Sampling Algebraic Sets in Local Intrinsic Coordinates*

Yun Guan† Jan Verschelde‡

2 September 2011

Abstract

Numerical data structures for positive dimensional solution sets of polynomial systems are sets of generic points cut out by random planes of complementary dimension. We may represent the linear spaces defined by those planes either by explicit linear equations or in parametric form. These descriptions are respectively called extrinsic and intrinsic representations. While intrinsic representations lower the cost of the linear algebra operations, we observe worse condition numbers. In this paper we describe the local adaptation of intrinsic coordinates to improve the numerical conditioning of sampling algebraic sets. Local intrinsic coordinates also lead to a better stepsize control. We illustrate our results with Maple experiments and computations with PHCpack on some benchmark polynomial systems.

2000 Mathematics Subject Classification. Primary 65H10. Secondary 14Q99, 68W30.
Key words and phrases. algebraic sets, condition numbers, generic points, local intrinsic coordinates, numerical algebraic geometry, path tracking, polynomial systems, sampling.

1 Introduction

By $f(\mathbf{x}) = \mathbf{0}$ we denote a system of polynomials f in the variables $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. Of interest is the solution set $f^{-1}(\mathbf{0})$ as a subset of $\mathbb{C}^n$. Classical is the notion of *a generic point* of an algebraic set: any polynomial vanishing at a generic point also vanishes on any other point of the algebraic set [31, Definition 1.3].

Using continuation [1, 28, 30], the numerical treatment of positive dimensional algebraic sets was first proposed in [45] and elaborated in a series of papers by the authors of [46] and the second author, see also [43] for another introduction. In [37], an $(n-k)$-dimensional algebraic set S of degree d is represented by *a witness set* which consists of d generic points of S, in the intersection of S and a general k-dimensional linear subspace L of $\mathbb{C}^n$, where L is general by a random choice of the coefficients of its defining equations. The algorithms in numerical algebraic geometry are implemented in PHCpack [48, 49] (see also [39]) and can be executed via MATLAB

*This material is based upon work supported by the National Science Foundation under Grant No. 0713018.

†Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, 851 South Morgan (M/C 249), Chicago, IL 60607-7045, USA. email: guan@math.uic.edu URL: http://www.math.uic.edu/~guan

‡Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, 851 South Morgan (M/C 249), Chicago, IL 60607-7045, USA. email: jan@math.uic.edu URL: http://www.math.uic.edu/~jan

(or Octave) [20], Maple [26], and Macaulay 2 [18], [25]. Bertini [4, 6] is another program for numerical algebraic geometry.

If we consider the construction of a witness set for a hypersurface, given by one polynomial f in n variables $\mathbf{x}$, then we generate a system $L(\mathbf{x}) = \mathbf{0}$ of $n-1$ linear equations with random coefficients. The equations L are not homogeneous, i.e.: they appear with a (random) nonzero constant coefficient. Instead of solving $f(\mathbf{x}) = 0$, augmented with $L(\mathbf{x}) = \mathbf{0}$, we can save many operations by substituting a parametric representation $\mathbf{x} = \mathbf{b} + \mathbf{v}\xi$ for the affine space defined by $L(\mathbf{x}) = \mathbf{0}$ into f, hereby reducing the construction of a witness set for a hypersurface to solving a polynomial in one variable ξ. The variable ξ is an example of *intrinsic coordinates*, introduced in [42].

Our first goal in this paper is to show (in section 3) that a substitution of $\mathbf{x} = \mathbf{b} + \mathbf{v}\xi$ into a polynomial leads to an ill-conditioned problem, for evaluation and consequently also for root finding. In the next section we introduce *local* intrinsic coordinates and as our second contribution, comparing condition numbers we demonstrate that local intrinsic coordinates restore the numerical conditioning to the state of the original extrinsic coordinates, as explained in section 4. Thirdly, we present in section 5 an algorithm to track a solution path using local intrinsic coordinates, along with an a priori stepsize control evaluation strategy to sample a component; see [46, pages 272-273] for application to the membership test. Computational examples are discussed in section 6.

Using multiprecision arithmetic during path tracking (see e.g. [5] or [50]) may avoid these numerical instabilities, better stepsize control strategies [7] will be effective for our problems, and also the methods of [14, 33] to sample around singular points may apply. But as in dealing with the high powers of the continuation parameter of polyhedral homotopies [24], our approach in this paper is specific to the type of homotopies. Moreover, as we show in this paper, the relative condition numbers of solutions in local intrinsic coordinates may grow exponentially in the degrees of the polynomials. This exponential growth of conditioning will slow down multiprecision path trackers enormously.

While the focus of this paper is *not* on the computation of a witness set, but rather on a different numerical representation with an improved conditioning, the use of local intrinsic coordinates improves the implementation of intrinsic diagonal homotopies [42]. Diagonal homotopies are used to intersect algebraic sets and constitute a critical component in an equation-by-equation solver [44], implemented in parallel in [19].

Last, and certainly not least, we want to emphasize that our notion of numerical conditioning is algebraic, as the problems with global intrinsic coordinates stem from a bad scaling and are not caused by the proximity of a singularity, which can as well be understood pure geometrically. Even as the substitution is not performed in an explicit symbolic manner leading to a fully expanded polynomial, the evaluation in badly scaled coordinates leads to a loss of accuracy in the numerical representation of the roots. Assuming the coefficient of $f(\mathbf{x}) = \mathbf{0}$ to be well scaled and $f^{-1}(\mathbf{0})$ reduced, by definition of their generality, generic points are well-conditioned points on an algebraic set. With local intrinsic coordinates we allow only small increments of the variables and using generic points as offset points we naturally stay close to the well-conditioned locus.

Acknowledgements. We thank Professor Hiroshi Murakami for his remarks made after the presentation of the first author at the session of Symbolic and Numeric Computation at ACA

2009. His remarks led us to local intrinsic coordinates. We are grateful to the referees for their comments which helped to improve earlier versions of this paper.

2 Local Intrinsic Coordinates

A polynomial system $f(\mathbf{x}) = \mathbf{0}$, $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, defines an algebraic set $f^{-1}(\mathbf{0}) \subset \mathbb{C}^n$. The polynomials of f belong to $\mathbb{C}[\mathbf{x}]$. We assume (for simplicity of exposition throughout the paper):

1. $f^{-1}(\mathbf{0})$ is pure dimensional, k is its codimension, so $\dim(f^{-1}(\mathbf{0})) = n - k$;
2. $f(\mathbf{x}) = \mathbf{0}$ is a complete intersection, and in particular: $f = (f_1, f_2, \ldots, f_k)$;
3. $f^{-1}(\mathbf{0})$ is reduced, i.e.: of multiplicity one.

To remove the third assumption, a deflation operator [27] (see also [10]) as proposed in [46, §13.3.2] should be applied. The first two assumptions are made for notational convenience. To deal with intersections that are not complete, we refer to [36] for embeddings with slack variables, or [46, §13.5] for randomization techniques.

One important assumption on f is that its coefficients should be well scaled, i.e.: they do not take extreme values. Scaling methods are explained in [30, Chapter 5]. Related to scaling are the magnitudes of the points of $f^{-1}(\mathbf{0})$. Therefore, in addition we assume $f^{-1}(\mathbf{0})$ lives in a convenient range of the floating-point numbers, eventually after an appropriate projective transformation (also addressed in [30]). So we assume that affine values for the components of $\mathbf{x}$ stay in the vicinity[1] of the complex unit circle.

Example 2.1 One of our benchmark examples is a family of systems, defined by all adjacent minors of a general 2-by-3 matrix ([13], [22]):

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \end{bmatrix} \quad f(\mathbf{x}) = \begin{cases} x_{11}x_{22} - x_{21}x_{12} = 0 \\ x_{12}x_{23} - x_{22}x_{13} = 0. \end{cases} \tag{1}$$

For this example, we have $n = 6$, $k = 2$, and we have a complete intersection: $\dim(f^{-1}(\mathbf{0})) = n - k = 4$. To compute $\deg(f^{-1}(\mathbf{0}))$, we add $n - k$ general linear equations $L(\mathbf{x}) = \mathbf{0}$ to $f(\mathbf{x}) = \mathbf{0}$ and we solve $\{f(\mathbf{x}) = \mathbf{0}, L(\mathbf{x}) = \mathbf{0}\}$. Generic points on the solution set defined by the system for all adjacent minors of a general 2-by-3 matrix satisfy (for random coefficients $c_{ij} \in \mathbb{C}$):

$$\begin{cases} x_{11}x_{22} - x_{21}x_{12} = 0 \\ x_{12}x_{23} - x_{22}x_{13} = 0 \\ c_{10} + c_{11}x_{11} + c_{12}x_{12} + c_{13}x_{13} + c_{14}x_{21} + c_{15}x_{22} + c_{16}x_{23} = 0 \\ c_{20} + c_{21}x_{11} + c_{22}x_{12} + c_{23}x_{13} + c_{24}x_{21} + c_{25}x_{22} + c_{26}x_{23} = 0 \\ c_{30} + c_{31}x_{11} + c_{32}x_{12} + c_{33}x_{13} + c_{34}x_{21} + c_{35}x_{22} + c_{36}x_{23} = 0 \\ c_{40} + c_{41}x_{11} + c_{42}x_{12} + c_{43}x_{13} + c_{44}x_{21} + c_{45}x_{22} + c_{46}x_{23} = 0. \end{cases} \tag{2}$$

[1] Stating that $\mathbf{x}$ should not be close to infinity does not make sense as every point is at the same distance of infinity, although intuitively it sounds equivalent.

Except for an algebraic set in the coefficient space c_{ij} for L, the system above has four solutions, we have four generic points for all adjacent minors of a general 2-by-3 matrix, corresponding to the degree $\deg(f^{-1}(\mathbf{0})) = 4$.

To save work, reducing the number of variables from 6 to 2, we choose a different representation for the linear space defined by the equations $L(\mathbf{x}) = \mathbf{0}$, representing the 2-plane $L^{-1}(\mathbf{0})$ in $\mathbb{C}^6$ as

$$\begin{bmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} + \xi_1 \begin{bmatrix} v_{11} \\ v_{12} \\ v_{13} \\ v_{14} \\ v_{15} \\ v_{16} \end{bmatrix} + \xi_2 \begin{bmatrix} v_{21} \\ v_{22} \\ v_{23} \\ v_{24} \\ v_{25} \\ v_{26} \end{bmatrix} \tag{3}$$

spanned by an offset point $\mathbf{b} \in \mathbb{C}^6$ and an orthonormal basis $\{\mathbf{v}_1, \mathbf{v}_2\}$. The tuple (ξ_1, ξ_2) defines *intrinsic coordinates* for the generic points, introduced in [42] to speedup the algorithms of [41]. We point out that intrinsic coordinates are basis dependent.

The reduction from six to two variables reduces the cost of solving linear systems by a factor of twenty seven. This reduction improves the efficiency of Newton's method when computing sample points on the algebraic set, one of the basic operations in numerical algebraic geometry [46]; in particular to connect points on the same irreducible component via monodromy [38].

Definition 2.2 For any $f^{-1}(\mathbf{0}) \in \mathbb{C}^n$ with $\dim(f^{-1}(\mathbf{0})) = n - k$, let L be a k-plane given as

1. $L(\mathbf{x}) = 0$: a system of $n - k$ general linear equations in $\mathbf{x}$; or
2. $(\mathbf{b} \in \mathbb{C}^n, V \in \mathbb{C}^{n \times k})$: $\mathbf{b}$ is an offset point and $V = [\mathbf{v}_1\ \mathbf{v}_2\ \cdots\ \mathbf{v}_k] \in \mathbb{C}^{n \times k}$, $V^H V = I_k$ is an orthonormal[2] basis of vectors.

Solving $\{f(\mathbf{x}) = \mathbf{0}, L(\mathbf{x}) = \mathbf{0}\}$ gives generic points in *extrinsic coordinates*. Using $(\mathbf{b}, V)$ for L gives *intrinsic coordinates* $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots, \xi_k)$ for generic points $\mathbf{x}$:

$$\mathbf{x} = \mathbf{b} + \xi_1 \mathbf{v}_1 + \xi_2 \mathbf{v}_2 + \cdots + \xi_k \mathbf{v}_k = \mathbf{b} + V\boldsymbol{\xi}. \tag{4}$$

With intrinsic coordinates for generic points, the original variables $\mathbf{x}$ become place holders when solving $f(\mathbf{x} = \mathbf{b} + V\boldsymbol{\xi}) = \mathbf{0}$. In Figure 1, we outline the two ways to compute generic points.

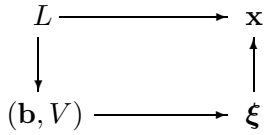

Figure 1: A commutative diagram for extrinsic $\mathbf{x}$ and intrinsic $\boldsymbol{\xi}$ coordinates of generic points. The vertical arrows require linear algebra while polynomial system solving is done horizontally.

[2]Although it suffices to require that the columns of the matrix V are linearly independent, the orthonormality condition $V^H V = I_k$ (using complex conjugated inner products and I_k is the k-by-k identity matrix) is beneficial.

We next define local intrinsic coordinate representations of generic points using the extrinsic coordinates of the generic point as the offset point for a k-plane. In the next two sections we will show that in some bad cases, the condition numbers of local intrinsic coordinates may significantly improve the intrinsic coordinate representation.

Definition 2.3 For any $f^{-1}(\mathbf{0}) \in \mathbb{C}^n$, $\dim(f^{-1}(\mathbf{0})) = n-k$ and $d = \deg(f^{-1}(\mathbf{0}))$ and $V \in \mathbb{C}^{n-k}$, $V^H V = I_k$ defined by a general k-plane L, consider d generic points $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_d\} = f^{-1}(\mathbf{0}) \cap L$. Because all $\mathbf{z}_\ell$ belong to the same witness set, $L(\mathbf{z}_\ell) = \mathbf{0}$ and

$$\mathbf{x} = \mathbf{z}_\ell + V\boldsymbol{\xi}, \quad \ell = 1, 2, \ldots, d. \tag{5}$$

are intrinsic coordinates with generic offset points. The *local intrinsic coordinates* to represent $f^{-1}(\mathbf{0})$ are defined by the tuple $(\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_d\}, V)$.

Note that in cases where $f(\mathbf{x}) = \mathbf{0}$ is not pure dimensional, the set $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_d\}$ does not coincide with the solution set of $f(\mathbf{x}) = \mathbf{0} = L(\mathbf{x})$. For example, consider in 3-space the case of a curve and a surface. To represent the curve we use one general plane L to find the generic points on the curve, but of course the plane L also has a nonempty intersection with the surface.

Obviously, the transition from *global* intrinsic coordinates $\mathbf{x} = \mathbf{b} + V\boldsymbol{\xi}$ to *local* intrinsic coordinates is performed by a mere evaluation of $\mathbf{b} + V\boldsymbol{\xi}$. While the diagram in Figure 1 commutes for exact operations, using floating-point arithmetic forces us to take into account condition numbers for the horizontal arrows of Figure 1.

3 Evaluation and Root Finding

The focus in this section is on one polynomial in one variable, the case of sampling witness sets of hypersurfaces and rational normal curves. We first experimentally illustrate how large condition numbers arise from particular choices of offset points, using the computer algebra system Maple so the experiments with the condition numbers can be replicated directly. Then we show that global intrinsic coordinates may lead to badly scaled companion matrices and therefore large condition numbers of the roots.

3.1 Condition Number Estimates

To estimate the condition numbers we use `LinearAlgebra[EigenConditionNumbers]` of Maple 12, with default settings of the `balance` parameter, and `UseHardwareFloats` set to `true`, see [29, Chapter 4]. The documentation of Maple 12 points to [2], see [12, §4.3] for the theory. Definition 3.8 defines the condition number of an eigenvalue applied to the root finding problem.

We consider one sparse polynomial f in $n = 10$ variables, of increasing degrees d, starting with t terms. In addition, we add all the linear terms $c_i x_i$, $i = 1, 2, \ldots, n$, to avoid ending up with the origin as a multiple root. The coefficients are taken on the complex unit circle, via the formula $e^{r\sqrt{-1}}$, with $r \in [0, 2\pi]$. The particular Maple commands used to generate an f are

```
[> n := 10: d := 10: t := 5:
```

```
[> c := () -> exp(I*stats[random,uniform[0,2*Pi]](1)):
[> X := [seq(x[i],i=1..n)]:
[> f := X[1]^d + randpoly(X,coeffs=c,degree=d-1,terms=5) + sum(c()*x[i],i=1..n);
```

The first term of `f` ensures that we have a monic polynomial after substitution $f(\mathbf{v}\xi)$, for $v_1 = 1$. That f is monic is convenient for the connection with the companion matrix when we look at the condition numbers of the corresponding eigenvalue problem.

To investigate the influence of the offset point on the condition number, we consider intrinsic coordinates once with and once without an offset point, using

$$\mathbf{x} = \mathbf{b} + \mathbf{v}\xi \quad \text{and} \quad \mathbf{x} = \mathbf{v}\xi, \quad \mathbf{b}, \mathbf{v} \in \mathbb{C}^n, \tag{6}$$

where all coefficients in the vectors are also taken on the complex unit circle. With $f(\mathbf{v}\xi) = 0$ we obtain still a sparse polynomial with all coefficients on the complex unit circle, which is not the case with $f(\mathbf{b} + \mathbf{v}\xi) = 0$. The offset vector of $\mathbf{b} + \mathbf{v}\xi$ is responsible for the variation in the coefficients and the fluctuation of the condition numbers (see Definition 3.8) we observe in our numerical experiments, summarized in Table 1. As we see from Table 1, the conditioning for $f(\mathbf{b} + \mathbf{v}\xi) = 0$ steadily worsens, whereas for $f(\mathbf{v}\xi) = 0$, the range between smallest and largest does not widen much implying that all roots of $f(\mathbf{v}\xi) = 0$ are well conditioned.

degrees	$f(\mathbf{b} + \mathbf{v}\xi) = 0$		$f(\mathbf{v}\xi) = 0$		ratios of	
of f	largest	smallest	largest	smallest	largest to smallest	
10	5.91e-01	9.02e-02	8.81e-01	4.01e-01	6.55e+00	2.20e+00
20	2.77e-01	1.76e-03	8.91e-01	3.31e-01	1.57e+02	2.70e+00
30	2.75e-01	6.16e-05	9.49e-01	7.25e-02	4.47e+03	1.31e+01
40	4.53e-01	7.14e-06	9.69e-01	1.87e-01	6.34e+04	5.17e+00

Table 1: Estimates for the inverse condition numbers of the zeroes of $f(\mathbf{b}+\mathbf{v}\xi) = 0$ and $f(\mathbf{v}\xi) = 0$. For degrees $d = 10$, 20, 30, and 40, we list the largest and smallest inverse condition numbers.

In global intrinsic coordinates, the offset vector $\mathbf{b}$ is generated before we solve $f(\mathbf{b} + \mathbf{v}\xi) = 0$. To compare the conditioning of global intrinsic with local intrinsic coordinates, we first solve $f(\mathbf{b} + \mathbf{v}\xi) = 0$ and take one root, say $\xi = z$. Then let $\mathbf{b}_z = \mathbf{b} + \mathbf{v}z$ so $f(\mathbf{b}_z + \mathbf{v}\xi) = 0$ has one solution $\xi = 0$ corresponding to z. For increasing degrees of f, the results of our numerical experiments are summarized in Table 2. As we see in Table 2, for increasing degrees, the condition number corresponding to z of $f(\mathbf{b}_z + \mathbf{v}\xi) = 0$ (for which the corresponding $\xi = 0$) is always reported as `1.00e+00`, or more precisely as `0.9999999...` which rounds to 1 within the working precision. For $\xi = 0$, we are at a root of the original f whose roots are all well conditioned. Condition numbers for other roots of $f(\mathbf{b}_z + \mathbf{v}\xi)$ worsen. So the good conditioning is maintained by taking the offset point $\mathbf{b}_{z_i}$ based on each root z_i and consider $f(\mathbf{b}_{z_i} + \mathbf{v}\xi) = 0$ locally.

This experiment illustrates that, although sampling a hypersurface is reduced to solving univariate polynomial equations, for hypersurfaces defined by polynomials of high degrees we cannot use the same representation of a general line to define generic points. If we adapt the offset point and switch to local intrinsic coordinates, then the generic points are well conditioned.

degrees	$f(\mathbf{b}+\mathbf{v}\xi)=0$			$f(\mathbf{b}_z+\mathbf{v}\xi)=0$		
of f	largest	2nd largest	smallest	largest	2nd largest	smallest
10	5.93e-01	4.71e-01	6.18e-02	1.00e+00	2.78e-03	1.99e-06
20	4.02e-01	3.30e-01	6.72e-03	1.00e+00	9.88e-06	6.94e-11
30	2.46e-01	1.08e-01	8.10e-04	1.00e+00	4.04e-08	3.44e-11
40	5.63e-01	2.36e-01	1.40e-04	1.00e+00	1.52e-08	3.90e-11

Table 2: Estimates for the inverse condition numbers of the roots of $f(\mathbf{b}+\mathbf{v}\xi)$ and $f(\mathbf{b}_z+\mathbf{v}\xi)$. In the right part of the table we have `1.00` as the condition number for $\xi=0$, corresponding to the root in local intrinsic coordinates.

The next experiment that reduces to polynomial in one variable is a rational normal curve. We consider a class of rational normal curves defined by a polynomial system

$$\left\{ \begin{array}{c} x_2 - x_1^{2d} = 0,\ x_3 - x_1^{3d} = 0,\ x_4 - x_1^{4d} = 0,\ x_5 - x_1^{5d} = 0 \\ c_0 + c_1 x_1 + c_2 x_2 + c_3 x_3 + c_4 x_4 + c_5 x_5 = 0. \end{array} \right. \tag{7}$$

After substitution, the computation of witness points amounts to solving a polynomial in one variable x_1. The Maple commands used to generate the rational normal curves of degree 10 are

```
[> n := 5: d:= 2: c := () -> exp(I*stats[random,uniform[0,2*Pi]](1)):
[> X := seq(x[i],i=1..n):
[> f := c() + sum(c()*x[i],i=1..n):
[> SX := seq(x[i]=x[1]^(i*d),i=2..n): f := subs(SX,f);
```

To compare the conditioning of global intrinsic with local intrinsic coordinates, we take one root z of $f(\mathbf{b}+\mathbf{v}\xi)=0$. We localize $\mathbf{b}$ to $\mathbf{b}_z$ taking $b_1=z$ and all other b_i plugging b_1 into the equations of (7) so $\mathbf{b}_z$ is a point on the curve and on the plane. The results are in Table 3. Compared to the hypersurface example, we see the condition numbers increase faster with increasing degrees and the same conclusions hold.

degrees	$f(\mathbf{b}+\mathbf{v}\xi)=0$			$f(\mathbf{b}_z+\mathbf{v}\xi)=0$		
of f	largest	2nd largest	smallest	largest	2nd largest	smallest
10	3.97e-01	3.55e-01	1.04e-02	1.00e+00	4.38e-03	5.71e-07
20	2.13e-01	2.11e-01	1.16e-05	1.00e+00	3.51e-03	9.19e-14
30	1.60e-02	1.47e-02	2.82e-08	1.00e+00	5.89e-06	9.50e-15
40	3.25e-02	1.26e-02	1.04e-10	1.00e+00	1.91e-06	2.36e-15

Table 3: Estimates for the inverse condition numbers of the roots of $f(\mathbf{b}+\mathbf{v}\xi)$ and $f(\mathbf{b}_z+\mathbf{v}\xi)$.

3.2 The Numerical Condition of Polynomial Evaluation

The numerical conditioning for the polynomial evaluation problem discussed in [12] leads to the formula (8) encapsulated in the following definition.

Definition 3.1 The *relative condition number* to evaluate a polynomial p of degree d in one variable x with complex coefficients is

$$\text{cond}(p,x) = \frac{\sum_{i=0}^{d} |c_i x^i|}{|p(x)|} \quad \text{for} \quad p(x) = \sum_{i=0}^{d} c_i x^i \quad \text{with} \quad c_i \in \mathbb{C}. \tag{8}$$

Observe that the condition number worsens as x approaches a root of p, as we consider the conditioning relative to the value of p at x. Following [12, Definition 1.1]: evaluating p at a root is an ill-posed problem because the condition number is infinite. Obviously the condition number also grows as x increases. Therefore we will assume that we evaluate at numbers on or close to the complex unit circle.

In this section we compare the numerical condition of evaluating a polynomial p

1. at $x = b + v\xi$, with $b, v \in \mathbb{C}$ chosen at random so $|b| = 1$ and $|v| = 1$; and
2. at $x = z + vh$, with the same $v \in \mathbb{C}$ as above and a small nonzero h: $0 < |h| \ll 1$.

The magnitude of h is such that we can neglect terms of order $O(h^2)$. The equation $b+v\xi = z+vh$ defines the relation between ξ and z.

Starting at the simple case for p equal to one monomial x^d of positive degree, we formulate a technical lemma, comparing the relative condition numbers. The lemma gives an upper bound gauging how much worse the global intrinsic coordinate representation can be compared to working with local intrinsic coordinates.

Lemma 3.2 *For $d > 1$, $|b| = 1$, $|v| = 1$, $|z| = 1$, and $0 < |h| \ll 1$, the ratio*

$$\frac{\text{cond}(x^d, x = b + v\xi)}{\text{cond}(x^d, x = z + vh)} \leq \frac{3^d}{1 - O(h)} \tag{9}$$

compares the condition of evaluating x^d as a polynomial in ξ to x^d as a polynomial in h.

Proof. Applying (8) with $x = b + v\xi = z + vh$ cancels the common denominator $|x^d|$.

Evaluating the numerator of (8) with binomial expansion:

$$\begin{aligned}
(b + v\xi)^d &= \sum_{i=0}^{d} \binom{d}{i} b^i (v\xi)^{d-i} && (10)\\
&\leq \sum_{i=0}^{d} \binom{d}{i} |b^i (v\xi)^{d-i}| && (11)\\
&= \sum_{i=0}^{d} \binom{d}{i} |\xi|^{d-i}, \quad \text{as } |b| = 1, |v| = 1 && (12)\\
&= (1 + |\xi|)^d. && (13)
\end{aligned}$$

Because $|h| \ll 1$: we ignore $O(h^i)$ for $i > 1$, and as $|v| = 1$: $(z + vh)^d \approx z^d + O(h)$.

To derive the upper bound of (9), the assumptions $|b| = |v| = |z| = 1$ imply $0 \leq |\xi| \leq 2 + |h|$, so for the numerator of the bound in (9) we have $(1 + |\xi|)^d \leq 3^d$, as we can work the $|h|$ of the $2 + |h|$ upper bound for ξ in the denominator. A lower bound on $z^d + O(h)$ is derived via the reverse triangle inequality: $|z^d - (-O(h))| \geq |\,|z^d| - |-O(h)|\,| = 1 - O(h)$, since $0 < |h| \ll 1$. □

Note that the factor 3^d in the upper bound of (9) is sharp, for example: plugging $z = 1$, $b = -1$, $v = 1$, $h = \epsilon$ into $b + v\xi = z + vh$ leads to $\xi = 2 + \epsilon$ and then the numerator of the formula (8) evaluates to 3^d for infinitesimal ϵ, illustrating the effect of an exponential propagation of errors.

Even for the case of one monomial, the condition number in the worst case is amplified by a huge factor. For example, $3^{17} = 1.3 \times 10^8$: with standard double precision we may lose half of the significant digits.

In a first generalization of Lemma 3.2, we consider two monomials. In contrast to the case $p = x^d$, the number z could be a root of p, even as $|z| = 1$. Therefore, we assume $|p(z)| \gg |h|$.

Lemma 3.3 *Let $p = c_k x^k + c_\ell x^\ell$. For $|b| = 1$, $|v| = 1$, $|z| = 1$, $|p(z)| \gg |h|$, and $0 < |h| \ll 1$:*

$$\frac{\text{cond}(p, x = b + v\xi)}{\text{cond}(p, x = z + vh)} \leq \frac{|c_k|3^k + |c_\ell|3^\ell}{|p(z)| - O(h)}. \tag{14}$$

Proof. Applying the binomial expansion to the two terms in p, using $|b| = 1$ and $|v| = 1$:

$$|c_k(b + v\xi)^k| + |c_\ell(b + v\xi)^\ell| \leq |c_k|(1 + |\xi|)^k + |c_\ell|(1 + |\xi|)^\ell. \tag{15}$$

By $|b| = |v| = |z| = 1$ we have $0 \leq |\xi| \leq 2 + |h|$ and thus the numerator of the upper bound of (14), as we also here take the $|h|$ of the $2 + |h|$ into account when we consider the denominator. For the denominator, we compute

$$c_k(z + vh)^k + c_\ell(z + vh)^\ell \approx (c_k z^k + O(h)) + (c_\ell z^\ell + O(h)) = p(z) + O(h), \tag{16}$$

ignoring higher order terms because $|h| \ll 1$. Application of the reverse triangle inequality assuming $|p(z)| \gg |h|$ leads to $|p(z)| - O(h)$ as the lower bound for the denominator of (14). □

Proposition 3.4 *Let $p = \sum_{i=0}^{d} c_i x^i$. For $|b| = 1$, $|v| = 1$, $|z| = 1$, $|p(z)| \gg |h|$, and $0 < |h| \ll 1$:*

$$\frac{\text{cond}(p, x = b + v\xi)}{\text{cond}(p, x = z + vh)} \leq \frac{\sum_{i=0}^{d} |c_i|3^i}{|p(z)| - O(h)}. \tag{17}$$

Proof. Similar as in Lemma 3.3, we apply triangle inequalities. □

Corollary 3.5 *In addition to all assumptions of Proposition 3.4, let* $|c_i| = 1$, *for* $i = 0, 1, \ldots, d$, *the ratio*

$$\frac{\text{cond}(p, x = b + v\xi)}{\text{cond}(p, x = z + vh)} \leq \frac{1}{2} \frac{3^{d+1} - 1}{|p(z)| - O(h)} \tag{18}$$

compares the condition of evaluating p *as a polynomial in* ξ *to* p *as a polynomial in* h.

Proof. $\sum_{i=0}^{d} 3^i = \frac{1}{2}(3^{d+1} - 1)$. □

The generalization of formula (8) to polynomials in several variables is immediate.

Definition 3.6 The *relative condition number* to evaluate a sparse polynomial f in n variables, with support set $A \in \mathbb{N}^n$, $\#A < \infty$, is

$$\text{cond}(f, \mathbf{x}) = \frac{\sum_{\mathbf{a} \in A} |c_{\mathbf{a}} \mathbf{x}^{\mathbf{a}}|}{|f(\mathbf{x})|}, \quad \text{for} \quad f(\mathbf{x}) = \sum_{\mathbf{a} \in A} c_{\mathbf{a}} \mathbf{x}^{\mathbf{a}}, \quad c_{\mathbf{a}} \in \mathbb{C} \setminus \{0\}, \ \mathbf{x}^{\mathbf{a}} = x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}. \tag{19}$$

Observe that only nonzero coefficients are stored, because unlike polynomials in one variable, the number of monomials that may occur grows exponentially in the degree and the number of variables. With the notation of (19), the degree of f is computed as

$$\deg(f) := \max_{\mathbf{a} \in A} (a_1 + a_2 + \cdots + a_n). \tag{20}$$

We are interested in the evaluations of f resulting from intersecting f with a general line. In particular, we compare the numerical conditioning of evaluating f

1. at $\mathbf{x} = \mathbf{b} + \mathbf{v}\xi$, with $\mathbf{b}, \mathbf{v} \in \mathbb{C}^n$ chosen at random so $|b_i| = 1$ and $|v_i| = 1$, $i = 1, 2, \ldots, n$; and
2. at $\mathbf{x} = \mathbf{z} + \mathbf{v}h$, with the same $\mathbf{v} \in \mathbb{C}^n$, and a small nonzero h: $0 < |h| \ll 1$.

Eliminating $\mathbf{x}$ from $\mathbf{x} = \mathbf{b} + \mathbf{v}\xi$ and $\mathbf{x} = \mathbf{z} + \mathbf{v}h$ defines the relation between ξ and $\mathbf{z}$. For fixed $\mathbf{b}$, $\mathbf{v}$, and $\mathbf{z}$, given sufficiently small ξ (or h) we solve for h (or respectively ξ) in the least squares sense. In local intrinsic coordinates, when $\xi = \mathbf{0}$, the offset vector $\mathbf{b}$ equals a generic point $\mathbf{z}$.

Theorem 3.7 *Let* $f = \sum_{\mathbf{a} \in A} c_{\mathbf{a}} \mathbf{x}^{\mathbf{a}}$. *For* $|b_i| = 1$, $|v_i| = 1$, $|z_i| = 1$, $i = 1, 2, \ldots, n$, $|f(\mathbf{z})| \gg |h|$, *and* $0 < |h| \ll 1$*:*

$$\frac{\text{cond}(f, \mathbf{x} = \mathbf{b} + \mathbf{v}\xi)}{\text{cond}(f, \mathbf{x} = \mathbf{z} + \mathbf{v}h)} \leq \frac{\sum_{\mathbf{a} \in A} |c_{\mathbf{a}}| 3^{a_1 + a_2 + \cdots + a_n}}{|f(\mathbf{z})| - O(h)}. \tag{21}$$

Proof. The denominator of (21) is obtained via binomial expansion: $f(\mathbf{x} = \mathbf{z} + \mathbf{v}h) = f(\mathbf{z}) + O(h)$, ignoring higher order terms as $|h| \ll 1$. The reverse triangle inequality and the assumption $|f(\mathbf{z})| \gg |h|$ leads to the lower bound $|f(\mathbf{z} + \mathbf{v}h)| \geq |f(\mathbf{z})| - O(h)$, for the denominator of (21).

The expression of the numerator is derived via application of Lemma 3.3. In particular, because all components of the tuples of variables have the same magnitude, estimating the magnitude of $c_{\mathbf{a}}x^{\mathbf{a}}$ is the same as estimating the value of $c_{\mathbf{a}}x^{a_1+a_2+\cdots+a_n}$. □

Summarizing Theorem 3.7: the relative condition number of evaluating a degree d polynomial at $\mathbf{x} = \mathbf{b} + \mathbf{v}\xi$ is in bad cases, approximately 3^d times larger than the relative condition number of evaluating at $\mathbf{x} = \mathbf{z} + \mathbf{v}h$, for $0 < |h| \ll 1$. The assumption that $|f(\mathbf{z})| \gg |h|$ means that $|h|$ is dominated by $|f(\mathbf{z})|$.

3.3 The Numerical Condition of Polynomial Roots

In this section we demonstrate that the shift $x = b + v\xi$ may lead to a significant deterioration of the condition number of the roots of a polynomial equation. Following [47], we define the condition number of roots of a polynomial in one variable via the condition numbers of the eigenvalues of the companion matrix.

Definition 3.8 Let C_p be the companion matrix of a polynomial p in one variable x and with complex coefficients. Solutions to $p(x) = 0$ are eigenvalues denoted by z with corresponding right eigenvectors $\mathbf{r} \in \mathbb{C}^n$: $C_p\mathbf{r} = z\mathbf{r}$ and left eigenvectors $\mathbf{q} \in \mathbb{C}^n$: $\mathbf{q}^H C_p = \mathbf{q}^H z$. The *condition number* $\kappa(p, z)$ *of a zero* z *of* p with corresponding left and right eigenvectors $\mathbf{q}_z$ and $\mathbf{r}_z$ is

$$\kappa(p, z) = \frac{||\mathbf{q}_z||_2||\mathbf{r}_z||_2}{|\mathbf{q}_z^H \mathbf{r}_z|}. \tag{22}$$

We start by considering polynomials with perfectly conditioned roots.

Lemma 3.9 *Consider* $p = x^d - 1$. *For all* z, $p(z) = 0$ *such that* $\kappa(p, z) = 1$.

Proof. The right eigenvectors $\mathbf{r}_z$ of the companion matrix of $x^d - 1$ for any zero z are defined by

$$\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ z \\ z^2 \\ \vdots \\ z^{d-2} \\ z^{d-1} \end{bmatrix} = z \begin{bmatrix} 1 \\ z \\ z^2 \\ \vdots \\ z^{d-2} \\ z^{d-1} \end{bmatrix}. \tag{23}$$

For the left eigenvectors $\mathbf{q}_z$ we consider the right eigenvectors of C_p^H with eigenvalue $\overline{z}$, as $(\mathbf{q}^H C_p)^H = (\mathbf{q}^H z)^H$ is equivalent to $C_p^H \mathbf{q} = \overline{z}\mathbf{q}$:

$$\begin{bmatrix} 0 & 0 & \cdots & 0 & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ z \\ z^2 \\ \vdots \\ z^{d-2} \\ z^{d-1} \end{bmatrix} = \overline{z} \begin{bmatrix} 1 \\ z \\ z^2 \\ \vdots \\ z^{d-2} \\ z^{d-1} \end{bmatrix}, \quad \overline{z} = z^{d-1}. \tag{24}$$

So corresponding to z, we have as left eigenvector $\mathbf{q}^H = [1 \;\; \overline{z} \;\; \overline{z^2} \;\; \cdots \;\; \overline{z^{d-2}} \;\; \overline{z^{d-1}}]$.

Because $z^k = \exp(ik\theta)$, $\overline{z^k} = \exp(-ik\theta)$ and $\overline{z}z = 1$, we compute

$$||\mathbf{r}_z||_2^2 = \mathbf{r}_z^H \mathbf{r}_z = 1 + \overline{z}z + \overline{z^2}z^2 + \cdots + \overline{z^{d-2}}z^{d-2} + \overline{z^{d-1}}z^{d-1} = d. \tag{25}$$

Similarly: $||\mathbf{q}_z||_2^2 = d$ and we have $||\mathbf{q}_z||_2||\mathbf{r}_z||_2 = d$. Then $\mathbf{q}_z^H \mathbf{r}_z = 1 + \overline{z}z + \overline{z^2}z^2 + \cdots + \overline{z^{d-2}}z^{d-2} + \overline{z^{d-1}}z^{d-1} = d$. Thus, $\kappa(p, z) = 1$. □

Our approach to numerical conditioning is algebraic as we study how sensitive the roots are to changes in the coefficients. Geometrically, we relate the numerical condition to the inverse of the distance between the roots. For $x^d - 1 = 0$, the roots get closer to each other asymptotically as d grows to infinity. In contrast, the algebraic condition number remains invariant. In [51, page 19], the roots of $x^d - 1$ are deemed extremely well conditioned as a perturbation of ϵ on any coefficient gives a perturbation of ϵ/d in all the roots. Derived formally in [15], roots of $x^d - 1$ have condition number $1/d$, improving with increasing degree. Note that formula (22) is unaware of the sparse structure of the polynomial.

The following lemma considers the conditioning of the roots if we shift the coefficients of the polynomial just a little bit. Because we start from a perfectly conditioned problem, the condition numbers remain very good.

Lemma 3.10 *Let $v \in \mathbb{C}$, $|v| = 1$, and h, $0 < |h| \ll 1$ consider $p = (x + vh)^d - 1$. For all z, $p(z) = 0$, we have $\kappa(p, z) = 1 + O(h)$.*

Proof. We express the companion matrix of p as

$$C_p(h) = C_p + C_1 h + O(h^2), \tag{26}$$

where C_p is the companion matrix of $x^d - 1$. We could redo all calculations in the proof of Lemma 3.9, ignoring higher order terms of h, as $|h| \ll 1$ or simply apply [47, Corollary 4.3.1]: $z(h) = z + O(h)$ for all eigenvalues $z(h)$ of $C_p(h)$ corresponding to the eigenvalues z of C_p. Ignoring higher order terms of h when evaluating (22) leads to $\kappa(p, z) = 1 + O(h)$, for any zero z of p. □

Next we turn our attention to $p(x) = (b + vx)^d - 1 = 0$ for constants b and v on the complex unit circle. Again we emphasize that our notion of numerical conditioning is algebraic, not geometric. In the geometric point of view, the roots of p compared to those of $x^d - 1$ are merely translated. As this translation preserves the distance between the roots one would geometrically not expect a worsening of the condition number. We could confirm this algebraically by plugging in the shifted roots into formula (22). However, even for $|b| = 1$ and $|v| = 1$ the companion matrix of p undergoes a drastic perturbation from the companion matrix of $x^d - 1$.

Lemma 3.11 *Let $b, v \in \mathbb{C}$, $|b| = 1$, $|v| = 1$, and consider $p = (b + xv)^d - 1$. For all z, $p(z) = 0$, we have $\kappa(p, z) \leq d\sqrt{\frac{4^d \Gamma(d+1/2)}{\sqrt{\pi}\Gamma(d+1)}}$.*

Proof. We apply the theorem of Bauer-Fike [12, Theorem 4.5]. Let C_p be the companion matrix of p, denote by C_d the companion matrix of $x^d - 1$, and consider $E = C_p - C_d$. By the Bauer-Fike Theorem all eigenvalues of C_p lie in disks centered at the eigenvalues λ of C_d and with radius $d||E||_2 \kappa(x^d - 1, \lambda)$.

By Lemma 3.9: $\kappa(x^d - 1, \lambda) = 1$ for all $\lambda^d - 1 = 0$. So we only need to find a bound for $||E||_2$. Because C_d and C_p are companion matrices, the only nonzero elements of E are on its last row. Computing $||E||_2$ via the spectral radius of $E^H E$ leads to the observation that $B := E^H E$ is a zero matrix, except for one element on row d and column d. The square root of that one nonzero element $B_{d,d}$ is the spectral radius of B (as all other eigenvalues of B are zero) and thus equals $||E||_2$.

In bounding $B_{d,d}$, we look at the coefficient vector $\mathbf{c}$ of p:

$$p(x) = \sum_{i=0}^{d} c_i x^i = \sum_{i=0}^{d} \left(\begin{array}{c} d \\ i \end{array} \right) b^{d-i} v^i x^i. \tag{27}$$

In taking the difference $C_p - C_d$, one was subtracted from the constant term of p, but this is only a minor subtraction and we compute $\mathbf{c}^H \mathbf{c}$ to get an upper bound of $B_{d,d}$. As $|b| = 1$ and $|v| = 1$ we focus on the binomial coefficients in $\mathbf{c}$ and find (with the help of Maple 12):

$$\mathbf{c}^H \mathbf{c} = \sum_{i=0}^{d} \left(\begin{array}{c} d \\ i \end{array} \right)^2 \overline{b^{d-i}} b^{d-i} \overline{v^i} v^i = \sum_{i=0}^{d} \left(\begin{array}{c} d \\ i \end{array} \right)^2 = \frac{4^d \Gamma(d + 1/2)}{\sqrt{\pi} \Gamma(d+1)}. \tag{28}$$

So $\sqrt{\frac{4^d \Gamma(d+1/2)}{\sqrt{\pi}\Gamma(d+1)}}$ bounds the spectral radius of B.

We still have to justify why we may use the number $d\sqrt{\frac{4^d \Gamma(d+1/2)}{\sqrt{\pi}\Gamma(d+1)}}$ as a bound on $\kappa(p, z)$, because the number bounds the radii of the disks centered around the original roots of $x^d - 1$ and not around the roots of p. To that end we apply [12, Theorem 4.7] (proven in [11]) to the eigenvalue problem defined by C_p. Adjusting the notation of [12, Theorem 4.7]: We have that $\kappa(p, z) \leq \text{cond}(S)$, where S is the matrix which contains in its columns the eigenvectors of C_p. Note that the eigenvectors of C_p are powers of the eigenvalues. To derive a bound on cond(S), we use the 2-norm in the formula for the condition number, $\text{cond}(S) = ||S||_2 \cdot ||S^{-1}||_2$, because then we can use the bound we derived on the spectral radius. In the worst case, one eigenvalue of C_p may lie on the boundary of the disk centered at root of unity, while another eigenvalue of C_p remained on the unit circle, thus skewing the size of the coefficients in the matrix S: in one column entries grow exponentially, while in another column the entries stay on the unit circle. In that case $||C_p||_2$ attains the number $d\sqrt{\frac{4^d \Gamma(d+1/2)}{\sqrt{\pi}\Gamma(d+1)}}$ and the bound on $\kappa(p, z)$ follows. □

Before the lemma we cautioned against the geometric interpretation of the condition numbers of the roots of the polynomial in intrinsic coordinates. At the end of the proof above we see that the companion matrix is very close to a singular matrix B for which we obtained an explicit bound on its spectral radius. So the companion matrix of the polynomial in intrinsic coordinates is possibly near a singular matrix.

We summarize the results of the lemmas in the following theorem.

Theorem 3.12 *Let $b, v \in \mathbb{C}$, $|b| = 1$, $|v| = 1$, and h such that $0 < |h| \ll 1$. Then, the ratio*

$$\frac{\kappa\left((b + vx)^d - 1, z\right)}{\kappa\left((x + vh)^d - 1, z\right)} \leq 2^{d-1} - O(h) \tag{29}$$

compares the conditioning of the solutions of $(b+vx)^d - 1 = 0$ with the solutions of $(x+vh)^d - 1 = 0$.

Proof. Follows from Lemma 3.10 and Lemma 3.11. Because $\log_2\left(\sqrt{\frac{4^d\Gamma(d+1/2)}{\sqrt{\pi}\Gamma(d+1)}}\right)$ increases fairly linearly and is bounded by $d-1$, we replace $\sqrt{\frac{4^d\Gamma(d+1/2)}{\sqrt{\pi}\Gamma(d+1)}}$ by 2^{d-1} as a more convenient upper bound. We use $1/(1-x) = 1+x+O(x^2)$ with $x = -O(h)$ to obtain $-O(h)$ in the upper bound. □

We end by pointing out that the upper bound of the theorem is attained for the case of $(-1+x)^d - 1 = 0$ where 2 is a solution and powers of 2 appear in the companion matrix. Although $x^d - 1 = 0$ is a very special polynomial, note that random polynomials are known to have their zeroes close to the complex unit circle [35].

4 Improved Numerical Conditioning

The previous section dealt with the special cases of sampling a hypersurface or rational normal curve. In this section we address the general case. Borrowing again from numerical linear algebra we first define the condition number of an isolated solution of a polynomial system. For embedding and randomization methods to represent generic points on components that do not occur as complete intersections we refer respectively to [36] and [46, §13.5].

Definition 4.1 Let $f(\mathbf{x}) = \mathbf{0}$ be a polynomial system of n equations in n unknowns. Denote the Jacobian matrix of f by J_f and let $\mathbf{z} \in \mathbb{C}^n$ be an isolated solution of $f(\mathbf{x}) = \mathbf{0}$. Then, *the relative condition number of the zero* $\mathbf{z}$ *as a solution of* $f(\mathbf{x}) = \mathbf{0}$ is

$$\kappa(f, \mathbf{z}) = ||J_f(\mathbf{z})||_2 ||J_f^{-1}(\mathbf{z})||_2, \tag{30}$$

i.e.: $\kappa(f, \mathbf{z})$ is the condition number of the Jacobian matrix of the polynomials in the system evaluated at $\mathbf{z}$.

The justification for this definition comes from [34, Theorem 3.2], showing that asymptotically near a zero, the numerical conditioning of the nonlinear system is the same as its derivative. Recall that the linear system $J_f(\mathbf{x})\Delta\mathbf{x} = -f(\mathbf{x})$ determines the update $\Delta\mathbf{x}$ in Newton's method. Although condition numbers for linear system solving are often defined for other norms [16] [21], we have $||C||_2 = \sqrt{\rho(C^H C)}$ where $\rho(\cdot)$ returns the spectral radius of a matrix.

We encountered eigenvalues earlier in Definition 3.8 for polynomials in one variable. We remark that the condition number of the companion matrix for a polynomial in one variable reflects much better our numerical problems caused by the offset vector in global intrinsic coordinates than the derivative does. Therefore, Definition 4.1 should be applied only for proper systems, i.e.: only for $n > 1$.

The condition number of the companion matrix reflects much better our numerical problems caused by the offset vector in global intrinsic coordinates than the derivative does. A small derivative f' signals the closeness to a multiple solution but is not necessarily small for our problems.

The notion of $\kappa(f, \mathbf{z})$ is local: for one solution $\mathbf{z}$ — although other solutions near to $\mathbf{z}$ will increase $\kappa(f, \mathbf{z})$ — and in particular: for the given values of the coefficients of f. Different coordinate systems will give different values for $\kappa(f, \mathbf{z})$. The local and particular properties of $\kappa(f, \mathbf{z})$ explain why κ gives a *relative* condition number.

Definition 4.2 Let $\mathbf{z} \in \mathbb{C}^n$ be a generic point on an $(n-k)$-dimensional component of $f^{-1}(\mathbf{0})$, satisfying k linear equations $L(\mathbf{z}) = \mathbf{0}$. Then, *the relative extrinsic condition number of* $\mathbf{z}$*, as a generic point on* $f^{-1}(\mathbf{0}) \cap L$ is

$$\kappa_{\mathcal{E}}(f, L, \mathbf{z}) = \kappa(f = (f, L), \mathbf{z}). \tag{31}$$

Writing the solutions to the linear equations $L(\mathbf{x}) = \mathbf{0}$ as $\mathbf{x} = \mathbf{b} + V\boldsymbol{\xi}$, for some offset point $\mathbf{b}$ and orthonormal matrix $V \in \mathbb{C}^{n \times k}$, we have $\mathbf{z} = \mathbf{b} + V\boldsymbol{\xi}_z$, where $\boldsymbol{\xi}_z$ are the intrinsic coordinates of $\mathbf{z}$. Then, *the relative intrinsic condition number of* $\mathbf{z}$*, as a generic point of* $f^{-1}(\mathbf{0})$ is

$$\kappa_{\mathcal{I}}(f, \mathbf{b}, V, \mathbf{z}) = \kappa(f = f(\mathbf{b} + V\boldsymbol{\xi}_{\mathbf{z}}), \boldsymbol{\xi}_{\mathbf{z}}). \tag{32}$$

Finally, *the relative local intrinsic condition number of* $\mathbf{z}$ *as a generic point on* $f^{-1}(\mathbf{0})$ is

$$\kappa_{\mathcal{L}}(f, V, \mathbf{z}) = \kappa(f = f(\mathbf{z} + V\boldsymbol{\xi}), \boldsymbol{\xi} = \mathbf{0}). \tag{33}$$

Like in Definition 4.1, we emphasize the *relative* aspect of our condition numbers as local and particular for the given values of coefficients. By given, we refer to the coefficients of the polynomials in f and not the coefficients of L, $\mathbf{b}$ and V because for L, $\mathbf{b}$, and V we generate well-conditioned representations.

Analogous to Lemma 3.9, we start considering generic points with optimal numerical condition numbers and consider as test equation $\mathbf{x}^{\mathbf{a}} - 1 = 0$. Similar as the univariate $x^d - 1$, this test equation corresponds to the best conditioned case. Intersecting the solution set of $x^{\mathbf{a}} - 1 = 0$ with a random line is equivalent to considering a polynomial in one variable, for which the roots are expected to be close to the complex unit circle.

Lemma 4.3 *Let* $f = \mathbf{x}^{\mathbf{a}} - 1 = \mathbf{0}$, $\mathbf{x}^{\mathbf{a}} = x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}$, *denote* $d = a_1 + a_2 + \cdots + a_n$. *One can find a generic point* $\mathbf{z}$ *and coefficients for hyperplanes* L, *such that* $f(\mathbf{z}) = 0$, $L(\mathbf{z}) = \mathbf{0}$, *and* $\kappa_{\mathcal{E}}(f, L, \mathbf{z}) \leq \sqrt{n} d^2$.

Proof. We consider the system

$$F(\mathbf{x}) = \begin{cases} f(\mathbf{x}) = 0 \\ L(\mathbf{x}) = \mathbf{0} \end{cases} \tag{34}$$

and its Jacobian matrix

$$J_F(\mathbf{x}) = \begin{bmatrix} a_1 x_1^{a_1 - 1} x_2^{a_2} \cdots x_n^{a_n} & a_2 x_1^{a_1} x_2^{a_2 - 1} \cdots x_n^{a_n} & \cdots & a_n x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n - 1} \\ c_{1,1} & c_{1,2} & \cdots & c_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n-1,1} & c_{n-1,2} & \cdots & c_{n-1,n} \end{bmatrix} \tag{35}$$

at the zero $\mathbf{z}$. The symbols $c_{i,j}$ are coefficients of L.

Let us consider first the case where $f = x_1^d - 1 = 0$. Let D by the n-by-n identity matrix with its element $D_{1,1}$ on the first row and first column equal to d. Then we write $J_F(\mathbf{z}) = DC$. The first row of the matrix C consists of z_1^{d-1}, followed by zeroes. The other rows of the matrix C contain the coefficients of the linear equations of L. Because z_1 satisfies $x_1^d - 1 = 0$, z_1^{d-1} also lies on the complex unit circle and thus we may assume that C is chosen as an orthonormal matrix,

with $||C||_2||C^{-1}||_2 = 1$. The property $||DC||_2 \leq ||D||_2||C||_2$, combined with $||D||_2 = d$ yields $||J_F(\mathbf{z})||_2 \leq d$. For $||J_F^{-1}(\mathbf{z})||_2$ we use $J_F^{-1} = C^{-1}D^{-1}$ and $||C^{-1}D^{-1}||_2 \leq ||C^{-1}||_2||D^{-1}||_2 = 1$, as the largest eigenvalue of D^{-1} is still 1. So we find $\kappa_{\mathcal{E}}(f, L, \mathbf{z}) = ||J_F(\mathbf{z})||_2||J_F^{-1}(\mathbf{z})||_2 \leq d$.

For the general case, note that we may assume that all $a_i \neq 0$, otherwise, for $a_i = 0$ we discard the corresponding variable x_i. While in (35), we may choose the coefficients $c_{i,j}$ of the linear equations so that the last $n-1$ rows are mutually orthogonal to each other, the growth of the numbers on the first row of $J_F(\mathbf{x})$ could stem from large values of a_i. Therefore we rewrite $J_F(\mathbf{x})$ as a product of three matrices:

$$J_F(\mathbf{x}) = ABC, \tag{36}$$

with

$$A = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1/(a_1\sqrt{n}) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/(a_{n-1}\sqrt{n}) \end{bmatrix}, \quad C = \begin{bmatrix} a_1\sqrt{n} & 0 & \cdots & 0 \\ 0 & a_2\sqrt{n} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_n\sqrt{n} \end{bmatrix}, \tag{37}$$

and

$$B = \begin{bmatrix} x_1^{a_1-1}x_2^{a_2}\cdots x_n^{a_n}/\sqrt{n} & x_1^{a_1}x_2^{a_2-1}\cdots x_n^{a_n}/\sqrt{n} & \cdots & x_1^{a_1}x_2^{a_2}\cdots x_n^{a_n-1}/\sqrt{n} \\ c_{1,1} & c_{1,2}a_1/a_2 & \cdots & c_{1,n}a_1/a_n \\ \vdots & \vdots & \ddots & \vdots \\ c_{n-1,1}a_{n-1}/a_1 & c_{n-1,2}a_{n-1}/a_2 & \cdots & c_{n-1,n}a_{n-1}/a_n \end{bmatrix}. \tag{38}$$

We have $||J_F(\mathbf{x})||_2 \leq ||A||_2||B||_2||C||_2$, where $||A||_2 = 1$ (because $a_i \geq 1$) and $||C||_2 = \max_{i=1}^{n} a_i\sqrt{n} \leq d\sqrt{n}$. For $J_F^{-1} = C^{-1}B^{-1}A^{-1}$, we have likewise $||J_F^{-1}(\mathbf{x})||_2 \leq ||C^{-1}||_2||B^{-1}||_2||A^{-1}||_2$ where $||C^{-1}||_2 \leq 1/\sqrt{n}$ and $||A^{-1}||_2 = \max_{i=1}^{n-1} a_i\sqrt{n} \leq d\sqrt{n}$. So the bound follows if we can choose the coefficients of B so that $||B||_2 = 1$ and $||B^{-1}||_2 = 1$.

While in the special case of one variable, the upper bound on the condition number for J_F was d, in general it could happen that a_1 dominates all other degrees and we end up with a_1^2 which is larger than d, but still bounded by d^2.

The rewriting of J_F has freed the first row of B from the degrees. We can simplify the first row: $x_1^{a_1}\cdots x_k^{a_k}\cdots x_n^{a_n} = 1$ implies $x_1^{a_1}\cdots x_k^{a_k-1}\cdots x_n^{a_n} = 1/x_k$, for $k = 1, 2, \ldots, n$. So the matrix B becomes

$$B = \begin{bmatrix} x_1^{-1}/\sqrt{n} & x_2^{-1}/\sqrt{n} & \cdots & x_n^{-1}/\sqrt{n} \\ c_{1,1} & c_{1,2}a_1/a_2 & \cdots & c_{1,n}a_1/a_n \\ \vdots & \vdots & \ddots & \vdots \\ c_{n-1,1}a_{n-1}/a_1 & c_{n-1,2}a_{n-1}/a_2 & \cdots & c_{n-1,n}a_{n-1}/a_n \end{bmatrix} \tag{39}$$

The solution we take satisfies $\mathbf{x}^{\mathbf{a}} - 1 = 0$. Because we take derivatives to form the Jacobian matrix of $F(\mathbf{x}) = \mathbf{0}$, the constant coefficients $c_{i,0}$ of the linear equations in $L(\mathbf{x}) = \mathbf{0}$ do not show up in the Jacobian matrix. We may thus choose the coefficients $c_{i,0}$ so that the solutions we take from $\mathbf{x}^{\mathbf{a}} - 1 = 0$ have their coordinates all on the complex unit circle. As the 2-norm of any n-vector with all its coordinates on the complex unit circle equals $\sqrt{n}$, we have that the 2-norm

of the first row of B equals one. The other rows of B depend then on the other coefficients $c_{i,j}$ and there is enough freedom to ensure that B is an orthogonal matrix. The bound then follows because B is orthogonal. □

In the statement of Lemma 4.3 we do not consider $\mathbf{z}$ as *given* (and thus fixed), because even as $\mathbf{z}$ has to satisfy $\mathbf{x}^{\mathbf{a}} - 1 = 0$, it is possible to choose extremal values for the components of $\mathbf{z}$ which would then lead to a badly scaled Jacobian matrix with relative high condition number. Lemma 3.11 is extended to address the condition of intrinsic coordinates of our test equation.

Lemma 4.4 *Let $\mathbf{z} \in \mathbb{C}^n$ be a generic point of $f(\mathbf{x}) = \mathbf{x}^{\mathbf{a}} - 1 = 0$, $\mathbf{x}^{\mathbf{a}} = x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}$, $d = a_1 + a_2 + \cdots + a_n$, with $\mathbf{z} = \mathbf{b} + \mathbf{v}\xi_{\mathbf{z}}$ for some offset point $\mathbf{b}$ and a vector $\mathbf{v}$. Then $\kappa_{\mathcal{I}}(f, \mathbf{b}, \mathbf{v}, \xi_{\mathbf{z}}) \leq 2^{d-1}$.*

Proof. Note that in case $a_1 = d$ and all other $a_i = 0$, $i = 2, \ldots, n$, Lemma 3.11 applies immediately. Repeated substitution of z_k by $b_k + \mathbf{v}\xi_{\mathbf{z}}$ leads to a case where Lemma 3.11 applies. Without loss of generality we may assume that all b_k are equal to the same b and all v_k equal to the same v. Then the equation $\mathbf{x}^{\mathbf{a}} - 1$ turns into $(b + v\xi)^d - 1$, the equation of Lemma 3.11. As in the proof of Theorem 3.12, we use the more convenient 2^{d-1} for the expression $\sqrt{\frac{4^d \Gamma(d+1/2)}{\sqrt{\pi}\Gamma(d+1)}}$. □

As we remarked after Lemma 3.11, we have cases where the bound 2^{d-1} is attained, but otherwise, this bound may be too pessimistic as the substitution of $b_k + v_k\xi$ into $\mathbf{x}^{\mathbf{a}} - 1$ in general leads to a polynomial in ξ with coefficients of modest size.

Lemma 4.5 *Consider $\mathbf{x} = \mathbf{z} + \mathbf{v}\xi$ for some vector $\mathbf{v}$, $||\mathbf{v}||_2 = 1$, and $\mathbf{z} \in \mathbb{C}^n$ a generic point for $f(\mathbf{x}) = \mathbf{x}^{\mathbf{a}} - 1$. Then, $\kappa_{\mathcal{L}}(f, \mathbf{v}, \mathbf{z}) = 1$.*

Proof. Follows from Lemma 3.10. □

Theorem 4.6 *For a generic point $\mathbf{z}$ for the equation $f(\mathbf{x}) = \mathbf{x}^{\mathbf{a}} - 1 = 0$, with $d = \deg(f)$, we have:*

$$\kappa_{\mathcal{L}}(f, \mathbf{v}, \mathbf{z}) \leq \kappa_{\mathcal{E}}(f, L, \mathbf{z}) \leq \kappa_{\mathcal{I}}(f, \mathbf{b}, \mathbf{v}, \mathbf{z}) \leq 2^{d-1}, \tag{40}$$

where $\mathbf{z}$ lies on some generic line with offset $\mathbf{b}$, direction $\mathbf{v}$, and linear equations $L(\mathbf{x}) = 0$.

Proof. The statement of the theorem is the summary of Lemma 4.3, Lemma 4.4, and Lemma 4.5. □

5 A Recentering Algorithm

In this section we consider the sampling of algebraic sets using local intrinsic coordinates. We define a recentering algorithm and address its numerical stability. In addition, using local intrinsic coordinates leads to a better stepsize control.

The essence of the recentering algorithm is that it keeps the intrinsic coordinates small during path following, so the coordinates of the generic points stay local. In the previous section we demonstrated the improved numerical conditioning of local intrinsic coordinates, so if we can

keep the coordinates local we avoid the accumulation of rounding errors and obtain a numerically stable algorithm.

When sampling points, the moving L from $(\mathbf{b}, V)$ to $(\mathbf{c}, W)$, is done via the obvious homotopy:

$$f\left(\begin{array}{ccccc} \mathbf{x} = & (1-t)\mathbf{b} + t\mathbf{c} & + & ((1-t)V + tW) & \boldsymbol{\xi} \\ & \text{moving offset point} & & \text{moving basis vectors} & \end{array} \right) = \mathbf{0}. \tag{41}$$

As t moves from 0 to 1, the solution paths $\boldsymbol{\xi}(t)$ are tracked with predictor-corrector methods and give new generic points on $f^{-1}(\mathbf{0})$. For introductions to path following and continuation methods we refer to [1] and [30], see also [28].

Similar to our assumption that we started at a linear space L that gave well-conditioned solutions to represent generic points, we assume that the new linear space we are moving to will give well-conditioned solutions. This means in particular that the homotopy (41) does not end in a singular point of $f^{-1}(\mathbf{0})$. Regarding the choice of W, we choose an orthonormal basis. Given that (41) moves to well-conditioned solutions, picking a bad choice for W has probability zero.

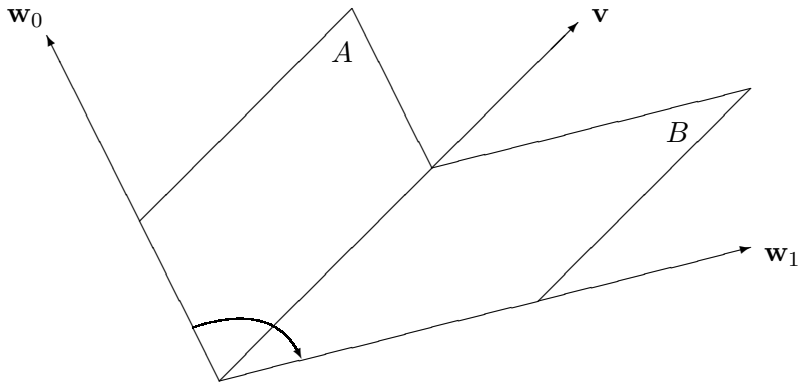

Figure 2: Two planes A and B in 3-space in general position meet in a common line with direction $\mathbf{v}$. The directions of A are spanned by $\mathbf{v}$ and $\mathbf{w}_0$, while B has in its basis the directions $\mathbf{v}$ and $\mathbf{w}_1$. Moving A to B is done by $(1-t)\mathbf{w}_0 + t\mathbf{w}_1$, for t going from 0 to 1.

We illustrate in Figure 2 an improvement to the obvious homotopy (41). For sampling a space curve in 3-space we consider two planes. Any two planes in relative general position with respect to each other intersect in a line. If the direction of the line of intersection is one of the two directions of the basis to represent both planes, then there is only one basis vector that moves during the sampling of the space curve.

Sampling a space curve in extrinsic coordinates in $\mathbb{C}^n$ happens by moving one hyperplane, from $L_A(\mathbf{x}) = \mathbf{0}$ to $L_B(\mathbf{x}) = \mathbf{0}$ typically via a homotopy $(1-t)L_A(\mathbf{x}) + tL_B(\mathbf{x}) = 0$ where the parameter t changes the plane A (defined by $L_A(\mathbf{x}) = \mathbf{0}$) into the plane B (defined by $L_B(\mathbf{x}) = \mathbf{0}$). As $\dim(A \cap B) = n-2$, there are $n-2$ vectors $\mathbf{v}_1$, $\mathbf{v}_2$, ..., $\mathbf{v}_{n-2}$ common to A and B when we consider the parametric representations of the hyperplanes A and B. For $\mathbf{w}_0 \in A \setminus (A \cap B)$ and $\mathbf{w}_1 \in B \setminus (A \cap B)$, the homotopy in intrinsic coordinates is then defined as $(1-t)\mathbf{w}_0 + t\mathbf{w}_1$, for t going from 0 to 1. Note that, as we work with orthogonal bases of directions, for any t, the vector

$(1-t)\mathbf{w}_0 + t\mathbf{w}_1$ is perpendicular to all $\mathbf{v}_i$, for $i = 1, 2, \ldots, n-2$. The main point is that just like in extrinsic coordinates, sampling a space curve is governed by the movement of one parameter t. The sampling of a space curve in extrinsic coordinates changes the $n+1$ coefficients of one linear equation and in intrinsic coordinates one direction vector of length n changes.

Given a witness set for a space curve and a target hyperplane for the new samples to belong to, computing a representation of the basis vectors so they share all except one direction vector with the start hyperplane may be called a *reorientation* of the linear slicing plane. Sampling any algebraic set of any dimension happens by tracing a space curve on that algebraic set. If we have some choice in the next k-plane we are moving to, then it is beneficial to choose the target k-plane to have $k-1$ directions in common with the start k-plane so only one direction vector moves.

Now we consider the stage when we have fixed one instance $(\mathbf{b}, W)$ of a moving k-plane. In local intrinsic coordinates when generic points $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_d\}$ are offset points, we could consider moving from $(\mathbf{z}_\ell, W)$ to $(\mathbf{b}, W)$ via a deformation defined by

$$f\left(\mathbf{x} = (1-t)\mathbf{z}_\ell + s\mathbf{b} + W\boldsymbol{\xi}\right) = \mathbf{0}, \quad \text{for } s \text{ from 0 to 1.} \tag{42}$$

In contrast to the obvious homotopy in (41), we see that only the offset point moves. Instead of using (42) and moving to $\mathbf{b}$, we point out that any point in the k-plane L can serve as an offset point. Therefore, we should choose the best offset point, i.e.: the point closest to the current generic point. To compute the closest point, let $\mathbf{c}$ be the orthogonal projection of $\mathbf{z}_\ell$ onto the k-plane L. For some step size h, we then consider:

$$f\left(\mathbf{x} = \mathbf{z}_\ell + h(\mathbf{c} - \mathbf{z}_\ell) + W\boldsymbol{\xi}\right) = \mathbf{0} \tag{43}$$

and apply Newton's method to find the correction $\Delta\boldsymbol{\xi}$, as illustrated in Figure 3.

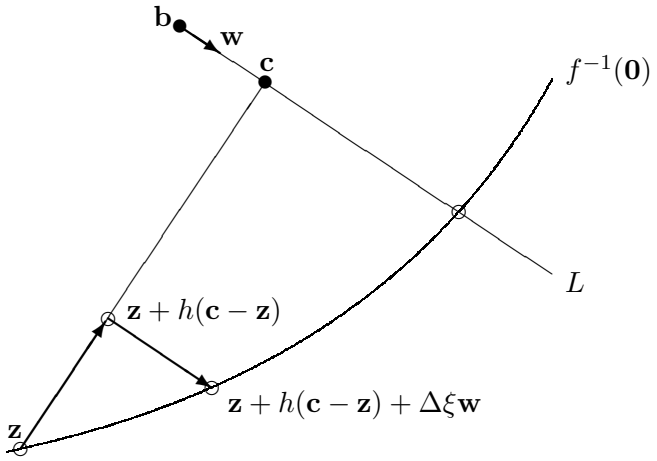

Figure 3: Schematic of one predictor-corrector step of the new sampling algorithm, moving from the point $\mathbf{z}$ to the point where L meets $f^{-1}(\mathbf{0})$. The line L is defined by $\mathbf{b} + \xi\mathbf{w}$. Using step size h, the prediction $h(\mathbf{c} - \mathbf{z})$ added to $\mathbf{z}$ occurs in a direction orthogonal to L while the correction $\Delta\xi\mathbf{w}$ is parallel to L.

After each step, we add the correction term $W\Delta\boldsymbol{\xi}$ to the offset point, recentering the intrinsic coordinates to local intrinsic coordinates at the end of the correction stage. Pseudocode for one predictor-corrector step is given in Algorithm 5.1, going from one generic point $\mathbf{z} \in f^{-1}(\mathbf{0}) \cap K$, where K is the current k-plane towards L, the target k-plane.

Algorithm 5.1 (one predictor-corrector step in local intrinsic coordinates)

Input:	$f = (f_1, f_2, \ldots, f_k), f_i(\mathbf{x}) \in \mathbb{C}[\mathbf{x}]$, $i = 1, 2, \ldots, k$;	$\dim(f^{-1}(\mathbf{0})) = n - k$
	$\mathbf{b} \in \mathbb{C}^n$;	*offset point of k-plane L*
	$W = [\mathbf{w}_1\ \mathbf{w}_2\ \cdots\ \mathbf{w}_k] \in \mathbb{C}^{n \times k}$, $W^*W = I_k$;	*orthonormal basis for L*
	$\mathbf{z} \in \mathbb{C}^n$: $f(\mathbf{z}) = \mathbf{0}$, $K(\mathbf{z}) = \mathbf{0}$;	*generic point on k-plane K*
	$h > 0$;	*step size*
	$\epsilon > 0$.	*accuracy requirement*
Output:	$\widehat{\mathbf{z}}$, $f(\widehat{\mathbf{z}}) = \mathbf{0}$, $L(\widehat{\mathbf{z}}) = \mathbf{0}$: $\|\|\widehat{\mathbf{z}} - \mathbf{b}\|\| < \|\|\mathbf{z} - \mathbf{b}\|\|$.	*generic point closer to L*
1.	$\mathbf{v} := \mathbf{z} - \mathbf{b}$;	*go towards offset point*
2.	$\mathbf{v} := \mathbf{v} - \sum_{i=1}^{k} (\mathbf{w}_i{}^H \mathbf{v}) \mathbf{w}_i$;	*move perpendicular to L*
3.	$\mathbf{v} := \frac{\mathbf{v}}{\|\|\mathbf{v}\|\|}$;	*normalize so $\|\|\mathbf{v}\|\| = 1$*
4.	$\widetilde{\mathbf{z}} := \mathbf{z} + h\ \mathbf{v}$;	*prediction for new generic point*
5.	$\widehat{\mathbf{z}} := \widetilde{\mathbf{z}}$; $\boldsymbol{\xi} := \mathbf{0}$;	*initialize for Newton corrector*
6.	while $\|\|f(\widehat{\mathbf{z}} + W\boldsymbol{\xi})\|\| > \epsilon$ do	*as long as not accurate enough*
6.1	$\Delta\boldsymbol{\xi} := f(\widehat{\mathbf{z}} + W\boldsymbol{\xi}) / f'(\widehat{\mathbf{z}} + W\boldsymbol{\xi})$;	*solve a linear system for $\Delta\boldsymbol{\xi}$*
6.2	$\boldsymbol{\xi} := \boldsymbol{\xi} + \Delta\boldsymbol{\xi}$;	*update correction*
7.	$\widehat{\mathbf{z}} := \widehat{\mathbf{z}} + W\boldsymbol{\xi}$.	*recenter to local coordinates*

The orthonormality condition $W^H W = I_k$ is important for instruction 2 in the Algorithm 5.1 because we can compute the projection just via inner products. The number of arithmetical operations needed to carry out instruction 2 in Algorithm 5.1 is $O(kn)$. Without the condition $W^H W = I_k$, this cost (e.g. via Gram-Schmidt orthogonalization) would be at least $O(kn^2)$.

For the numerical stability of Algorithm 5.1, we first discuss the relationship between the step size h and the accuracy requirement ϵ. If on the one hand h is too small, then the condition $||f(\widehat{\mathbf{z}} + W\boldsymbol{\xi})|| > \epsilon$ in the while-do instruction 6 of Algorithm 5.1 is directly satisfied. If on the other hand h is too large, satisfying the accuracy requirement of instruction 6 may require too many iterations, or Newton's method may not converge at all. We point out that the cost of instruction 6.1 is $O(k^3)$ and if f is sufficiently sparse (if evaluation and differentiation go fast), then the cost of execution of Newton's method dominates the cost of Algorithm 5.1.

In general path tracking algorithms, the step size h is determined via a feedback mechanism. If Newton's method does not converge fast enough, then the step size is reduced. If Newton's method needs only two steps or less, then the step size might be enlarged. See [7] for stepsize control strategies. The problem with this feedback mechanism is that it comes at the great expense of the most costly portion of the predictor-corrector method, i.e.: each reduction of h comes at the expense of a failed and thus wasted Newton step. With local intrinsic coordinates, we can predict the fitness of the step size with a simple evaluation. For some step size h and

direction $\mathbf{v}$, we evaluate and estimate the residual as

$$||f(\mathbf{x} = \mathbf{z}_\ell + h\mathbf{v})|| \text{ is } ||f(\mathbf{z}_\ell) + O(h)|| \text{ is } O(h). \tag{44}$$

For example, if $h = 10^{-2}$ and we see that the residual is $O(10^{-2})$, then it is fair to expect that after one iteration of Newton's method, the residual becomes $O(10^{-4})$, and then $O(10^{-8})$ after the second iteration.

In Algorithm 5.2 we define how to cut back on the step size just by evaluation, *before* the start of the Newton correction.

Algorithm 5.2 (a priori stepsize control by evaluation)

Input: $f = (f_1, f_2, \ldots, f_k), f_i(\mathbf{x}) \in \mathbb{C}[\mathbf{x}]$ $i = 1, 2, \ldots, k$;	$\dim(f^{-1}(\mathbf{0})) = n - k$
$\mathbf{z} \in \mathbb{C}^n$: $f(\mathbf{z}) = \mathbf{0}$, $K(\mathbf{z}) = \mathbf{0}$;	*generic point on k-plane K*
$\mathbf{v} \in \mathbb{C}^n$, $\|\|\mathbf{v}\|\| = 1$;	*direction vector*
$h > 0$;	*current step size*
$\delta > 0$.	*threshold to reduce h*
$1 > \rho > 0$.	*reduction factor for h*
Output: $h > 0$.	*updated step size*
1. $y := \|\|f(\mathbf{z} + h\mathbf{v})\|\|$;	*evaluate the predicted point*
2. if $y/h > \delta$ then $h := \rho h$.	*reduce the step size*

The reduction of the step size in instruction 2 of Algorithm 5.2 could be followed by another evaluation of f to see if y is reduced sufficiently or has become even too small.

Algorithm 5.2 is called after instruction 3 of Algorithm 5.1.

By application of Algorithm 5.2, occurrences of a diverging Newton's method can be greatly reduced because the size of the residual $||f(\mathbf{x} = \mathbf{z} + W\boldsymbol{\xi})||$ is $O(h)$. Another indicator for success or failure of the corrector is the distance of the current solution to the new position of the slicing plane.

We conclude with a quick cost estimate for the total number of Newton steps along one path. In sampling for generic points, we typically choose the new random coefficients for the k-plane as complex numbers on the unit circle, so the distance between two k-planes (and in particular their offset points) is $O(1)$. For h: $0 < h < 1$, we can see that the total number of Newton iterations along a solution path is proportional to $1/h$. For example if $h = 0.01$ and we need about 2 or 3 Newton iterations per step, then the total number of Newton iterations along a solution path will vary between 200 and 300.

The homotopy continuation methods of this paper are different from the so-called linear homotopies for which an experimental study to certify path tracking recently appeared in [8]. A potential future research direction could be to expand the quick cost estimate of the previous paragraph into a formal complexity study, along the lines of [9] and [32].

6 Computational Results

Local intrinsic coordinates are available since version 2.3.53 of PHCpack [48, 49]. In version 2.3.66, sampling was made more explicitly available via a new option `phc -y`. The option `-y` of `phc` takes on input a witness set and returns a new witness set.

We first describe numerical experiments done with Maple to compare condition numbers of generic points on a hypersurface of polynomials of increasing degrees. Then we report preliminary results on small benchmark problems with the sampling routines in PHCpack. All computations were done on one core of a Mac OS X 3.2 Ghz Intel Xeon.

The polynomial systems we selected occur in the literature. We briefly summarize the main characteristics of these systems:

1. All adjacent minors of a general 2-by-n matrix, $n = 3, 4, \ldots$. This is a family of nice quadratic equations arising in algebraic statistics [13].

2. The cyclic n-roots systems are well known academic benchmarks. If n has a quadratic divisor, then the system has a positive dimensional solution set [3]. In our experiments we use the cyclic 8-roots system, which has a one dimensional solution set of degree 144.

3. Griffis-Duffy platforms [17] are architecturally singular mechanisms [23], their motion correspond to curves of degree 40 in 8-space [40].

For the purposes of this paper, the computation of the first set of generic points is considered as given, typically in extrinsic coordinate representation.

Except for the adjacent minors, the systems are not complete intersections. For $m > k$, to make an m-by-k system f square, we generate a random k-by-m matrix C and work with $C \times f$.

To test the improvement from using local intrinsic coordinates, we sample new generic points from the solution sets. Our computational experimental setup consists of three stages:
(1) Given one set of generic points, we generate another random k-plane L.
(2) We then move the given set of generic points to lie on L.
(3) At the end we check results for accuracy, count #predictor-corrector steps, record elapsed cpu times.

Note that the recorded cpu times are only meant to give an indication on the relative practical difficulties of these problems. More relevant are the number of iterations performed by Newton's method along the paths.

In Table 4 we summarize the results. Even as the systems we selected as benchmark examples are not challenging, we observe a clear benefit of using local intrinsic coordinates, even for the systems defined by quadratic equations. The benefit is perhaps most significant for the cyclic 8-roots problem where the degree of the ith polynomial equals i.

The modest size of the benchmark systems (and of the improvements in the timings) are for comparison purposes with the original global intrinsic coordinates. On larger systems with many more solution paths, the numerical conditioning with global intrinsic coordinates becomes so bad that the tracker in global intrinsic coordinates fails. Because of the unpredictable rate at which such failures occur, a systematic comparison between global and local intrinsic coordinates requires a much larger benchmark collection which is not in the scope of the current paper.

polynomial system	n	$n-k$	d	#iterations	timings
Griffis-Duffy platform	8	1	40	207/164	550/535 μsec
cyclic 8-roots system	8	1	144	319/174	5.3/3.2 sec
all adjacent minors of 2-by-11 matrix	22	12	1,024	285/219	44.6/40.3 sec

Table 4: Preliminary experiments on three systems. For each system we respectively list the ambient dimension n, the dimension $n-k$ of the solution set, and the degree d of the set. We list the average number of Newton iterations along a path for intrinsic and local intrinsic coordinates, as well as user cpu timings.

7 Conclusions

We list at least three advantages of using local intrinsic coordinates for sampling: (1) only the offset point moves; (2) the sparse structure of the polynomials is kept; and (3) we can control the step size by evaluation. Applications to numerical algebraic geometry include (1) implicitization via interpolation; (2) monodromy breakup algorithm [38]; and (3) diagonal homotopies [42]. In particular, local intrinsic coordinates will add to the robustness of our parallel subsystem-by-subsystem solver [19].

References

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

[2] E. Anderson, Z. Bai, C. Bischof, J. Blackford, S. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK's users guide*, volume 9 of *Software, Environments, and Tools*. SIAM, 3rd edition, 1999.

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

[4] D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler. Bertini: Software for numerical algebraic geometry. Available at `http://www.nd.edu/~sommese/bertini/`.

[5] D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler. Adaptive multiprecision path tracking. *SIAM J. Numer. Anal.*, 46(2):722–746, 2008.

[6] D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler. Software for numerical algebraic geometry: a paradigm and progress towards its implementation. 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 1–14. Springer-Verlag, 2008.

[7] D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler. Stepsize control for path tracking. In D.J. Bates, G. Besana, S. Di Rocco, and C.W. Wampler, editors, *Interactions of Classical and Numerical Algebraic Geometry*, volume 496 of *Contemporary Mathematics*, pages 21–31. AMS, 2009.

[8] C. Beltran and A. Leykin. Certified numerical homotopy tracking. Preprint `arXiv:0912.0920v1 [math.NA]`.

[9] L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and Real Computation*. Springer-Verlag, 1998.

[10] B.H. Dayton and Z. Zeng. Computing the multiplicity structure in solving polynomial systems. In M. Kauers, editor, *Proceedings of the 2005 International Symposium on Symbolic and Algebraic Computation (ISSAC'05), July 24-27 2005, Beijing, China.*, pages 116–123. ACM, 2005.

[11] J. Demmel. The condition number of equivalence transformations that block diagonalize matrix pencils. *SIAM J. Numer. Anal.*, 20(3):599–610, 1983.

[12] J.W. Demmel. *Applied Numerical Linear Algebra*. SIAM, 1997.

[13] P. Diaconis, D. Eisenbud, and B. Sturmfels. Lattice walks and primary decomposition. In B.E. Sagan and R.P. Stanley, editors, *Mathematical Essays in Honor of Gian-Carlo Rota*, volume 161 of *Progress in Mathematics*, pages 173–193. Birkhäuser, 1998.

[14] A Galligo and A. Poteaux. Continuations and monodromy on random Riemann surfaces. In H. Kai and H. Sekigawa, editors, *SNC'09. Proceedings of the 2009 International Workshop on Symbolic-Numeric Computation*, pages 115–123. ACM, 2009.

[15] W. Gautschi. Questions of numerical condition related to polynomials. In G.H. Golub, editor, *Studies in Numerical Analysis*, volume 24 of *MAA Studies in Mathematics*, pages 140–177. MAA, 1984.

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

[17] M. Griffis and J. Duffy. Method and apparatus for controlling geometrically simple parallel mechanisms with distinctive connections. US Patent 5,179,525, 1993.

[18] E. Gross, S. Petrović, and J. Verschelde. PHCpack in Macaulay2. Preprint `arXiv:1105.4881v1 [math.AG]`.

[19] Y. Guan and J. Verschelde. Parallel implementation of a subsystem-by-subsystem solver. In *The proceedings of the 22th High Performance Computing Symposium, Quebec City, 9-11 June 2008*, pages 117–123. IEEE Computer Society, 2008.

[20] Y. Guan and J. Verschelde. PHClab: A MATLAB/Octave interface to PHCpack. 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 15–32. Springer-Verlag, 2008.

[21] N.J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, 1996.

[22] S. Hosten and J. Shapiro. Primary decomposition of lattice basis ideals. *Journal of Symbolic Computation*, 29(4 and 5):625–639, 2000.

[23] M.L. Husty and A. Karger. Self-motions of Griffis-Duffy type parallel manipulators. In *Proc. 2000 IEEE Int. Conf. Robotics and Automation*, 2000. San Francisco, CA, April 24–28, CDROM.

[24] S. Kim and M. Kojima. Numerical stability of path tracing in polyhedral homotopy continuation methods. *Computing*, 73(4):329–348, 2004.

[25] A. Leykin. Numerical algebraic geometry for Macaulay 2. `arXiv:0911.1783v1 [math.AG]`.

[26] A. Leykin and J. Verschelde. Interfacing with the numerical homotopy algorithms in PHCpack. In N. Takayama and A. Iglesias, editors, *Proceedings of ICMS 2006*, volume 4151 of *Lecture Notes in Computer Science*, pages 354–360. Springer-Verlag, 2006.

[27] A. Leykin, J. Verschelde, and A. Zhao. Newton's method with deflation for isolated singularities of polynomial systems. *Theoret. Comput. Sci.*, 359(1-3):111–122, 2006.

[28] T.Y. Li. Numerical solution of polynomial systems by homotopy continuation methods. In F. Cucker, editor, *Handbook of Numerical Analysis. Volume XI. Special Volume: Foundations of Computational Mathematics*, pages 209–304. North-Holland, 2003.

[29] M.B. Monagan, K.O. Geddes, K.M. Heal, G. Labahn, S.M. Vorkoetter, J. McCarron, and P. DeMarco. *Maple Advanced Programming Guide*. Maplesoft, 2008.

[30] A. Morgan. *Solving polynomial systems using continuation for engineering and scientific problems*. Prentice-Hall, 1987. Volume 57 of Classics in Applied Mathematics Series, SIAM 2009. Pages with code at `http://www.siam.org/books/cl57`.

[31] D. Mumford. *Algebraic Geometry I. Complex Projective Varieties*. Classics in Mathematics. Springer-Verlag, 1995. Corrected Second Printing of the 1976 Edition. Originally published as volume 221 of the Grundlehren der mathematischen Wissenschaften.

[32] M. Petković. *Point Estimation of Root Finding Methods*, volume 1933 of *Lecture Notes in Mathematics*. Springer-Verlag, 2007.

[33] A. Poteaux. Computing monodromy groups defined by plane curves. In J. Verschelde and S.M. Watt, editors, *SNC'07. Proceedings of the 2007 International Workshop on Symbolic-Numeric Computation*, pages 239–246. ACM, 2007.

[34] W.C. Rheinboldt. On measures of ill-conditioning for nonlinear equations. *Mathematics of Computation*, 30(133):104–111, 1976.

[35] L.A. Shepp and R.J. Vanderbei. The complex zeros of random polynomials. *Transactions of the American Mathematical Society*, 347(11):4365–4384, 1995.

[36] A.J. Sommese and J. Verschelde. Numerical homotopies to compute generic points on positive dimensional algebraic sets. *J. of Complexity*, 16(3):572–602, 2000.

[37] A.J. Sommese, J. Verschelde, and C.W. Wampler. Numerical decomposition of the solution sets of polynomial systems into irreducible components. *SIAM J. Numer. Anal.*, 38(6):2022–2046, 2001.

[38] A.J. Sommese, J. Verschelde, and C.W. Wampler. Using monodromy to decompose solution sets of polynomial systems into irreducible components. In C. Ciliberto, F. Hirzebruch, R. Miranda, and M. Teicher, editors, *Application of Algebraic Geometry to Coding Theory, Physics and Computation*, pages 297–315. Kluwer Academic Publishers, 2001. Proceedings of a NATO Conference, February 25 - March 1, 2001, Eilat, Israel.

[39] A.J. Sommese, J. Verschelde, and C.W. Wampler. Numerical irreducible decomposition using PHCpack. In M. Joswig and N. Takayama, editors, *Algebra, Geometry, and Software Systems*, pages 109–130. Springer-Verlag, 2003.

[40] A.J. Sommese, J. Verschelde, and C.W. Wampler. Advances in polynomial continuation for solving problems in kinematics. *ASME Journal of Mechanical Design*, 126(2):262–268, 2004.

[41] A.J. Sommese, J. Verschelde, and C.W. Wampler. Homotopies for intersecting solution components of polynomial systems. *SIAM J. Numer. Anal.*, 42(4):552–1571, 2004.

[42] A.J. Sommese, J. Verschelde, and C.W. Wampler. An intrinsic homotopy for intersecting algebraic varieties. *J. Complexity*, 21(4):593–608, 2005. Festschrift for the 70th Birthday of Arnold Schönhage, edited by T. Lickteig and L.M. Pardo.

[43] A.J. Sommese, J. Verschelde, and C.W. Wampler. Introduction to numerical algebraic geometry. In A. Dickenstein and E.Z. Emiris, editors, *Solving Polynomial Equations. Foundations, Algorithms and Applications*, volume 14 of *Algorithms and Computation in Mathematics*, pages 301–337. Springer-Verlag, 2005.

[44] A.J. Sommese, J. Verschelde, and C.W. Wampler. Solving polynomial systems equation by equation. In A. Dickenstein, F.-O. Schreyer, and A.J. Sommese, editors, *Algorithms in Algebraic Geometry*, volume 146 of *The IMA Volumes in Mathematics and Its Applications*, pages 133–152. Springer-Verlag, 2008.

[45] A.J. Sommese and C.W. Wampler. Numerical algebraic geometry. In J. Renegar, M. Shub, and S. Smale, editors, *The Mathematics of Numerical Analysis*, volume 32 of *Lectures in Applied Mathematics*, pages 749–763. AMS, 1996. Proceedings of the AMS-SIAM Summer Seminar in Applied Mathematics. Park City, Utah, July 17-August 11, 1995, Park City, Utah.

[46] A.J. Sommese and C.W. Wampler. *The Numerical solution of systems of polynomials arising in engineering and science*. World Scientific, 2005.

[47] E.E. Tyrtyshnikov. *A Brief Introduction to Numerical Analysis*. Birkhäuser, 1997.

[48] J. Verschelde. Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation. *ACM Trans. Math. Softw.*, 25(2):251–276, 1999. Software available at `http://www.math.uic.edu/~jan/download.html`.

[49] J. Verschelde. Polynomial homotopy continuation with PHCpack. *ACM Communications in Computer Algebra*, 44(4):217–220, 2010.

[50] J. Verschelde and G. Yoffe. Polynomial homotopies on multicore workstations. In M.M. Maza and J.-L. Roch, editors, *PASCO 2010. Proceedings of the 2010 International Workshop on Parallel Symbolic Computation*, pages 131–140. ACM, 2010.

[51] J.H. Wilkinson. The perfidious polynomial. In G.H. Golub, editor, *Studies in Numerical Analysis*, volume 24 of *MAA Studies in Mathematics*, pages 1–28. MAA, 1984.