series $A(t)$ is regular, the equations in (8) can be solved with $O(n^3) + O(dn^2)$ operations. The cost is $O(n^3)$ for the decomposition of the matrix A_0, and $O(dn^2)$ for the back substitutions using the decomposition of A_0 and the convolutions to compute the right hand sides.

The big block matrix. Ignoring the triangular matrix structure, the cost of solving the larger linear system (9) is $O((dn)^3)$.

The lower triangular echelon version. If the leading coefficient A_0 in the matrix series is regular (as illustrated by Figure 4), we may copy the lower triangular echelon form $L_0 = A_0Q_0U_0$ of A_0 to all blocks on the diagonal and apply the permutation Q_0 and column operations as defined by U_0 to all other column blocks in A. The regularity of A_0 implies that we may use the lower triangular echelon form of L_0 to solve (9) with substitution. Thus with this quick optimization we obtain the same cost as solving the staggered system (8).

In general, in case A_0 and several other matrix coefficients may be rank deficient, the diagonal of nonzero pivot elements will shift towards the bottom of L. We then find as solutions vectors in the null space of the upper portion of the matrix A.

3.2 Cost of computing D terms

Assume that $D = 2^k$. In the regular case, assuming quadratic convergence, it will take k steps to compute 2^k terms. We can reuse the factorization of A_0 at each step, so we have $O(n^3)$ for the decomposition plus

$$O(2n^2 + 4n^2 + 8n^2 + \dots 2^{k-1}n^2) = O(2^k n^2) \qquad (31)$$

for the back substitutions. Putting these together, we find the cost of computing D terms to be $O(n^3) + O(Dn^2)$.

4 Computational Experiments

Our power series methods have been implemented in PHCpack [28] and are available to the Python programmer via phcpy [29]. We used the computer algebra system Sage [27].

4.1 The Problem of Apollonius

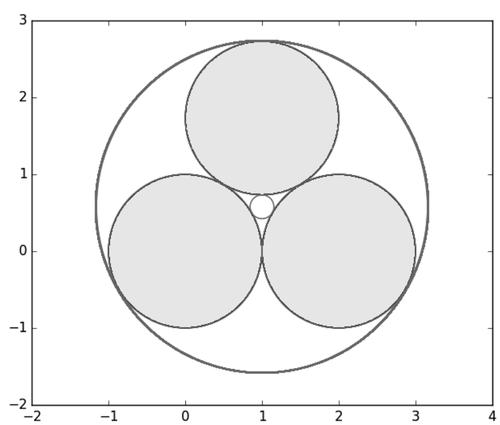

Figure 5: Singular configuration of Apollonius circles. Original circles are filled in, solution circles are dark gray.

The classical problem of Apollonius consists in finding all circles that are simultaneously tangent to three given circles. A special case is when the three circles are mutually tangent and have the same radius; see Figure 4.1. Here the solution variety is singular – the circles themselves are double solutions. In this figure, all have radius 3, and centers $(0, 0)$, $(2, 0)$, and $(1, \sqrt{3})$. We can study this configuration with power series techniques by introducing a parameter t to represent a vertical shift of the third circle, and then examining the solutions as we vary t. This is represented algebraically as a solution to

$$\left\{\begin{array}{rcl} x_1^2+x_2^2-r^2-2r-1 &=& 0 \\ x_1^2+x_2^2-r^2-4x_1-2r+3 &=& 0 \\ t^2+x_1^2-2tx_2+x_2^2-r^2+2\sqrt{3}t-2x_1-2\sqrt{3}x_2+2r+3 &=& 0. \end{array}\right. \tag{32}$$

Because we are interested in power series solutions of (32) near $t=0$, we use t as our free variable. To simplify away the $\sqrt{3}$, we substitute $t \to \sqrt{3}t$, $x_2 \to \sqrt{3}x_2$, and the system becomes

$$\left\{\begin{array}{rcl} x_1^2+3x_2^2-r^2-2r-1 &=& 0 \\ x_1^2+3x_2^2-r^2-4x_1-2r+3 &=& 0 \\ 3t^2+x_1^2-6tx_2+3x_2^2-r^2+6t-2x_1-6x_2+2r+3 &=& 0. \end{array}\right. \tag{33}$$

Call this system $\mathbf{f}$. Now we examine the system at $(t,x_1,x_2,r)=(0,1,1,1)=\mathbf{p}$. The Jacobian $J_{\mathbf{f}}$ at $\mathbf{p}$ is

$$\begin{bmatrix} 0 & 2 & 6 & -4 \\ 0 & -2 & 6 & -4 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{34}$$

so $\mathbf{f}$ — and by extension $\mathbf{f}_{\text{aug}}$ — is singular at $\mathbf{p}$, and we are in the second case of Remark 1. Tropical methods give two possible starting solutions, which rounded for readability are $(t,1,1+0.536t,1+0.804t)$ and $(t,1,1+7.464t,1+11.196t)$. We will continue with the second; call it $\mathbf{z}$. For the first step of Newton's method, $\mathbf{A}$ is

$$\begin{bmatrix} 2 & 6 & -4 \\ -2 & 6 & -4 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 44.785 & -22.392 \\ 0 & 44.785 & -22.392 \\ 0 & 38.785 & -22.392 \end{bmatrix} t \tag{35}$$

and $\mathbf{b}$ is

$$\begin{bmatrix} 41.785 \\ 41.785 \\ 0 \end{bmatrix} t^2. \tag{36}$$

From these we can construct the linearized system:

$$\begin{bmatrix} A_0 & & \\ A_1 & A_0 & \\ & A_1 & A_0 \end{bmatrix} \Delta\mathbf{z} = \begin{bmatrix} \mathbf{b}_0 \\ 0 \\ 0 \end{bmatrix}. \tag{37}$$

Solving in the least squares sense, we obtain two more terms of the series, so in total we have

$$\left\{\begin{array}{rcl} x_1 &=& 1 \\ x_2 &=& 1+7.464t+45.017t^2+290.992t^3 \\ r &=& 1+11.196t+77.971t^2+504.013t^3. \end{array}\right. \tag{38}$$

From this, we can get a better idea of what happens near $t=0$. For example, in Figure 4.1 one can see the solutions of the system at $t=0.13$. The series help to explain why some solution circles of the perturbation of the special configuration grow faster than others.

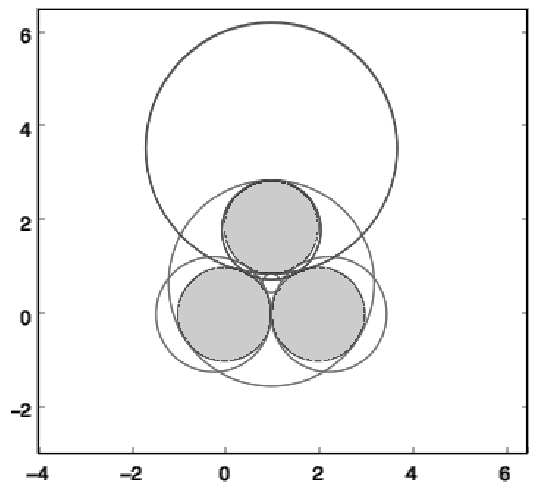

Figure 6: Solution to (32) for $t = 0.13$.

4.2 Tangents to Four Spheres

Our next example is that of finding all lines mutually tangent to four spheres in $\mathbb{R}^3$; see [12], [21], [25], and [26]. If a sphere S has center $\mathbf{c}$ and radius r, the condition that a line in $\mathbb{R}^3$ is tangent to S is given by

$$\|\mathbf{m} - \mathbf{c} \times \mathbf{t}\|^2 - r^2 = 0, \tag{39}$$

where $\mathbf{m} = (x_0, x_1, x_2)$ and $\mathbf{t} = (x_3, x_4, x_5)$ are the moment and tangent vectors of the line, respectively. For four spheres, this gives rise to four polynomial equations; if we add the equation $x_0x_3 + x_1x_4 + x_2x_5 = 0$ to require that $\mathbf{t}$ and $\mathbf{m}$ are perpendicular and $x_3^2 + x_4^2 + x_5^2 = 1$ to require that $\|\mathbf{t}\| = 1$, we have a system of 6 equations in 6 unknowns which we expect to be 0-dimensional.

If we choose the centers to be $(+1, +1, +1)$, $(+1, -1, -1)$, $(-1, +1, -1)$, and $(-1, -1, +1)$ and the radii to all be $\sqrt{2}$, the spheres all mutually touch and the configuration is singular; see Figure 7. In this case, the number of solutions drops to three, each of multiplicity 4.

Next we introduce an extra parameter t to the equations so that the radii of the spheres are $\sqrt{2}+t$. This results in a 1-dimensional system F that is singular at $t = 0$, so we are once again in the second case of Remark 1. Tropical and algebraic techniques — in particular, the tropical basis [8] in Gfan [17] and the primary decomposition in Singular [11] — decompose F into three systems, one

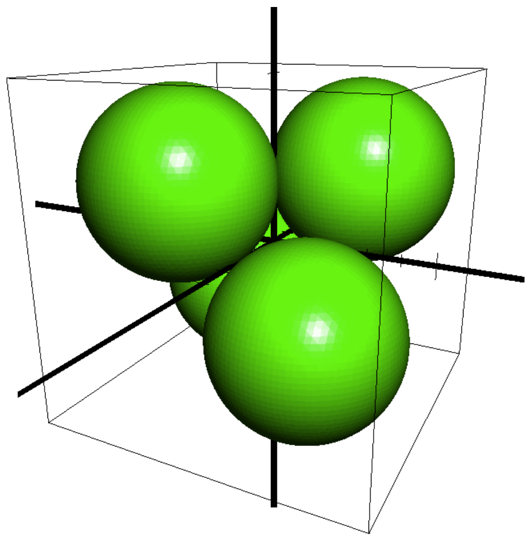

Figure 7: A singular configuration of spheres

of which is

$$\mathbf{f} = \left\{ \begin{array}{rcl} x_0 & = & 0 \\ x_3 & = & 0 \\ x_4^2 + x_2x_5 + x_5^2 & = & 0 \\ x_1x_4 + x_2x_5 & = & 0 \\ x_1x_2 - x_2x_4 + x_1x_5 & = & 0 \\ x_1^2 + x_2^2 - 1 & = & 0 \\ 2s^4 + 4s^2 + x_2x_5 & = & 0 \\ x_2^2x_4 - x_2x_4x_5 + x_1x_5^2 - x_4 & = & 0 \\ x_2^3 - x_2 - x_5 & = & 0. \end{array} \right. \quad (40)$$

Using our methods we can find several solutions to this, one of which is

$$\left\{ \begin{array}{l} x_0 = 0 \\ x_1 = 2t + 4.5t^3 + 30.9375t^5 + 299.3906t^7 + 3335.0889t^9 + 40316.851t^{11} \\ x_2 = 1 - 2t^2 - 11t^4 - 94t^6 - 986.5t^8 - 11503t^{10} \\ x_3 = 0 \\ x_4 = 2t - 3.5t^3 - 23.0625t^5 - 193.3594t^7 - 2019.3486t^9 - 23493.535t^{11} \\ x_5 = -4t^2 - 10t^4 - 64t^6 - 614t^8 - 6818t^{10} - 82283t^{12} \end{array} \right. .$$

Substituting back into $\mathbf{f}$ yields series in $O(t^{12})$, confirming the calculations. This solution could be used as the initial predictor in a homotopy beginning at the singular configuration.

4.3 Series Developments for Cyclic 8-Roots

A vector $\mathbf{u} \in \mathbb{C}^n$ of a unitary matrix A is biunimodular if for $k = 1, 2, \ldots, n$: $|u_k| = 1$ and $|v_k| = 1$ for $\mathbf{v} = A\mathbf{u}$. The following system arises in the study [13] of biunimodular vectors:

$$\mathbf{f}(\mathbf{x}) = \begin{cases} x_0 + x_1 + \cdots + x_{n-1} = 0 \\ i = 2, 3, 4, \ldots, n-1 : \displaystyle\sum_{j=0}^{n-1} \prod_{k=j}^{j+i-1} x_{k \bmod n} = 0 \\ x_0 x_1 x_2 \cdots x_{n-1} - 1 = 0. \end{cases} \tag{41}$$

Cyclic 8-roots has solution curves not reported by Backelin [5]. Note that because of the last equation, the system has no solution for $x_0 = 0$, or in other words $\mathbb{V}(\mathbf{f}_{aug}) = \emptyset$. Thus we are in the third case of Remark 1.

In [1, 2], the vector $\mathbf{v} = (1, -1, 0, 1, 0, 0, -1, 0)$ gives the leading exponents of the series. The corresponding unimodular coordinate transformation $\mathbf{x} = \mathbf{z}^M$ is

$$M = \begin{bmatrix} 1 & -1 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \begin{array}{l} x_0 \rightarrow z_0 \\ x_1 \rightarrow z_1 z_0^{-1} \\ x_2 \rightarrow z_2 \\ x_3 \rightarrow z_3 z_0 \\ x_4 \rightarrow z_4 \\ x_5 \rightarrow z_5 \\ x_6 \rightarrow z_6 z_0^{-1} \\ x_7 \rightarrow z_7. \end{array} \tag{42}$$

Solving the transformed system with z_0 set to 0 gives the leading coefficient of the series.

After 2 Newton steps with `phc -u`, the series for z_1 is

```
(-1.25000000000000E+00 + 1.25000000000000E+00*i)*z0^2
+( 5.00000000000000E-01 - 2.37676980513323E-17*i)*z0
+(-5.00000000000000E-01 - 5.00000000000000E-01*i);
```

After a third step, the series for z_1 is

```
( 7.12500000000000E+00 + 7.12500000000000E+00*i)*z0^4
+(-1.52745512076048E-16 - 4.25000000000000E+00*i)*z0^3
+(-1.25000000000000E+00 + 1.25000000000000E+00*i)*z0^2
+( 5.00000000000000E-01 - 1.45255178343636E-17*i)*z0
+(-5.00000000000000E-01 - 5.00000000000000E-01*i);
```

Bounds on the degree of the Puiseux series expansion to decide whether a point is isolated are derived in [15].

References

[1] D. Adrovic and J. Verschelde. Computing Puiseux series for algebraic surfaces. In J. van der Hoeven and M. van Hoeij, editors, *Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation (ISSAC 2012)*, pages 20–27. ACM, 2012.

[2] D. Adrovic and J. Verschelde. Polyhedral methods for space curves exploiting symmetry applied to the cyclic n-roots problem. In V.P. Gerdt, W. Koepf, E.W. Mayr, and E.V. Vorozhtsov, editors, *Computer Algebra in Scientific Computing, 15th International Workshop, CASC 2013, Berlin, Germany*, volume 8136 of *Lecture Notes in Computer Science*, pages 10–29, 2013.

[3] E. L. Allgower and K. Georg. *Introduction to Numerical Continuation Methods*, volume 45 of *Classics in Applied Mathematics*. SIAM, 2003.

[4] A. Amiraslani, R. M. Corless, L. Gonzalez-Vega, and A. Shakoori. Polynomial algebra by values. Technical report, Ontario Research Centre for Computer Algebra, 2004.

[5] J. Backelin. Square multiples n give infinitely many cyclic n-roots. Reports, Matematiska Institutionen 8, Stockholms universitet, 1989.

[6] G. A. Baker and P. Graves-Morris. *Padé Approximants*, volume 59 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, 2nd edition, 1996.

[7] D. A. Bini and B. Meini. Solving block banded block Toeplitz systems with structured blocks: Algorithms and applications. In D. A. Bini, E. Tyrtyshnikov, and P. Yalamov, editors, *Structured Matrices*, pages 21–41. Nova Science Publishers, Inc., Commack, NY, USA, 2001.

[8] T. Bogart, M. Hampton, and W. Stein. `groebner_fan` module of Sage. The Sage Development Team, 2008.

[9] T. Bogart, A. N. Jensen, D. Speyer, B. Sturmfels, and R. R. Thomas. Computing tropical varieties. *Journal of Symbolic Computation*, 42(1):54–73, 2007.

[10] A. Chesnokov and M. Van Barel. A direct method to solve block banded block Toeplitz systems with non-banded Toeplitz blocks. *Journal of Computational and Applied Mathematics*, 234(5):1485–1491, 2010.

[11] Wolfram Decker, Gert-Martin Greuel, Gerhard Pfister, and Hans Schönemann. Singular 4-1-0 — A computer algebra system for polynomial computations. `http://www.singular.uni-kl.de`, 2016.

[12] C. Durand. *Symbolic and Numerical Techniques for Constraint Solving*. PhD thesis, Purdue University, 1998.

[13] H. Führ and Z. Rzeszotnik. On biunimodular vectors for unitary matrices. *Linear Algebra and its Applications*, 484:86–129, 2015.

[14] K. O. Geddes, S. R. Czapor, and G. Labahn. *Algorithms for Computer Algebra*. Kluwer Academic Publishers, 1992.

[15] M. I. Herrero, G. Jeronimo, and J. Sabia. Puiseux expansions and nonisolated points in algebraic varieties. *Communications in Algebra*, 44(5):2100–2109, 2016.

[16] A. N. Jensen, H. Markwig, and T. Markwig. An algorithm for lifting points in a tropical variety. *Collectanea Mathematica*, 59(2):129–165, 2008.

[17] A.N. Jensen. Computing Gröbner fans and tropical varieties in Gfan. In M.E. Stillman, N. Takayama, and J. Verschelde, editors, *Software for Algebraic Geometry*, volume 148 of *The IMA Volumes in Mathematics and its Applications*, pages 33–46. Springer-Verlag, 2008.

[18] G. Jeronimo, G. Matera, P. Solernó, and A. Waissbein. Deformation techniques for sparse systems. *Found. Comput. Math.*, 9:1–50, 2009.

[19] E. Katz. A tropical toolkit. *Expositiones Mathematicae*, 27:1–36, 2009.

[20] H. T. Kung and J. F. Traub. Optimal order of one-point and multipoint iteration. *Journal of the Association of Computing Machinery*, 21(4):643–651, 1974.

[21] I. G. Macdonald, J. Pach, and T. Theobald. Common tangents to four unit balls in $\mathbb{R}^3$. *Discrete and Computational Geometry*, 26(1):1–17, 2001.

[22] D. Maclagan and B. Sturmfels. *Introduction to Tropical Geometry*, volume 161 of *Graduate Studies in Mathematics*. American Mathematical Society, 2015.

[23] A. Morgan. *Solving polynomial systems using continuation for engineering and scientific problems*, volume 57 of *Classics in Applied Mathematics*. SIAM, 2009.

[24] S. Payne. Fibers of tropicalization. *Mathematische Zeitschrift*, 262:301–311, 2009.

[25] F. Sottile. *Real Solutions to Equations from Geometry*, volume 57 of *University Lecture Series*. AMS, 2011.

[26] F. Sottile and T. Theobald. Line problems in nonlinear computational geometry. In J.E. Goodman, J. Pach, and R. Pollack, editors, *Computational Geometry - Twenty Years Later*, pages 411–432. AMS, 2008.

[27] W. A. Stein et al. *Sage Mathematics Software (Version 6.9)*. The Sage Development Team, 2015. `http://www.sagemath.org`.

[28] J. Verschelde. Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation. *ACM Trans. Math. Softw.*, 25(2):251–276, 1999.

[29] J. Verschelde. Modernizing PHCpack through phcpy. In P. de Buyl and N. Varoquaux, editors, *Proceedings of the 6th European Conference on Python in Science (EuroSciPy 2013)*, pages 71–76, 2014.