SIAM J. NUMER. ANAL.
Vol. 40, No. 6, pp. 2026–2046
© 2002 Society for Industrial and Applied Mathematics

SYMMETRIC FUNCTIONS APPLIED TO DECOMPOSING SOLUTION SETS OF POLYNOMIAL SYSTEMS*

ANDREW J. SOMMESE†, JAN VERSCHELDE‡, AND CHARLES W. WAMPLER§

Abstract. Many polynomial systems have solution sets comprised of multiple irreducible components, possibly of different dimensions. A fundamental problem of numerical algebraic geometry is to decompose such a solution set, using floating-point numerical processes, into its components. Prior work has shown how to generate sets of generic points guaranteed to include points from every component. Furthermore, we have shown how monodromy can be used to efficiently predict the partition of these points by membership in the components. However, confirmation of this prediction required an expensive procedure of sampling each component to find an interpolating polynomial that vanishes on it. This paper proves theoretically and demonstrates in practice that linear traces suffice for this verification step, which gives great improvement in both computational speed and numerical stability. Moreover, in the case that one may still wish to compute an interpolating polynomial, we show how to do so more efficiently by building a structured grid of samples, using divided differences, and applying symmetric functions. Several test problems illustrate the effectiveness of the new methods.

Key words. components of solutions, divided differences, embedding, generic points, traces, homotopy continuation, irreducible components, Newton identities, Newton interpolation, numerical algebraic geometry, monodromy, polynomial system, symmetric functions

AMS subject classifications. Primary, 65H10; Secondary, 13P05, 14Q99, 68W30

PII. S0036142901397101

1. Introduction.

Polynomial systems arising in scientific and engineering applications often have positive dimensional solution sets; moreover, the solution set may have components of different dimensions. For instance, in mechanical engineering, we may be given a set of rigid parts and a prescription for how they are to be connected by joints. These specifications can be formulated as a system of polynomial equations whose solution set describes the locations in space of all the parts. It may happen that some assemblies of the mechanism are rigid, whereas other assemblies of the same parts and joints allow an internal motion having one or more degrees of freedom. The notion of "degrees of freedom of motion" as used by a kinematician is thus equivalent to the "dimension of a solution set" for the polynomial system. Problems with similar characteristics arise in other disciplines.

For such polynomial systems, our task is to identify all irreducible components of the solution set, characterizing each component by its dimension and degree and providing witness points on the set. This problem is central in a developing new field: *numerical algebraic geometry*, a research program initiated in [33]. The goal is to

*Received by the editors October 28, 2001; accepted for publication (in revised form) June 4, 2002; published electronically December 13, 2002. This research was supported by the Volkswagen-Stiftung (RiP-program at Oberwolfach).

http://www.siam.org/journals/sinum/40-6/39710.html

†Department of Mathematics, University of Notre Dame, Notre Dame, IN 46556-4618 (sommese@nd.edu, http://www.nd.edu/~sommese). This material is based upon work supported by the National Science Foundation under grant 0105653 and the Duncan Chair of the University of Notre Dame.

‡Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, 851 South Morgan (M/C 249), Chicago, IL 60607-7045 (jan@math.uic.edu, jan.verschelde@na-net.ornl.gov, http://www.math.uic.edu/~jan). This material is based upon work supported by the National Science Foundation under grant 0105739 and grant 0134611.

§General Motors Research and Development, Mail Code 480-106-359, 30500 Mound Road, Warren, MI 48090-9055 (Charles.W.Wampler@gm.com).

design numerically stable algorithms to efficiently solve polynomial systems arising in science and engineering. In the next two paragraphs we explain the relation of the current paper to previous work.

In [26], we presented a new method of embedding a polynomial system into a larger polynomial system, such that the numerical solution of a sequence of homotopies computes generic points on all solution components of the original polynomial system. These witness points form the basic data to decompose the solution sets into irreducible components, as we showed in [27]. Starting at any solution point, one may use numerical continuation to sample the component that contains it and construct an interpolation polynomial vanishing on the component. This polynomial can then be used as a filter to find all other witness points in the same component. In this way, all of the points can be sorted into components, eventually producing a list of all the components and certain properties of them, such as degree and dimension. However, the construction of the interpolating polynomial is both expensive and numerically difficult for high degree components in many variables. In [28], we reduced the number of variables by detecting the linear span of a component and reduced the degree of the interpolant by using central projections; but even so, numerically challenging cases remain.

An alternative approach to determining which witness points lie on the same component is to use monodromy to find paths connecting them [29]. In computational experiments, this approach has been found to be numerically stable on high degree components and highly successful in predicting the correct decomposition. However, it is heuristic in that connections are discovered via randomly generated monodromy loops, with no a priori way to know when all connections have been found. Thus, the prediction must still be validated by other means. In [29], this was accomplished by computing an interpolating polynomial, as before, so the problem of high degree polynomials was not eliminated.

The most significant contribution of this paper is to prove theoretically and demonstrate computationally that *linear* traces are sufficient for validating a proposed decomposition. Due to the superior numerical stability of linear systems, we are able to run our decomposition method entirely with standard machine arithmetic. For polynomial systems with coefficients given as double floating-point numbers, whose evaluation map is numerically well conditioned, and whose irreducible components have multiplicity one, our algorithm does not need multiprecision arithmetic to decompose the solution sets, even in the occurrence of high degree components. For components with multiplicity higher than one, multiprecision arithmetic is required to track the singular paths [31].

In the case that one still wishes to compute polynomials that vanish on a component, the higher order traces can be used to good effect. First, the witness points on a component can be marched forward together to provide a structured grid of sample points. Then, with a "bootstrapping" technique, we can construct the Newton form of the interpolating polynomial. The use of traces enables the direct application of Newton interpolation, eliminating the need for extra bootstrapping samples. A final improvement in efficiency is gained by using Newton identities to reduce the number of samples to the number of monomials in the interpolant, which is the minimum possible. However, this last shortcut is inadvisable for high degree components as our tests show that it is numerically less stable than using a full grid of samples.

Several test problems illustrate the effectiveness of the methods. Particularly notable are the results on a problem from mechanical engineering: a special Stewart–

Gough platform mechanism that has internal motion. For one case in which the motion is one irreducible component of degree 28, the computing time for validating the decomposition predicted by monodromy is reduced from 1.3 hours using our former methods to less than 5 seconds with the linear trace. This is now comparable to the time required for a related example in which the degree 28 component breaks up into several low degree irreducibles. Hence the running time of the algorithm is no longer sensitive to such changes in the geometry of the solution set. Moreover, it is interesting to note that the automated numerical method discovered a solution component that was missed by experts using a manual approach aided by computer symbolic processing.

In brief, the paper proceeds as follows. In the next section, we collect some results on traces, which are then applied to monodromy in section three. In sections four and five, we outline the interpolation algorithms and apply them in the last section on the cyclic 8-roots and 9-roots problems and on the mechanism problem just mentioned.

2. Traces of functions. The results in this section are quite old, e.g., Theorem 2.1 is for the most part just a statement of the constructions that go along with one of the main approaches to the Weierstrass preparation theorem [8, 9]. Since we do not know a reference for the full result, we include a proof. The statement in Corollary 2.2 is equivalent to the zero-sum relations that have been used in a similar context, e.g., [3, 6, 20, 21, 22, 23, 24].

It is natural to consider functions $f(x_1,\ldots,x_d)$ of points $(x_1,\ldots,x_d)\in\mathbb{C}^d$ which are invariant under the symmetric group S_d of permutations of the variables. To be precise, given any permutation, $\sigma(i)=j_i$ for $i=1,\ldots,d$, of the integers from 1 to d, we have a linear transformation of $\mathbb{C}^d$, which by abuse of notation we also label σ, which takes $(x_1,\ldots,x_d)$ to $(x_{j_1},\ldots,x_{j_d})$. A function on $\mathbb{C}^d$ is said to be *symmetric* if

$$f\circ\sigma = f(x_{j_1},\ldots,x_{j_d}) = f(x_1,\ldots,x_d) \tag{2.1}$$

for all $\sigma\in S_d$. It is a basic fact of invariant theory that the ring of symmetric polynomials on $\mathbb{C}^d$, denoted by $\mathbb{C}[x_1,\ldots,x_d]^{S_d}$, is abstractly isomorphic to the ring of polynomials on $\mathbb{C}^d$, i.e., there exists a ring isomorphism $\mathbb{C}[z_1,\ldots,z_d]\cong\mathbb{C}[x_1,\ldots,x_d]^{S_d}$. There are many useful choices of assignments of symmetric functions to the z_i making this isomorphism explicit. The two that we use are

1. the assignment leading to the elementary symmetric functions

$$z_i\mapsto t_i := \frac{1}{(d-i)!i!}\sum_{\sigma\in S_d} x_{\sigma(1)}\cdots x_{\sigma(i)} = \sum_{1\le j_1<\ldots<j_i\le d} x_{j_1}\ldots x_{j_i} \tag{2.2}$$

   for i from 1 to d, and
2. the assignment leading to the power sums

$$z_i\mapsto p_i := \frac{1}{(d-1)!}\sum_{\sigma\in S_d} x_{\sigma(1)}^i = \sum_{j=1}^{d} x_j^i \tag{2.3}$$

   for i from 1 to d.

The connection of t_i with roots of a polynomial of degree d is easy to see, upon noting that

$$(w-x_1)(w-x_2)\cdots(w-x_d) = w^d - t_1w^{d-1} + t_2w^{d-2} - \cdots + (-1)^d t_d. \tag{2.4}$$

If, in the right-hand side of (2.4), we (like in [16]) substitute w by x_i, for i from 1 to d, and add up these n sums, then we obtain Newton's relation:

$$p_d - t_1 p_{d-1} + t_2 p_{d-2} - \cdots + d(-1)^d t_d = 0. \tag{2.5}$$

Equation (2.5) allows us to write the elementary symmetric functions in terms of the power sums and vice versa.

The following fact is classical.

THEOREM 2.1. *Let $p : X \to Y$ be a proper, finite, d-sheeted surjective complex analytic morphism from a reduced pure n-dimensional complex analytic space X to a normal irreducible and reduced complex analytic variety Y. Given a holomorphic function $f : X \to \mathbb{C}$ and a symmetric polynomial $g : \mathbb{C}^d \to \mathbb{C}$, there is a unique holomorphic function $f_g : Y \to \mathbb{C}$ such that, for all points y in the Zariski open and dense set $U \subset Y$ such that $p : p^{-1}(U) \to U$ is an unramified cover, it follows that for the d points $\{x_1, \ldots, x_d\} = p^{-1}(y)$, we have $f_g(y) := g(f(x_1), \ldots, f(x_d))$.*

For the symmetric functions t_i we denote f_{t_i} by $\mathrm{tr}_{i,p}(f)$ and call it the ith trace. It is traditional to call $\mathrm{tr}_{1,p}(f)$, or $\frac{\mathrm{tr}_{1,p}(f)}{d}$, the trace and $\mathrm{tr}_{d,p}(f)$ the norm of f. In fact, the t_i all occur naturally as traces, e.g., letting $A : \mathbb{C}^d \to \mathbb{C}^d$ denote a matrix with eigenvalues $\{x_1, \ldots, x_d\}$, t_i is the trace of the matrix A induces on $\wedge^i \mathbb{C}^d \cong \mathbb{C}^{\binom{d}{i}}$, the ith exterior product of $\mathbb{C}^d$. The parameterized version of (2.4) is

$$f^d - \mathrm{tr}_{1,p}(f) f^{d-1} + \mathrm{tr}_{2,p}(f) f^{d-2} + \cdots + (-1)^d \mathrm{tr}_{d,p}(f) = 0. \tag{2.6}$$

This line of reasoning is used in one main approach to the Weierstrass preparation theorem [8, 9].

Proof of Theorem 2.1. The proof that f_g is holomorphic follows by a minor variant of the argument used in [8, Theorem A4]. To see this, note that there is a codimension one analytic subset $B \subset Y$ such that $X \setminus p^{-1}(B)$ and $Y \setminus B$ are smooth and such that $p_{X \setminus p^{-1}(B)} : X \setminus p^{-1}(B) \to Y \setminus B$ is a d-sheeted unramified cover. Indeed, define U' equal to Y minus the union of

1. the singular set $\mathrm{Sing}(Y)$ of Y, which is an analytic set of complex codimension 2 [8, Theorem Q12], and
2. the image under p of the singular set of X, which is an analytic set by Remmert's proper mapping theorem [8, Theorem N1].

Thus $p_{p^{-1}(U')} : p^{-1}(U') \to U'$ is a proper and finite map between complex manifolds. The set of branch points R of this map is an analytic set on U', since it is the set defined by the local condition that the determinant of the Jacobian of the mapping $p_{p^{-1}(U')}$ is zero. Define $U := U' \setminus p(R)$.

Thus for $y \in Y \setminus B$, $f_g(y) := g(f(x_1), \ldots, f(x_d))$ is a holomorphic function. Since p is proper, given any point $y \in B$, there is a relatively compact open set U containing y such that $p^{-1}(U)$ is relatively compact and f is bounded in absolute value on $p^{-1}(U)$. Thus f_g is bounded on $U \setminus (B \cap U)$. Thus by the Riemann extension theorem for bounded holomorphic functions, $f_{g|U}$ has a unique holomorphic extension to $Y \setminus \mathrm{Sing}(Y)$, where $\mathrm{Sing}(Y)$ denotes the singular set of Y. Since Y is normal, it follows from the Levi extension theorem [8, Theorem Q15i] that $f_{g|Y \setminus \mathrm{Sing}(Y)}$ has a unique holomorphic extension to Y. □

In Theorem 2.1, if in addition it is assumed that X, Y, p, and f are algebraic, then it follows that f_g is also algebraic. Rather than introduce all the needed definitions and algebraicity criteria to state the general case, we prove only a corollary that covers our needs.

COROLLARY 2.2. *Let $Z \subset \mathbb{C}^n$ be a pure k-dimensional algebraic subvariety of $\mathbb{C}^n$. Assume*

1. *that $\pi : \mathbb{C}^n \to \mathbb{C}^k$ is a generic linear projection, and*
2. *that ϕ is a linear function on $\mathbb{C}^n$, which is one-to-one on a fiber $\pi_Z^{-1}(y)$ for some $y \in \mathbb{C}^k$ with $\pi^{-1}(y)$ consisting of smooth points of Z at which the tangent space, $d\pi_Z$, has rank k.*

Then, it follows that for all j, $\mathrm{tr}_{j,\pi_Z}(\phi)$ is a polynomial on $\mathbb{C}^k$ of degree less than or equal to j. In particular, $\mathrm{tr}_{1,\pi_Z}(\phi)$ is linear or constant.

Proof. Let q denote the map $\mathbb{C}^n \to \mathbb{C}^{k+1}$ given by (ϕ, π). Let z denote the coordinate on $\mathbb{C}^{k+1}$ such that $z(q(\mathbf{x})) = \pi(\mathbf{x})$, and let L denote the projection of $\mathbb{C}^{k+1} \to \mathbb{C}^k$ such that $\pi = L \circ q$. By the Noether normalization theorem it follows that the genericity of π implies that $p := \pi_Z$ is a proper finite-to-one morphism. Since $\pi_Z = L \circ q_Z$, it follows by elementary point set topology that q_Z is proper and finite also. Moreover, by genericity it follows that the degree d of p is $\deg Z$, by the hypothesis on ϕ that q_Z maps Z generically one-to-one to $\mathbb{C}^{k+1}$, and, therefore, that $\deg q(Z) = \deg Z$. Since for a dense open set $U \subset \mathbb{C}^k$ q gives an isomorphism of $p^{-1}(U)$ with $L_{q(Z)}^{-1}(U)$, we conclude that $\mathrm{tr}_{j,p}(\phi_Z)$ and $\mathrm{tr}_{j,L_{q(Z)}}(z_{q(Z)}$ agree on U and hence on all of $\mathbb{C}^k$. Thus, with the convention that $\mathrm{tr}_{0,L_{q(Z)}}(z_{q(Z)}) = \mathrm{tr}_{0,p}(\phi_Z) = 1$, we have the equivalent relations given in (2.6) $f^d - \mathrm{tr}_{1,p}(f) f^{d-1} + \cdots + (-1)^d \mathrm{tr}_{d,p}(f) = 0$,

$$\sum_{i=0}^{d} (-1)^i \mathrm{tr}_{i,L_{q(Z)}}(z_{q(Z)}) z_{q(Z)}^{d-i} = 0, \tag{2.7}$$

$$\sum_{i=0}^{d} (-1)^i \mathrm{tr}_{i,p}(\phi_Z) {\phi_Z}^{d-i} = 0. \tag{2.8}$$

Since $q(Z)$ is a degree d hypersurface and $z^d - \mathrm{tr}_{1,p}(z_{q(Z)}) z^{d-1} + \cdots + (-1)^d \mathrm{tr}_{d,p}(z_{q(Z)})$ vanishes when restricted to it, we conclude that this must be a minimum degree defining polynomial of $q(Z)$. Thus we have proved the assertions of the corollary. ∎

Remark 2.3. Note that, assuming the genericity hypothesis on π in Corollary 2.2, the hypothesis on ϕ can be replaced by the equivalent more easily checked condition that ϕ is a linear function on $\mathbb{C}^n$, which is one-to-one on a fiber $\pi_Z^{-1}(y)$ for some $y \in \mathbb{C}^k$ with $\pi^{-1}(y)$ consisting of $\deg Z$ distinct points.

Remark 2.4. Note that, without the genericity assumption, Corollary 2.2 fails for a number of reasons. First, it might be that π_Z is not proper. In this case, the traces are only rational functions. For example, taking $Z := \{xy - 1 = 0\} \subset \mathbb{C}^2$ and $\pi : \mathbb{C}^2 \to \mathbb{C}$ given by $\pi(x, y) = x$, we get $\mathrm{tr}_{1,\pi_Z}(y) = \frac{1}{x}$. By the Noether normalization theorem, the genericity assumption about the linear projection π implies that π_Z is proper and finite, but even proper and finite, without the genericity assumption, is not enough. The key implication of genericity of the linear projection π, beyond the properness of π_Z, is the fact that for generic linear projections π, $\deg Z = \deg \pi_Z$. For example, taking $Z := \{y^2 - x^d = 0\} \subset \mathbb{C}^2$ and $\pi : \mathbb{C}^2 \to \mathbb{C}$ given by $\pi(x, y) = x$, we get $\mathrm{tr}_{2,\pi_Z}(y) = x^d$. In general, if

1. $Z \subset \mathbb{C}^n$ is a pure k-dimensional algebraic subvariety of $\mathbb{C}^n$,
2. f is linear on $\mathbb{C}^n$, and
3. $\pi : \mathbb{C}^n \to \mathbb{C}^k$ is a linear projection with π_Z proper, finite, but not necessarily satisfying $\deg Z = \deg \pi_Z$,

then it follows that $\deg \mathrm{tr}_{i,\pi_Z}(f) \leq \deg Z - \deg \pi + i$.

3. An application to monodromy. In this section we show that the linear trace gives necessary and sufficient conditions to determine the breakup of the set witness points of the algorithm of [27] into the disjoint subsets of generic points corresponding to the numerical irreducible decomposition. Usually, we use this result to give a very fast verification of the monodromy breakup of the algorithm of [29], but it is also called into play if the monodromy breakup is too fine. The algorithms will be described in the next section.

The important observation that the linear trace is sufficient is due to Rupprecht [20] in the case of curves. We note that there are two serious gaps in the argument of [20], which are filled by Theorems 3.3 and 3.4 below.

The strategy is to slice and project to reduce to the case of a curve in $\mathbb{P}^2$. To do this we need a number of lemmas on linear projections and the intersections with linear spaces that are general subject to certain constraints. Unless otherwise said, closure is in either the Zariski topology or the usual topology induced by the Euclidean metric on $\mathbb{C}^n$.

LEMMA 3.1. *Let A be a pure k-dimensional reduced algebraic subset of $\mathbb{C}^n$, with irreducible decomposition $\cup_{i=1}^r A_i$. Assume that L is an $(n-k)$-dimensional linear subspace of $\mathbb{C}^n$ meeting A in a finite set $\mathcal{A}$ consisting of $\deg A$ distinct isolated points. Then, taking closures in $\mathbb{P}^n$, $\overline{L} \cap \overline{A} = \mathcal{A}$. Moreover, if $k \geq 2$, then letting $\mathcal{L}$ be a general member of the set of $(n-k+1)$-dimensional linear subspaces of $\mathbb{C}^n$ that contain L, it follows that $\mathcal{L} \cap A_i$ is an irreducible curve for each $i = 1, \ldots, r$.*

Proof. The statement that $\overline{L} \cap \overline{A} = \mathcal{A}$ follows from any of a number of related results, e.g., [5, Example 8.4.6] or [5, Example 12.3.2].

To prove the second statement it suffices to prove the analogous result on $\mathbb{P}^n$ using the closures of the sets A_i, L, $\mathcal{L}$. Since $\overline{L} \cap \overline{A}$ is a set $\mathcal{A}$ of cardinality $\deg \overline{A}$, it follows that $\mathcal{A}$ consists of smooth points of $\overline{A}$, and that the intersection of $\overline{A}$ and $\overline{L}$ are transverse at the points of intersection. From this it follows from Bertini's theorem that the intersection with A_i of a general member of the set M of $(n-k+1)$-dimensional linear subspaces of $\mathbb{C}^n$ that contain L is smooth away from the singular locus of A_i. If $k = 2$, so that the set M consists of hyperplanes, the rest of the argument follows exactly as in [25, Theorem 3.42]. If $k \geq 2$, the result follows by a straightforward descending induction. For example, if $k = 3$, then it follows using [25, Theorem 3.42] that the intersection with A_i of a general member of the set of $(n-1)$-dimensional linear subspaces H of $\mathbb{C}^n$ that contain L is irreducible. Keeping L as it is, taking H in place of $\mathbb{C}^n$, and replacing the A_i with $A_i \cap H$, we now have $\dim A_i \cap H = 2$, i.e., we have reduced to the proven result. ∎

LEMMA 3.2. *Let n, L, $\mathcal{L}$, $\mathcal{A}$, $A = A_1 \cup \cdots \cup A_r$ be as in Lemma 3.1. Choose a general linear projection $\pi : L \to \mathbb{C}$, which is one-to-one on $\mathcal{A}$. Let $\widetilde{\pi} : \mathbb{C}^n \to \mathbb{C}^{k+1}$ be a linear map extending π, so that the fibers of $\widetilde{\pi}$ are parallel to the fibers of π. Then $\widetilde{\pi}_A$ and $\widetilde{\pi}_{A \cap \mathcal{L}}$ are generically one-to-one and proper. Moreover, $\widetilde{\pi}_A$ (respectively, $\widetilde{\pi}_{A \cap \mathcal{L}}$), maps a neighborhood of $\mathcal{A}$ in A (respectively, in $A \cap \mathcal{L}$) isomorphically onto a neighborhood of the image of $\mathcal{A}$ in A (respectively, in $A \cap \mathcal{L}$).*

Proof. We give the proof that $\widetilde{\pi}_A$ is generically one-to-one and proper, and we leave the remaining argument, which follows the same line of reasoning to the reader. We work in the projective space $\mathbb{P}^n$. As explained in [27], $\widetilde{\pi}$ corresponds to the central projection from a linear $I := \mathbb{P}^{n-k-2}$ contained in the linear $\mathbb{P}^{n-1}$ at infinity, i.e., in $\mathbb{P}^n \setminus \mathbb{C}^n$. The condition that L is mapped to a line corresponds to $I \subset \overline{L} \setminus L$. Since by Lemma 3.1 we know that $(\overline{L} \setminus L) \cap (\overline{A} \setminus A) = \emptyset$, we know, as discussed in [27], that $\widetilde{\pi}_A$ is proper and therefore finite-to-one on A. If $\widetilde{\pi}_A$ was not generically one-to-one, then

$\deg \widetilde{\pi}(A)$ would be less than $\deg A$. But this does not happen since $\widetilde{\pi}$ is one-to-one on $\mathcal{A}$, and the degree of $\widetilde{\pi}(A)$ is the cardinality of $\widetilde{\pi}(\mathcal{A})$, which is equal to the intersection of $\widetilde{\pi}(A)$ with the line $\widetilde{\pi}(L)$. ∎

THEOREM 3.3. *Let A be a reduced pure k-dimensional algebraic subset of $\mathbb{C}^n$. Let $A := A_1 \cup \cdots \cup A_r$ be the decomposition into distinct irreducible components. Let $L \subset \mathbb{C}^n$ be a general linear subspace of dimension $n-k$ meeting A transversely in the set $\mathcal{A} := A \cap L$. For all $i = 1, \ldots, r$, let $\mathcal{A}_i := A_i \cap L$, and let d_i be the cardinality of the set $\mathcal{A}_i$. Let d denote the cardinality of $\mathcal{A}$, i.e., $\sum_{i=1}^{r} d_i$. Let U denote the Zariski open set of the Grassmannian of $(n-k)$-dimensional linear subspaces of $\mathbb{C}^n$ consisting of linear spaces transverse to A. Let* $\mathrm{Sym}(\mathcal{A})$ *(respectively,* $\mathrm{Sym}(\mathcal{A}_i)$*) denote the symmetric group of all permutations of $\mathcal{A}$ (respectively, $\mathcal{A}_i$). Considering L as a basepoint of U, the image in* $\mathrm{Sym}(\mathcal{A})$ *of the natural monodromy action of $\pi_1(U, L)$ on $\mathcal{A}$ is the direct sum* $\bigoplus_{i=1}^{r} \mathrm{Sym}(\mathcal{A}_i)$.

Proof. We can assume without loss of generality that no components of A are linear, since such components do not affect the result.

Choose $\mathcal{L}$ and $\widetilde{\pi} : \mathbb{C}^n \to \mathbb{C}^{k+1}$ as in Lemma 3.2. Let U' denote the Zariski open set of the projective space of lines in $\mathbb{C}^{k+1}$, consisting of the lines transverse to $\widetilde{\pi}(A)$. Note that for $\ell \in U'$ we have that $\widetilde{\pi}^{-1}(\ell) \in U$. Thus identifying $\mathcal{A}$ with $\widetilde{\pi}(\mathcal{A})$, the homomorphism $\pi_1(U', \widetilde{\pi}(L)) \to \mathrm{Aut}(\mathcal{A})$ factors $\pi_1(U', \widetilde{\pi}(L)) \to \pi_1(U, L) \to \mathrm{Aut}(\mathcal{A})$. Thus, from here on we can assume without loss of generality that $n = k+1$. Moreover, a line in $\mathcal{L}$ transverse to $\mathcal{L} \cap A$ is transverse to A in $\mathbb{C}^n$. Thus, letting U'' denote the Zariski open set of lines in $\mathcal{L} \cong \mathbb{C}^2$ that are transverse to $\mathcal{L} \cap A$, we have a composition $\pi(U'', L) \to \pi_1(U, L) \to \mathrm{Aut}(\mathcal{A})$. Thus, without loss of generality we can assume that $n = 2$.

Thus we have $n = 2$ and $k = 1$. We have the classical fact [1, Lemma, p. 111] that the image of $\pi_1(U, L) \in \mathrm{Aut}(\mathcal{A})$ surjects onto $\mathrm{Sym}(\mathcal{A}_i)$ for each $i = 1, \ldots, r$. In particular, we can assume without loss of generality that $r \geq 2$. By elementary algebra, we see that using this surjectivity, we would be done if we showed that, for each i, there exist two distinct points $a, b \in \mathcal{A}_i$ and a $\gamma \in \pi_1(U, L)$ such that γ acts on $\mathcal{A}$ by sending $a \to b$, $b \to a$ and leaves the remaining points of $\mathcal{A}$ fixed. The classical argument for the existence of such a γ in the irreducible case [1, Lemma, p. 111] carries over with no change to the reducible case if we show that for each $i = 1, \ldots, r$, and a generic point $x \in \mathcal{A}_i$, the tangent line $\ell \subset \mathbb{C}^2$ to A_i at x is transverse to A_j for $j \neq i$. To show this, it suffices to work projectively, i.e., show the fact for the closures B_i in $\mathbb{P}^2$ of the A_i. To see this, consider the dual curves $\widehat{B}_i \subset (\mathbb{P}^2)^*$. Here $(\mathbb{P}^2)^*$ is the $\mathbb{P}^2$ whose points correspond to lines in the $\mathbb{P}^2$ that the B_i belong to, and $\widehat{B}_i$ is the closure in $(\mathbb{P}^2)^*$ of the set of points corresponding to tangent lines to smooth points of B_i. What we need is exactly that B_i and B_j for distinct i, j go to distinct curves $\widehat{B}_i$ and $\widehat{B}_j$. Noting that none of the B_i are linear, this would follow if we knew that the dual of $\widehat{B}_i$ is B_i. This is a basic fact about dual curves (and more generally varieties) [14]. ∎

To prove Corollary 3.5 below, we need the following generalization of the classical first Lefschetz theorem. This topological result is a special case of a useful general result of Goresky and MacPherson [7, Theorem, section 5.2, p. 199].

THEOREM 3.4 (Goresky–MacPherson). *Let D be an arbitrary algebraic subset of $\mathbb{C}^n$, and let $U := \mathbb{C}^n \setminus D$. Then given a general 1-dimensional linear subspace $L \subset \mathbb{C}^n$ and a point $x \in L \cap U$, it follows that we have a surjective map of fundamental groups*

$$\pi_1(L \cap U, x) \to \pi_1(U, x) \to 0.$$

Proof. Following the notation of [7, section 5.2, Theorem, p. 199], take X to be the Zariski open dense set U of $\mathbb{P}^n$ with $n := N$, π the inclusion, $c = n-1$, which gives $\phi(k) = n-1$ for $k = 0$, and $-\infty$ for $k \neq 0$, which gives $\hat{n} = 1$. □

COROLLARY 3.5. *Let A be a reduced pure k-dimensional algebraic subset of $\mathbb{C}^n$. Let $A := A_1 \cup \cdots \cup A_r$ be the decomposition into distinct irreducible components. Let $\pi : \mathbb{C}^n \to \mathbb{C}^k$ denote a generic linear projection, and let $x, y \in \mathbb{C}^k$ denote general points, with $L := \pi^{-1}(x) \subset \mathbb{C}^n$ a general linear subspace of dimension $n-k$ meeting A transversely in the set $\mathcal{A} := A \cap L$. For all $i = 1, \ldots, r$, let $\mathcal{A}_i := A_i \cap L$, and let d_i be the cardinality of the set $\mathcal{A}_i$. Let d denote the cardinality of $\mathcal{A}$, i.e., $\sum_{i=1}^r d_i$. Let U denote the Zariski open set of the line $\ell \subset \mathbb{C}^k$ containing x, y, consisting of the $u \in \ell$ such that $\pi^{-1}(u)$ is transverse to A. Let* $\mathrm{Sym}(\mathcal{A})$ *(respectively,* $\mathrm{Sym}(\mathcal{A}_i)$*) denote the symmetric group of all permutations of $\mathcal{A}$ (respectively, $\mathcal{A}_i$). Considering L as a basepoint of U, the image in* $\mathrm{Sym}(\mathcal{A})$ *of the natural monodromy action of $\pi_1(U, L)$ on $\mathcal{A}$ is the direct sum* $\bigoplus_{i=1}^r \mathrm{Sym}(\mathcal{A}_i)$.

Proof. By the same reduction as in Theorem 3.3, it can be assumed that $n = 2$, $k = 1$, and that we are working with compact curves in $\mathbb{P}^2$. Given the Zariski open set U' of points in $(\mathbb{P}^2)^*$ corresponding to lines transverse to A, and given a general line $\ell \subset (\mathbb{P}^2)^*$, with a point L on ℓ, then setting $U := \ell \cap U'$, we will be done if we show that $\pi_1(U, L) \to \pi_1(U, L) \to 0$. This is guaranteed by Theorem 3.4. □

THEOREM 3.6. *Let $A \subset \mathbb{C}^n$ be an affine algebraic set of pure dimension k. Let $A := A_1 + \cdots + A_r$ denote the irreducible decomposition of A. Let $\pi : \mathbb{C}^n \to \mathbb{C}^k$ be a generic linear projection, and let $\ell \subset \mathbb{C}^k$ be a general line. Let $L := \pi^{-1}(\mathbf{x})$ for a general point $\mathbf{x} \in \ell$. Let ϕ be a linear function on $\mathbb{C}^n$ which is one-to-one on $\mathcal{A} := \pi_A^{-1}(\mathbf{x})$. For all $i = 1, \ldots, r$, let $\mathcal{A}_i := \pi_{A_i}^{-1}(\mathbf{x})$. Let U denote the Zariski open set of the Grassmannian of $(n-k)$-dimensional linear spaces of $\mathbb{C}^n$ corresponding to $(n-k)$-dimensional linear spaces transverse to A. Let B denote a subset of $\mathcal{A}$. Then the following are equivalent:*

(1) *B is invariant under the monodromy action of $\pi_1(U, L)$ on $\mathcal{A}$;*
(2) *$B = \cup_{i \in I} \mathcal{A}_i$ for some subset $I \subset \{1, \ldots, r\}$;*
(3) *The analytic continuation of $\sum_{b \in B} \phi(b)$ as a function of $\mathbf{x}$ is linear.*

Proof. The equivalence of (1) and (2) follows from Theorem 3.3. Corollary 2.2 shows that (2) implies (3). So it remains to show that (3) implies (1).

To see this, assume that B contains a point $b \in \mathcal{A}_i$ but not a point $a \in \mathcal{A}_i$. Let $y_1, \ldots, y_N$ denote the points of $B \setminus b$. Let U' denote the Zariski open subset of ℓ consisting of the points $\mathbf{x}' \in \ell$ with $\pi^{-1}(\mathbf{x}')$ transverse to A. By Corollary 3.5, there is a $\gamma \in \pi_1(U', L)$ which takes $y_j \to y_j$ for all j and interchanges a and b. Thus, since the analytic continuation of $\sum_{b \in B} \phi(b)$ is linear in $\mathbf{x}$, we conclude that

$$\phi(a) + \sum_{j=1}^{N} \phi(y_j) = \phi(b) + \sum_{j=1}^{N} \phi(y_j), \tag{3.1}$$

and thus that $\phi(a) = \phi(b)$. But this contradicts ϕ being one-to-one on $\mathcal{A}$. Thus if $b \in \mathcal{A}_i$ for some point $b \in B$, we conclude that $\mathcal{A}_i \subset B$. □

Remark 3.7. Here is a simple example to show the sort of bad behavior that genericity rules out. Let A be the curve in $\mathbb{C}^2$ defined by $(y^2 - x)(y^2 - 4x)(y^2 - 9x) = 0$. Consider the projection $\pi : \mathbb{C}^2 \to \mathbb{C}$ given by $\pi(x, y) = x$. Note that π_A is proper and generically six-to-one with $\deg A = 6$. For the linear function ϕ on $\mathbb{C}^2$, choose y. Over $x \in \mathbb{C}$, the values of ϕ on the fiber $\pi_A^{-1}(y)$ are

$$\{\sqrt{x}, -\sqrt{x}, 2\sqrt{x}, -2\sqrt{x}, 3\sqrt{x}, -3\sqrt{x}\}. \tag{3.2}$$

The groupings corresponding to the irreducible components are $\{\sqrt{x}, -\sqrt{x}\}$, $\{2\sqrt{x}, -2\sqrt{x}\}$, and $\{3\sqrt{x}, -3\sqrt{x}\}$. Notice that the sum $(-1)\sqrt{x} + (-2)\sqrt{x} + 3\sqrt{x}$ is identically zero, and hence linear, even though the grouping $\{-\sqrt{x}, -2\sqrt{x}, 3\sqrt{x}\}$ does not correspond to a union of irreducible components of A.

Remark 3.8. In practice, when we use Theorem 3.6, it is convenient to use a generic projection $\pi : \mathbb{C}^n \to \mathbb{C}^{k+1}$. Letting $A := A_1 + \cdots + A_r$ be as in Theorem 3.6, it follows, e.g., from [27, section 5.2], that

1. the map π_A from A to its image $\pi(A)$ is proper,
2. the images of $A, A_1, \ldots, A_r$ under π are affine algebraic sets with $\pi(A)$ having the irreducible decomposition $\pi(A) := \pi(A_1) + \cdots + \pi(A_r)$,
3. $\deg \pi(A_i) = \deg A_i$ for all i, and
4. π_A is one-to-one on $\pi_A^{-1}(U)$ for some Zariski open dense set of $\pi(A)$.

Moreover, since the composition of π with the projection $\mathbb{C}^{k+1} \to \mathbb{C}^k$ given by $(z_1, \ldots, z_{k+1}) \to (z_1, \ldots, z_k)$ is generic, we can use $\pi(A)$ and this projection in place of A and the generic projection π of Theorem 3.6. Then, z_{k+1} has the properties required of ϕ in Theorem 3.6, and, since the projection $\pi : \mathbb{C}^n \to \mathbb{C}^{k+1}$ is generic, we can take one of the coordinate axes, e.g., the z_k axis, as ℓ.

It is worth noting that the defining equation of $\pi(A)$ under a generic projection will have every monomial of total degree less than or equal to the degree of A occurring, no matter how sparse the defining equations of A are.

4. Algorithms for the linear trace.

In this section we present an algorithm to verify the decomposition predicted by the monodromy algorithm. We first define a projection operator which organizes the samples in a structured grid. The main part of this section is the algorithm **Certify**, followed by comments on how to integrate this algorithm in the numerical irreducible decomposition of a pure dimensional component.

4.1. Sampling on parallel slices.

To compute the linear trace, a structured grid of sample points is useful. The same construction is used in the following section concerning higher traces. Our technique is to use random slicing hyperplanes to define the projection operator π, as follows.

DEFINITION 4.1. *Consider a k-dimensional component in $\mathbb{C}^n$ and suppose we use the k hyperplanes*

$$c_{i0} + c_{i1}x_1 + c_{i2}x_2 + \cdots + c_{in}x_n = 0, \qquad i = 1, 2, \ldots, k, \tag{4.1}$$

as slices to obtain generic points on the component. To project the generic points down to $\mathbb{C}^{k+1}$, we use the map $\pi : \mathbb{C}^n \to \mathbb{C}^{k+1}$ defined by

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} \mapsto \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{k1} & c_{k2} & \cdots & c_{kn} \\ a_1 & a_2 & \cdots & a_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix}, \tag{4.2}$$

where the numbers a_i, $i = 1, 2, \ldots, k$ are chosen at random.

The main property of this projection operator is highlighted in the following proposition.

PROPOSITION 4.2. *For a k-dimensional component in $\mathbb{C}^n$, let $\pi : \mathbb{C}^n \to \mathbb{C}^{k+1}$ be as in* (4.2). *Then for any generic point $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ on the slices used in the definition of π, we have*

$$\pi(\mathbf{x}) = (-c_{10}, -c_{20}, \ldots, -c_{k0}, a_1x_1 + a_2x_2 + \cdots + a_nx_n). \tag{4.3}$$

To obtained a structured grid of samples from the k-dimensional component, we let only the constant terms c_{i0} of the slicing planes vary, for $i = 1, 2, \ldots, k$. Geometrically this means we take samples on slices parallel to each other.

4.2. Certification of monodromy groupings with linear traces. Suppose we are given a set S of d generic points on random hyperplanes $L = (L_1, L_2, \ldots, L_k)$, where the points are known to be from the same k-dimensional irreducible component because the monodromy algorithm found loops connecting them. As the monodromy algorithm might miss some connections, the actual degree of the component could be higher than d. With linear traces we verify whether the degree of the component equals d, described by the **Certify** algorithm.

ALGORITHM 4.3. $b = \textbf{Certify}(f, L, S, \epsilon)$

Input: $f(\mathbf{x}) = \mathbf{0}$ is a polynomial system with $\mathbf{x} \in \mathbb{C}^n$;
$L = (L_1, L_2, \ldots, L_k)$ is a tuple of k random hyperplanes;
S is set of d generic points satisfying $f(\mathbf{x}) = \mathbf{0}$ and $L(\mathbf{x}) = \mathbf{0}$;
ϵ is tolerance to decide whether a number is close enough to zero.

Output: $b \in \{\text{false}, \text{true}\}$, b is true when S is a set of generic points on a degree d irreducible component, false otherwise.

let $S^{(0)} = S$, $c_0^{(0)} = c_0$; [*notational convenience*]
for $i = 1, 2$ do [*sample to get test points*]
  choose $c_0^{(i)} \in \mathbb{C}$ at random;
  $L_k^{(i)} := c_0^{(i)} + c_{k1}x_1 + c_{k2}x_2 + \cdots + c_{kn}x_n$; [$L_k^{(i)}$ *is parallel to* L_k]
  compute $S^{(i)}$ as solutions to $f(\mathbf{x}) = \mathbf{0}$, [*apply homotopy from* L_k *to* $L_k^{(i)}$
    and $L_1(\mathbf{x}) = L_2(\mathbf{x}) = \cdots = L_k^{(i)}(\mathbf{x}) = 0$; *using* S *as start solutions*]
end for;
use L to define $\pi : \mathbb{C}^n \to \mathbb{C}^{k+1}$ as in (4.2); [*projection operator*]
let $\phi(\mathbf{x}) = z_{k+1}$, where $z = \pi(\mathbf{x})$; [*definition of* ϕ *in Theorem* 3.6]
for $i = 0, 1, 2$ compute $s_i := \sum_{\mathbf{x} \in S^{(i)}} \phi(\mathbf{x})$; [*sum* $(k+1)$*st coordinate*]
find a, b such that $s_i = a + bc_0^{(i)}$ for $i = 0, 1$; [*linear interpolation of trace*]
return $(|s_2 - (a + bc_0^{(2)})| < \epsilon)$. [*the comparison certifies*]

The justification for this algorithm is Theorem 3.6. The first k coordinates of $\pi(\mathbf{x})$ in (4.2) are the generic projection required by the theorem and the $(k+1)$st coordinate is the linear function ϕ. We test linearity of the trace by sampling in a generic direction: the kth coordinate of $\pi(\mathbf{x})$ suffices due to the genericity of the coefficients used to define it. By the theorem, linearity implies that the set of points S is the union of witness points for irreducible sets whose degrees sum to d, while by assumption, monodromy has found that all the points are in one irreducible set. Thus, there is one set and its degree is d.

4.3. An integrated decomposition algorithm. The linear trace test can be used as a simple replacement for the filtering polynomials used in our earlier papers [27, 28, 29], but additional efficiencies can be gained by integrating the technique more deeply into the algorithms.

First, we can improve the termination condition for the monodromy method. Previously, we continued to compute monodromy loops until either all points are connected into one group or until some preset number of consecutive loops fails to find any new connections. Setting the number of these stable loops too high is costly, while too low means that some connections could be missed. With the linear trace test, one can determine when a group is complete and immediately remove it from further iterations. Once the number of uncertified groups is reduced to a small number, combinations of them can be examined to discover which ones sum to form linear traces, thereby completing the decomposition without further monodromy loops.

In this vein, it is possible to perform the decomposition only using linear traces, as is done for a single multivariate polynomial in [6, 20]. However, without monodromy, the algorithm is combinatorial and is likely to be too expensive for high degrees. The use of traces in [21] (with predecessors [22, 23, 24]) is followed by linear algebra techniques. Recently, monodromy and traces have been combined to factor a single multivariate polynomial in [3].

The factorization of a single multivariate polynomial can be regarded as a special case of the decomposition of the solution sets of polynomial systems. For this general problem, we indicate a second improvement. In implementing the monodromy algorithm in [29], it is worthwhile to compute the linear span of the components as we described in [28]. Generic points that lie in different spans lie on different irreducible components, so we have only to execute the monodromy starting at points that lie in the same linear span. Also, the restriction to the linear span will give a speedup when there is a gap between the dimension of the linear span and the dimension of the ambient space.

Finally, the main decomposition algorithm [27] requires a test to determine if a generic point obtained from the embedding algorithm at dimension k is a member of some irreducible set of dimension greater than k. Originally, we used the interpolating polynomials for these sets to determine membership, but as in [28], it is possible to use a homotopy test of membership. Using the homotopy membership test for higher dimensional sets and certifying irreducible groups by linear traces, we eliminate completely the expensive and numerically difficult step of computing interpolating polynomials, which represents a big improvement in our overall algorithm.

5. Interpolation algorithms via traces. As just mentioned, the computation of interpolating polynomials is no longer required to complete the numerical irreducible decomposition. Nevertheless, in the case that one still wishes to compute such polynomials, the higher order traces can be useful, as we show in this section. For components of low degree and span, interpolating polynomials can be competitive with a homotopy membership test.

Our techniques for computing interpolating polynomials can be briefly summarized as follows. Since the witness points for a component lie on a linear slice, they can be marched forward together to compute a structured grid of sample points. In [29], a "bootstrapping" technique was used to construct the Newton form of the interpolating polynomial. Here, by using traces, we eliminate the expense of extra samples for bootstrapping and apply Newton interpolation directly. Finally, using the Newton identities, we reduce the number of samples to the number of monomials, which is optimal.

5.1. Newton interpolation with divided differences. To interpolate a bivariate function $f(x, y)$ with a polynomial $p(x, y)$ of degree d, we need to sample

the function at points (a_i, b_j), for all i, j: $0 \leq i + j \leq d$. The Newton form of the interpolation polynomial $p(x, y)$ is classical (see, e.g., [12, 16]):

$$p(x,y) = \sum_{k=0}^{d} \sum_{l=0}^{d-k} f[a_0 \cdots a_k; b_0 \cdots b_l] \prod_{i=0}^{k-1} (x - a_i) \prod_{j=0}^{l-1} (y - b_j). \tag{5.1}$$

The coefficients $f[a_0 \cdots a_k; b_0 \cdots b_l]$ are divided differences, defined inductively. Starting with $f[a_k; b_l] = f(a_k, b_l)$, all divided differences are generated by

$$f[a_0 a_1 \cdots a_k; b_l] = \frac{f[a_0 a_1 \cdots a_{k-1}; b_l] - f[a_1 a_2 \cdots a_k; b_l]}{a_0 - a_k} \tag{5.2}$$

and

$$f[a_0 a_1 \cdots a_k; b_0 b_1 \cdots b_l]$$

$$= \frac{f[a_0 a_1 \cdots a_k; b_0 b_1 \cdots b_{l-1}] - f[a_0 a_1 \cdots a_k; b_1 b_2 \cdots b_l]}{b_0 - b_l}, \tag{5.3}$$

for $k = 0, 1, \ldots, d$ and $l = 0, 1, \ldots, d - k$. The efficient computation of divided differences is organized in a table, requiring only one vector of elements to store. Generalizing (5.1) to any number of variables is burdened only by notation.

The direct application of Newton interpolation is prevented because the interpolation points must lie on a grid structured for *all* directions. When we sample curves or surfaces we always have one last component which is different for all samples. To overcome this we may apply a "bootstrapping" technique. We explain the idea in the case of two variables. For $x = a_k$, we construct a univariate polynomial $p(y)$ interpolating through the roots. Note that at those roots, y is usually different from the chosen grid points b_l. Once we have $p(y)$, we use it to find $f(a_k, b_l) = p(b_l)$, and we have a complete structured grid on which the above formulas (5.1) apply. This construction generalizes to the Newton form of the interpolating polynomial to represent any surface of any degree and dimension. It was implemented and used in [29] to certify groupings predicted by the monodromy algorithm. We provide an alternative to the bootstrapping technique using traces, as explained next.

5.2. The trace form with a complete grid. With traces, the classical multivariate interpolation schemes with generalized divided differences are directly applicable. We will show how to interpolate with a polynomial p of degree d in three variables (x, y, z), where p is expressed like

$$p(x,y,z) = z^d - t_1(x,y) z^{d-1} + t_2(x,y) z^{d-2} - \cdots + (-1)^d t_d(x,y), \tag{5.4}$$

where $t_i(x, y)$ is the ith trace with $\deg(t_i(x, y)) = i$. To represent the polynomials $t_i(x, y)$ we use the Newton form (5.1) with coefficients constructed with divided differences, given in formulas (5.2) and (5.3).

The major cost in the construction of the interpolating polynomial is the number of required sample points. While the number of monomials grows exponentially as the degree d and dimension k of the irreducible component increases, the number of samples in a complete grid grows as $d(d+1)^k$, which is much faster than the number of monomials, as illustrated in Table 5.1.

TABLE 5.1

Number of terms for degrees d and dimensions k, respectively, increasing in the rows and columns, versus $d(d+1)^k$, which is the number of samples needed for the trace form using the full grid, without exploiting Newton identities.

Number of monomials					Number of samples				
$d\backslash k$	2	3	4	5	$d\backslash k$	2	3	4	5
1	3	4	5	6	1	2	4	8	16
2	6	10	15	21	2	6	18	54	162
3	10	20	35	56	3	12	48	192	768
4	15	35	70	126	4	20	100	500	2500
5	21	56	126	252	5	30	180	1080	6480

5.3. Using the Newton identities. Ideally, we would like to take no more samples than the number of monomials. Exploiting the Newton identities (2.5), we show how to achieve this goal on an example, the interpolation of a planar quartic. Figure 5.1 gives a schematic representation of the grid of sample points.

In the interpolation of a planar quartic, we compute the four traces consecutively. To compute the second trace t_2, we may already use t_1, and to compute t_3, we already dispose of t_1, t_2, and for t_4, we make use of t_1, t_2, and t_3. We show how this saves sample points:

1. At $x = a_2$, instead of four, we compute three samples (a_2, b_{21}), (a_2, b_{22}), (a_2, b_{23}) and compute b_{24} using the first trace t_1, evaluated at $x = a_2$:

$$b_{24} := t_1(a_2) - b_{21} - b_{22} - b_{23}. \tag{5.5}$$

2. At $x = a_3$, we know already the coefficient of the Newton forms of t_1 and t_2 and use continuation only for two samples: (a_3, b_{31}) and (a_3, b_{32}). For the values b_{33} and b_{34} we solve the system

$$\begin{cases} t_1(a_3) = b_{31} + b_{32} + b_{33} + b_{34}, \\ t_2(a_3) = b_{31}b_{32} + b_{31}b_{33} + b_{31}b_{34} + b_{32}b_{33} + b_{32}b_{34} + b_{33}b_{34}. \end{cases} \tag{5.6}$$

With the Newton identities (2.5), we compute from the values of the elementary symmetric functions $(t_1(a_3), t_2(a_3))$ the power sums for $x = a_3$: $(s_1(a_3), s_2(a_3))$. This means that the b_{ij}'s satisfy

$$\begin{cases} s_1(a_3) = b_{31} + b_{32} + b_{33} + b_{34}, \\ s_2(a_3) = b_{31}^2 + b_{32}^2 + b_{33}^2 + b_{34}^2 \end{cases} \tag{5.7}$$

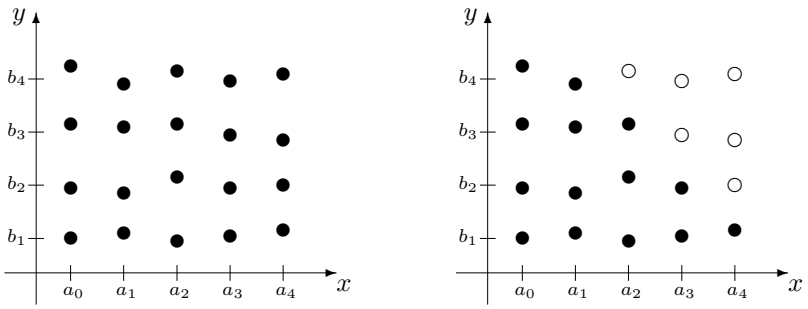

FIG. 5.1. *Two grids of sample points to interpolate a planar quartic: the grid on the left is complete, while on the right we find the white dots using Newton identities. The semiregularity of the grid (same x-value in one column, different y-values in each row) is typical.*

or

$$\begin{cases} b_{33} + b_{34} = s_1(a_3) - b_{31} - b_{32}, \\ b_{33}^2 + b_{34}^2 = s_2(a_3) - b_{31}^2 - b_{32}^2. \end{cases} \tag{5.8}$$

Let $\widetilde{s}_1(a_3) = s_1(a_3) - b_{31} - b_{32}$ and $\widetilde{s}_2(a_3) = s_2(a_3) - b_{31}^2 - b_{32}^2$. Then we invoke the Newton identities (2.5) to compute from the modified powers sums $(\widetilde{s}_1(a_3), \widetilde{s}_2(a_3))$ evaluated at a_3 into the values of the elementary symmetric functions at a_3: $(\widetilde{t}_1(a_3), \widetilde{t}_2(a_3))$. With $(\widetilde{t}_1(a_3), \widetilde{t}_2(a_3))$ we define

$$y^2 - t_1(a_3)y + t_2(a_3) = y^2 - (b_{33} + b_{34})y + b_{33}b_{34} \tag{5.9}$$

$$= (y - b_{33})(y - b_{34}) \tag{5.10}$$

$$= 0. \tag{5.11}$$

Thus, finding the missing samples (white dots on Figure 5.1 for $x = a_3$) has been reduced to solving a quadratic univariate equation.

3. At $x = a_4$ we apply the Newton identities also twice to construct a univariate equation of degree three to find the missing samples.

This procedure generalizes to any degree and any dimension, requiring only as many samples as the number of monomials.

To find the roots of univariate polynomials, we use the method of Weierstrass (also known as Durand–Kerner), described in [17] as "quite effective and increasingly popular." Convergence is global and quadratic in the limit [18]. Our implementation is basic; see [13] for algorithmic improvements to this method.

5.4. Numerical aspects and experiments. In this section we first compare our new algorithms with our approach to interpolation in [27], where we solved the linear system of interpolation conditions directly. Then we illustrate the numerical stability of the new algorithms on a test polynomial.

Compared to the direct approach of [27], we first observe an improved conditioning of the interpolation problem when a structured grid of samples is used. This improved conditioning leads to more accurate results regardless of the interpolation algorithm. In an unstructured grid, errors on the samples creep in with greater fluctuation than on a structured grid, where the ith coordinate, $i = 1, 2, \ldots, k$, is the same for all samples from a k-dimensional component. The second advantage of our new algorithms concerns time. Using divided differences to solve the linear system of N interpolation conditions requires $O(N^2)$ operations instead of $O(N^3)$ for plain Gaussian elimination or QR factorization.

The bootstrapping method for Newton interpolation we used in [29] and the basic interpolation with traces both require more samples than the number of monomials; see Table 5.1. Exploiting the Newton identities, we get the complete interpolating polynomial with an optimal number of samples. We next describe an experiment to illustrate that this exploitation is numerically stable.

We consider to interpolate the polynomial

$$p(x, y) = \sum_{0 \leq i+j \leq d} x^i y^j, \quad \text{for } d = 2, 3, \ldots, 10. \tag{5.12}$$

To this polynomial we apply our general implementation, treating $p(x, y)$ as a polynomial system. Assuming monodromy has shown the d generic points to belong to the

TABLE 5.2

Numerics on the dense polynomial with unit coefficients for degrees d from 2 up to 10, on square and triangular grids (i.e., without and with the exploitation of Newton identities) and in the last column for the linear traces. Because only magnitudes matter, 8.559E–16 is shortened to –16, and $-\infty$ stands for a zero residual. We list the accuracy of the grid (eps) and the magnitude of the highest residual after evaluation at the grid (gres) and at some test points (tres).

	Square grid			Triangular grid			Linear trace		
d	eps	gres	tres	eps	gres	tres	eps	gres	tres
2	–16	–15	–13	–15	–16	–13	–16	$-\infty$	–15
3	–15	–15	–14	–16	–15	–15	–16	$-\infty$	–15
4	–14	–14	–13	–15	–15	–14	–16	$-\infty$	–15
5	–14	–13	–12	–15	–15	–14	–16	$-\infty$	–15
6	–15	–13	–14	–16	–14	–12	–16	$-\infty$	–15
7	–15	–13	–10	–15	–14	–12	–16	$-\infty$	–15
8	–15	–12	–13	–15	–13	–11	–15	$-\infty$	–15
9	–15	–12	–14	–15	–13	–11	–16	$-\infty$	–15
10	–15	–11	–08	–14	–12	–10	–16	–16	–15

same connected component, as a test (because in this particular case there is nothing to certify) we compare three methods of certification:

1. construction of the complete trace form on a square grid;
2. construction of the complete trace form on a triangular grid, exploiting Newton identities;
3. construction of only the linear trace.

As in the general method for polynomial systems, we compute the magnitude of the highest value the interpolation polynomial returns at the grid and at some extra test points sampled from the component. These residuals would all be zero on exact data and with exact arithmetic. Due to approximate samples—accurate up to machine precision—and floating-point arithmetic, we observe errors when evaluating at the grid and at the test points; see Table 5.2.

In Table 5.2 we see an increasing loss of precision as the degree increases, for both with and without the exploitation of the Newton identities. This loss is due to the intrinsic complexity of high degree polynomials. There is no significant difference between the first two methods. From the last three columns of Table 5.2 we observe no error propagation; i.e., residuals at test points are of the same magnitude as the errors on the sample points. Compared to the complete trace form, linear traces are tolerant to approximate data and require no extra precision, at least not for the case of moderate degrees.

In Table 5.3 we list timings (on a Pentium III 800 MHz Linux machine) for the three certification methods. We see an efficiency gain with the exploitation of the Newton identities and a drastic difference when only the linear traces are computed.

In this experiment, the exploitation of the Newton identities is beneficial: fewer samples are needed and the loss of accuracy is not significantly different from the basic construction. However, for higher degrees, in situations where multiprecision arithmetic is necessary we experienced severe losses of accuracy. In particular, we applied this exploitation of Newton identities to one of the curves of degree 16 arising in the cyclic 8-roots problem (described in greater detail below). Even if the roots of the univariate polynomials were computed at full precision, the evaluation of those roots at high degree polynomials turned out to be insufficient to reach the same accuracy as without exploitation of the Newton identities. Based on these experiences, we recommend the exploitation of Newton identities only for moderate degrees.

TABLE 5.3

Timings on the dense polynomial with unit coefficients, for degrees d from 2 to 10 to construct the complete interpolator plainly on a square grid, exploiting Newton identities on a triangular grid. In the last column are timings to construct linear traces.

d	Square grid	Triangular grid	Linear traces
2	40 ms	50 ms	20 ms
3	110 ms	70 ms	30 ms
4	210 ms	170 ms	60 ms
5	420 ms	300 ms	110 ms
6	890 ms	480 ms	130 ms
7	1s 540 ms	760 ms	240 ms
8	2s 570 ms	1s 260 ms	320 ms
9	3s 730 ms	1s 800 ms	410 ms
10	4s 520 ms	3s 40 ms	600 ms

6. Applications. The algorithms have been implemented in a separate module of PHCpack [34], available online from http://www.math.uic.edu/∼jan and recently described in [30]. All reported timings are user cpu times on a Pentium III 800 Mhz Linux machine.

In the applications we consider here, the positive dimensional components are pure dimensional. Therefore, we restrict the numerical irreducible decomposition of [27] to the following three stages:

1. computation of the generic points with the embedding of [26];
2. application of monodromy [29], grouping the generic points which belong to the same irreducible component;
3. validation of the breakup predicted by monodromy by interpolation
   (a) either with the complete polynomials,
   (b) or with only the linear traces.

The methods presented in this paper only affect the last stage. We report timings for the other two stages to show the overall impact of our new approach. The experiments do not exploit the possible improvements that could result by integrating linear traces into the monodromy phase, as discussed in section 4.3.

6.1. The cyclic 8-roots and 9-roots problems. In [27] we had to limit ourselves to the *reduced* versions of those problems. With the recent advances in the decomposition algorithms we can factor the components into irreducibles *without* recourse to multiprecision arithmetic.

6.1.1. The cyclic 8-roots problem. In this section we confirm earlier results obtained in [2] by computer algebra methods. The timings for the three stages are as follows:

1. The computation of all 144 generic points on the one-dimensional components using the embedding in [26] takes 1h 12m 42s 650ms. Note that this computation also contains the calculation of the start solutions of paths leading to all isolated roots.
2. The set of 144 generic points breaks up into 8 subsets of 16 points and 8 subsets of 2 points. The monodromy breakup algorithm of [29] requires 6m 24s 930ms.
3. (a) In [29] we did the validation constructing interpolating polynomials, using standard arithmetic for the eight quadrics and using 32 decimal places for the eight curves of degree 16. This whole process took 41m 54s 780ms to complete. So stage three accounts for 35% of the total execution time.

TABLE 6.1

Numerical results of the certification of cyclic 8-roots. The columns contain the degree d, the accuracy of the samples in the grid, the largest value of the linear trace polynomial evaluated at the grid res, and the absolute value of the difference between the predicted and computed sum of the roots.

d	Accuracy of samples	Residual at grid	Difference at test pts
2	6.055E–16	1.110E–16	4.929E–14
2	4.733E–16	2.776E–16	4.308E–14
2	1.608E–15	8.882E–16	8.882E–15
2	4.143E–16	5.551E–17	5.551E–15
2	1.812E–15	1.776E–15	1.954E–14
2	1.095E–15	8.882E–16	3.642E–14
2	5.403E–16	2.220E–16	8.238E–14
2	1.815E–15	5.551E–16	2.132E–14
16	1.318E–14	6.661E–16	2.665E–14
16	6.182E–14	8.882E–16	1.199E–13
16	2.991E–14	8.882E–16	9.326E–14
16	1.239E–13	8.882E–16	9.859E–14
16	1.667E–13	8.882E–16	2.167E–13
16	8.589E–14	8.882E–16	7.372E–14
16	9.708E–15	2.220E–16	1.030E–13
16	8.168E–15	1.776E–15	5.418E–14

(b) With linear traces we need fewer samples, and expensive multiprecision arithmetic can be avoided. This interpolation takes only 27s 540ms. Compared to the time needed in stage three with the complete interpolation polynomial, this runs more than 150 times faster. The total time for the three stages reduces from about two hours to one hour and 18 minutes.

We summarize the numerical results of this calculation in Table 6.1. See [28] for the computation of the linear span of the component.

We wish to point out that the sample points are distributed widely to have a good conditioning of the interpolating polynomial. Comparing the second column with column four in Table 6.1, we observe that there is hardly any loss of accuracy for any of the roots; neither the quadrics nor the 16th degree polynomials show any significant loss.

6.1.2. The cyclic 9-roots problem. This problem has been solved with Gröbner basis methods in [4]. The timings for the three stages with our approach are as follows:

1. To compute generic points, we used the mixed-volume calculator of T.Y. Li and X. Li [15] to set up the homotopy to solve the embedding of [26]. The mixed-volume computation took 13m 4s 540ms, and the total time to compute all 20,376 paths of the embedding for two-dimensional components was 9h 11m 27s 820ms. Only 18 of these paths land on two-dimensional components: the other paths either diverge to infinity or are paths destined to lead to the isolated solutions at a later stage of the embedding technique.
2. The set of 18 generic points breaks up into 6 subsets of 3 points each. The monodromy breakup algorithm of [29] requires 2m 32s 400ms.
3. (a) In [29] we did the validation constructing interpolating polynomials, with 32 decimal places, which took 14m 56s 570ms.
   (b) The validation using only linear traces took 9s 350ms. Details are in Table 6.2.

TABLE 6.2

Numerical results of the certification of cyclic 9-roots. The columns contain the degree d, the accuracy of the samples in the grid, the largest value of the linear trace polynomial evaluated at the grid res, and the absolute value of the difference between the predicted and computed sum of the roots.

d	Accuracy of samples	Residual at grid	Difference at test pts
3	3.507E-13	0.000E+00	6.864E-14
3	5.118E-13	2.776E-17	1.456E-13
3	9.343E-13	1.388E-17	2.313E-13
3	1.529E-13	5.551E-17	3.583E-14
3	6.984E-13	0.000E+00	8.460E-14
3	1.165E-13	0.000E+00	4.080E-15

6.2. A moving Stewart–Gough platform. A generic Stewart–Gough platform mechanism has forty isolated solutions, first established by continuation [19] and later proven analytically [10, 35]. A special case of this mechanism, due to Griffis and Duffy, has a solution curve of degree forty. This means that a Griffis–Duffy platform has a one-degree-of-freedom motion, whereas a generic Stewart–Gough platform is rigid. Husty and Karger [11] pointed out this fact and also identified a more special Griffis–Duffy platform for which the solution curve breaks up into lower degree irreducible components. We treat both cases here with our numerical methods and briefly discuss some differences we found from Husty and Karger's results.

For the general Griffis–Duffy platform, which herein we call case A, the solution set consists of 12 lines and one irreducible curve of degree 28. The lines all correspond to degeneracies that do not give actual assembly configurations of the mechanism. The specialized case B also has 12 degenerate lines, but now the curve of degree 28 breaks up into lower degree irreducible components—four sextics and a quartic. The timings for the three stages in our approach are as follows:

1. To compute forty generic points using the embedding of [26] requires 52s 490ms for case A and 55s 810ms for case B.
2. For case A, the monodromy algorithm of [29] takes 33s 430ms to predict a single component of degree 28. For case B, it takes 27s 630ms for the monodromy algorithm to group the 28 generic points into five sets; four of the five have cardinality six, and one set has four points.
3. (a) For case A, the validation with Newton interpolation for the curve of degree 28 requires multiprecision (with 64 decimal places) and 812 samples (for 435 monomials), and it completes in 1h 19m 13s 110ms. For case B, using 32 decimal places in constructing the complete interpolating polynomials with divided differences takes 2m 34s 50ms.
   (b) With linear traces, case A takes 4s 750ms and case B requires 4s 320ms.

This example shows several advantages of using linear traces for validation of the monodromy breakup. Compared to the use of interpolating polynomials, the computation time is not only drastically reduced when using linear traces, but also becomes nearly identical for both cases A and B. Interpolating a degree 28 polynomial in two variables for case A is expensive and requires high precision arithmetic. In fact, because of numerical instability of traces in this case, we used the bootstrapping Newton technique as in [29] to construct the complete interpolation filter. Case B, comprised of five irreducible curves whose degrees sum to 28, is much more tractable by interpolating polynomials, but still the use of only linear traces is much superior. Table 6.3 lists numerical results of the methods, showing that the linear traces are quite stable using only double precision arithmetic. In summary, compared to inter-

TABLE 6.3

Numerical results of the certification of case A ($d = 28$) and case B ($d = 6, 6, 6, 6, 4$) for the irreducible curves occurring in moving Stewart–Gough platforms. The columns contain the accuracy of the samples in the grid, the largest value of the interpolating filter (or the linear trace) evaluated at the grid, and the residual at the test points. With linear traces, we list the difference between the predicted and computed sum at the test points.

	With complete interpolation			Using linear traces only		
d	Accuracy of samples	Residual at grid	Residual at test pts	Accuracy of samples	Residual at grid	Difference at test pts
28	1.316E–59	3.800E–37	1.107E–20	4.013E–13	1.791E–12	1.791E–12
6	3.259E–28	2.800E–27	1.020E–20	1.272E–12	2.442E–15	4.694E–13
6	5.243E–29	8.495E–28	6.416E–21	9.944E–13	1.332E–15	3.659E–13
6	1.152E–28	6.000E–30	2.502E–21	8.660E–13	2.220E–16	5.853E–13
6	4.730E–29	2.540E–28	4.936E–20	7.438E–14	2.220E–15	1.083E–11
4	4.758E–30	4.300E–31	3.357E–27	1.063E–14	2.220E–16	3.408E–14

polating polynomials, our **Certify** algorithm, based on linear traces, eliminates the large fluctuation in timings with superior numerical stability and efficiency.

We show how we can observe the propagation of roundoff errors. Compare the accuracy of the samples with the residuals at the test points in Table 6.3. For case A, the accuracy of the samples is 10^{-59}, while the residuals at the test points evaluate to 10^{-20}. During the calculation we lost about 30 decimal places. The loss in case B is more modest, between 7 and 9 decimal places, and makes the difference in exponents in the second and fourth columns of Table 6.3. With linear traces, we observe from the data in Table 6.3 that we lose at most 3 decimal places.

While the reduction in execution times may turn modest in the near future as more and faster machines will become even more widely available, the major benefit of using linear traces is that reliable results are solely obtained with standard machine arithmetic, that is, without using any multiprecision numbers. This means that errors on the coefficients of the input system that are less than the standard machine precision can be neglected and the algorithm is numerically stable.

We conclude this section with some remarks not related to numerical performance, but rather concerning the decomposition itself. The decompositions we have computed for the Griffis–Duffy platforms differ from the results obtained by Husty and Karger in [11]; one discrepancy is reconcilable, but others are not. First, for the general example, case A, we find a degree 28 curve, which at first seems to conflict with their result of a degree 20 curve. This is not, however, a contradiction, because we have analyzed the curve in the full space of rotation and translation (represented in Study coordinates). The degree falls to 20 when the curve is projected onto its rotational component only, as done by Husty and Karger. However, we also find a significant difference for the specialized case B: Husty and Karger do not list one of the five irreducible components in their analysis. Their approach, using a combination of special reasoning and computer algebra, gives some extra insight in some respects, but our automated numerical method is less subject to human error. To tackle difficult problems, it will sometimes be beneficial to use both numeric and symbolic processes. In this case, knowing about the existence of the fifth irreducible component and seeing its numerical structure, one might return to the symbolic approach to further elucidate it. (Of course, one might also pursue a completely automated symbolic approach as well, but that is another story.)

We refer to [32] for a description of this and other applications of our approach to polynomial systems in mechanism design.

7. Conclusions. In [27], we presented a numerical algorithm to decompose solution sets of polynomial systems into irreducible components of various dimensions and degrees. The main drawback of that algorithm is numerical instability on components of high degree, due to the reliance on interpolating polynomials to filter generic points into irreducible components. To deal with this difficulty, multiprecision arithmetic was used whenever high degrees were encountered, requiring a high accuracy for the input coefficients of the polynomial systems and requiring much more computer time than standard precision for each arithmetic operation. While the sequels [28] and [29] lessened the need for high precision arithmetic to some extent, it is only in this paper that we can present a numerically stable decomposition algorithm to solve the cornerstone problem in numerical algebraic geometry [33].

We summarize how standard machine arithmetic can be employed throughout the numerical irreducible decomposition algorithm. The sequence of homotopies of [26] produces generic points on every positive dimensional component, mixed with "junk": points on higher solution components. To separate those junk points from the generic points, we now propose to use the homotopy membership test of [28] instead of the filtering polynomials in [27]. Unlike with high degree polynomials, this homotopy membership test does not require multiprecision arithmetic. Also the monodromy algorithm of [29] predicts the breakup of pure dimensional components using only standard machine arithmetic. With this paper, we finally remove any need for interpolating polynomials, because linear traces suffice to certify the predicted breakup. As linear polynomials are tolerant to roundoff and efficient to interpolate, our decomposition algorithms have gained significantly in speed and robustness. Practical evidence for these claims is provided in the reports on benchmark applications.

Acknowledgment. We would like to thank the referees for their helpful remarks.

REFERENCES

[1] E. Arbarello, M. Cornalba, P.A. Griffiths, and J. Harris, *Geometry of Algebraic Curves*, Vol. I., Grundlehren Math. Wiss. 267, Springer–Verlag, New York, 1985.

[2] G. Björck and R. Fröberg, *Methods to "divide out" certain solutions from systems of algebraic equations, applied to find all cyclic* 8*-roots*, in Analysis, Algebra and Computers in Mathematical Research, Lecture Notes in Pure and Appl. Math. 156, M. Gyllenberg and L.E. Persson, eds., Dekker, New York, 1994, pp. 57–70.

[3] R.M. Corless, A. Galligo, I.S. Kotsireas, and S.M. Watt, *A geometric-numeric algorithm for factoring multivariate polynomials*, in Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation (ISSAC 2002), T. Mora, ed., ACM, New York, to appear.

[4] J.C. Faugère, *A new efficient algorithm for computing Gröbner bases (F_4)*, J. Pure Appl. Algebra, 139 (1999), pp. 61–88

[5] W. Fulton, *Intersection Theory*, Ergeb. Math. Grenzgeb. (3) 2, Springer–Verlag, Berlin, 1984.

[6] A. Galligo and D. Rupprecht, *Semi-numerical determination of irreducible branches of a reduced space curve*, in Proceedings of the 2001 International Symposium on Symbolic and Algebraic Computation (ISSAC 2001), B. Mourrain, ed., ACM, New York, 2001, pp. 137–142.

[7] M. Goresky and R. Macpherson, *Stratified Morse Theory*, Ergeb. Math. Grenzgeb. 14, Springer–Verlag, Berlin, 1988.

[8] R.C. Gunning, *Introduction to Holomorphic Functions of Several Complex Variables. Vol.* II. *Local Theory*, Wadsworth & Brooks/Cole Advanced Books & Software, Monterey, CA, 1990.

[9] R.C. Gunning and H. Rossi, *Analytic Functions of Several Complex Variables*, Prentice–Hall, Englewood Cliffs, NJ, 1965.

[10] M.L. Husty, *An algorithm for solving the direct kinematics of general Stewart-Gough platforms*, Mech. Mach. Theory, 31 (1996), pp. 365–380.

[11] M.L. Husty and A. Karger, *Self-motions of Griffis-Duffy type parallel manipulators*, in

Proceedings of the 2000 IEEE International Conference on Robotics and Automation, San Francisco, CA, CDROM, IEEE, Piscataway, NJ, 2000.

[12] E. Isaacson and H.B. Keller, *Analysis of Numerical Methods*, Dover, New York, 1994.

[13] P. Kirrinnis, *Fast numerical improvement of factors of polynomials and of partial fractions*, in Proceedings of ISSAC'98, O. Gloor, ed., ACM, New York, 1998, pp. 260–267.

[14] S.L. Kleiman, *Tangency and duality*, in Proceedings of the 1984 Vancouver Conference in Algebraic Geometry, CMS Conf. Proc. 6, J.B. Carrell, A.V. Geramita, and P. Russell, eds., AMS, Providence, RI, 1986, pp. 163–225.

[15] T.Y. Li and X. Li, *Finding mixed cells in the mixed volume computation*, Found. Comput. Math., 1 (2001), pp. 161–181.

[16] M. Mignotte and D. Ştefănescu, *Polynomials. An Algorithmic Approach*, Springer–Verlag, Singapore, 1999.

[17] V.V. Pan, *Solving a polynomial equation: Some history and recent progress*, SIAM Rev., 39 (1997), pp. 187–220.

[18] L. Pasquini and D. Trigiante, *A globally convergent method for simultaneously finding polynomial roots*, Math. Comp., 44 (1985), pp. 135–149.

[19] M. Raghavan, *The Stewart platform of general geometry has* 40 *configurations*, ASME J. Mech. Design, 115 (1993), pp. 277–282.

[20] D. Rupprecht, *Semi-numerical absolute factorization of polynomials with integer coefficients*, J. Symbolic Comput., to appear.

[21] T. Sasaki, *Approximate multivariate polynomial factorization based on zero-sum relations*, in Proceedings of the 2001 International Symposium on Symbolic and Algebraic Computation (ISSAC 2001), B. Mourrain, ed., ACM, New York, 2001, pp. 284–291.

[22] T. Sasaki, T. Saito, and T. Hilano, *Analysis of approximate factorization algorithm* I, Japan J. Indust. Appl. Math., 9 (1992), pp. 351–368.

[23] T. Sasaki and M. Sasaki, *A unified method for multivariate polynomial factorizations*, Japan J. Indust. Appl. Math., 10 (1993), pp. 21–39.

[24] T. Sasaki, M. Suzuki, M. Kolár, and M. Sasaki, *Approximate factorization of multivariate polynomials and absolute irreducibility testing*, Japan J. Indust. Appl. Math., 8 (1991), pp. 357–375.

[25] B. Shiffman and A.J. Sommese, *Vanishing Theorems on Complex Manifolds*, Prog. Math. 56, Birkhäuser, Boston, 1985.

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

[27] 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 (2001), pp. 2022–2046.

[28] A.J. Sommese, J. Verschelde, and C.W. Wampler, *Numerical irreducible decomposition using projections from points on the components*, in Symbolic Computation: Solving Equations in Algebra, Geometry, and Engineering, Contemp. Math. 286, E.L. Green, S. Hoşten, R.C. Laubenbacher, and V. Powers, eds., AMS, Providence, RI, 2001, pp. 37–51.

[29] A.J. Sommese, J. Verschelde, and C.W. Wampler, *Using monodromy to decompose solution sets of polynomial systems into irreducible components*, in Application of Algebraic Geometry to Coding Theory, Physics and Computation, C. Ciliberto, F. Hirzebruch, R. Miranda, and M. Teicher, eds., Kluwer Academic Publishers, Norwell, MA, 2001, pp. 297–315.

[30] A.J. Sommese, J. Verschelde, and C.W. Wampler, *Numerical irreducible decomposition using PHCpack*, in Mathematics and Visualization, M. Joswig and N. Takayama, eds., Springer–Verlag, to appear.

[31] A.J. Sommese, J. Verschelde, and C.W. Wampler, *A method for tracking singular paths with application to the numerical irreducible decomposition*, in Algebraic Geometry, a Volume in Memory of Paolo Francia, M.C. Beltrametti, F. Catanese, C. Ciliberto, A. Lanteri, C. Pedrini. and W. de Gruyter, eds., to appear.

[32] A.J. Sommese, J. Verschelde, and C.W. Wampler, *Advances in polynomial continuation for solving problems in kinematics*, in Proceedings of the ASME Design Engineering Technical Conference, Montreal, 2002, CD-ROM, ASME International, Fairfield, NJ, 2002.

[33] A.J. Sommese and C.W. Wampler, *Numerical algebraic geometry*, in The Mathematics of Numerical Analysis, Lectures in Appl. Math. 32, J. Renegar, M. Shub, and S. Smale, eds., AMS, Providence, RI, 1996, pp. 749–763.

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

[35] C.W. Wampler, *Forward displacement analysis of general six-in-parallel SPS (Stewart) platform manipulators using soma coordinates*, Mech. Mach. Theory, 31 (1996), pp. 331–337.