Nearest Multivariate System with Given Root Multiplicities

Scott R. Pope

Department of Mathematics
North Carolina State University
Campus Box 8205
Raleigh NC 27695, USA.

Agnes Szanto

Department of Mathematics
North Carolina State University
Campus Box 8205
Raleigh NC 27695, USA.

Abstract

We present a symbolic-numeric technique to find the closest multivariate polynomial system to a given one which has roots with prescribed multiplicity structure. Our method generalizes the "Weierstrass iteration", defined by Ruatta, to the case when the input system is not exact, i.e. when it is near to a system with multiple roots, but itself might not have multiple roots. First, using interpolation techniques, we define the "generalized Weierstrass map", a map from the set of possible roots to the set of systems which have these roots with the given multiplicity structure. Minimizing the 2-norm of this map formulates the problem as an optimization problem over all possible roots. We use Gauss-Newton iteration to compute the closest system to the input with given root multiplicity together with its roots. We give explicitly an iteration function which computes this minimum. These results extends previous results of Zhi and Wu and results of Zeng from the univariate case to the multivariate case. Finally, we give a simplified version of the iteration function analogously to the classical Weierstrass iteration, which allows a componentwise expression, and thus reduces the computational cost of each iteration. We provide numerical experiments that demonstrate the effectiveness of our method.

Key words: Approximate polynomial systems, multiple roots, multiplicity structure, Weierstrass iteration

⋆ This research was partly supported by NSF grants CCR-0306406 and CCR-0347506.

Email addresses: srpope@ncsu.edu (Scott R. Pope), aszanto@ncsu.edu (Agnes Szanto).

URLs: http://www.math.ncsu.edu/~srpope/ (Scott R. Pope), http://www.math.ncsu.edu/~aszanto/ (Agnes Szanto).

1. Introduction

The paper considers systems of multivariate equations which are near systems with multiple roots. The motivation for this work comes from the fact that the main bottleneck of the most commonly used numerical solvers, for example the Newton iteration or the homotopy continuation method, is when the solver runs into a system that is close to one with multiple roots. Since the Jacobian matrix to be inverted in the Newton iteration is close to being singular, these systems are ill-conditioned. We combine multivariate interpolation techniques and the algebraic multiplicity structure of the roots to construct an iterative method which is well defined for systems with the given multiplicity structure, and we prove that it computes the roots of the nearest system with the given multiplicity structure.

Our main tool is the multivariate interpolation method developed in Mourrain (1997) and in Mourrain and Ruatta (2002). Using interpolation we were able to generalize to the multivariate case similar results of Zhi and Wu (1998) and Zeng (2005) in the univariate case which use polynomial division. By close inspection of the interpolation method, we could express the distance of the input system from the set of systems with a given root multiplicity structure as an optimization problem over the roots. This new optimization formulation for the above distance resulted in a new iterative method which computes the roots of the nearest system with the given multiplicity structure. This approach was presented in Ruatta et al. (2004) for simple roots; in the present paper we generalize it to the case of roots with multiplicities.

We start our discussion with the univariate case. We address the following problem in the univariate case:

Problem 1. Given a polynomial $f \in \mathbb{C}[x]$, a finite set of exponents $E \subset \mathbb{N}$, the number of roots $m \in \mathbb{N}$, and the multiplicity structure $l = (l_1, \ldots, l_m) \in \mathbb{N}_{\geq 1}^m$. Find $\tilde{f} \in \mathbb{C}[x]$ and $z_1, \ldots, z_m \in \mathbb{C}$ such that $\prod_{i=1}^m (x - z_i)^{l_i}$ divides $\tilde{f}$, and $f - \tilde{f} = \sum_{\alpha \in E} c_\alpha x^\alpha$ with $\sum_{\alpha \in E} |c_\alpha|^2$ minimal.

Our first contribution in the univariate case is an explicit formula for the gradient of the norm square function, extending the results of Zhi and Wu (1998) to the $m > 1$ case. The second contribution in the univariate case is a component-wise formula for the Gauss-Newton iteration to find the optimum, given here explicitly for the first time.

Next we present the extension of our results to the multivariate case. Instead of multiplicities, we must consider a more complicated notion of root multiplicity structure. A multiple root of the multivariate system $(f_1, \ldots, f_N)$ is a root $\mathbf{z}$, along with "tangential conditions" $\mathbf{\Lambda} = \{\Lambda_1, \ldots, \Lambda_k\}$ where each Λ_i is a differential operator that vanishes on the defining polynomials at $\mathbf{z}$, i.e. $\Lambda_j(f_i)(\mathbf{z}) = 0$ for all $i = 1, \ldots N$ and $j = 1, \ldots k$. The multiplicity of the root $\mathbf{z}$ is $k = |\mathbf{\Lambda}|$.

For the method presented in this paper we assume that a description of the multiplicity structure of all or some of the roots of a nearby system to the input is known in advance, and this description is given as a set of tangential conditions as described above. We refer to the following papers on the subject of computation of the multiplicity structure of multivariate polynomial systems: the theoretical foundations in the exact case go back to the works of Macaulay (1994) and Gröbner (1970), more recently it was studied for example in Marinari et al. (1995); Stetter (2004); in the approximate case Dayton and Zeng (2005) and Wu and Zhi (2008) presented a method to compute the multiplicity structure from an approximation of a multiple root.

We consider the following problem in the multivariate case:

Problem 2. Given $N \geq 1$, $f_1, \ldots f_N \in \mathbb{C}[x_1, \ldots, x_n]$, $E_1, \ldots, E_N \subset \mathbb{N}^n$ finite sets of exponents, $m \in \mathbb{N}$ the number of roots, and the tangential conditions $\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_m$, each of them being a finite set of differential polynomials from $\mathbb{C}[\partial_{x_1} \ldots \partial_{x_n}]$. Find $\tilde{f}_1, \ldots, \tilde{f}_N \in \mathbb{C}[x_1, \ldots, x_n]$ and $\mathbf{z}_1, \ldots, \mathbf{z}_m \in \mathbb{C}^n$ such that for all $i = 1, \ldots, N$, $r = 1, \ldots, m$, $\Lambda_{r,j} \in \mathbf{\Lambda}_j$ we have $\Lambda_{r,j}(\tilde{f}_i)(\mathbf{z}_r) = 0$, moreover $f_i - \tilde{f}_i = \sum_{\alpha \in E_i} c_\alpha \mathbf{x}^\alpha$ with $\sum_{i=1}^N \sum_{\alpha \in E_i} |c_\alpha|^2$ minimal.

Our first contribution in the multivariate case is to express the distance of the input from systems with given root multiplicity structure as the least squares norm of the multivariate Weierstrass map, a map from the set of possible roots to the set of systems which have these roots with the given multiplicity structure. Secondly, we give explicit formulas for the gradient of the square norm function. Thirdly, we also give explicit formulas for the Gauss-Newton iteration of the Weierstrass map, however, in the multivariate case these expressions are not given component-wise.

Lastly, we give a simplified version of the iteration, which might be of independent interest. Analogously to the classical Weierstrass map, we use the multivariate Hermite interpolation polynomials in each Gauss-Newton iteration step to transform the Jacobian matrix to a block diagonal matrix. As a consequence, we get a simple component-wise formula for the iteration function. However, this iteration will not converge to the least squares minimum, but we do give a description of its fixed points. As our numerical experiments indicate, the simplified method computes roots with the smallest residual value $\sum_{r=1}^N \sum_{i=1}^m \sum_{t=1}^{l_i} |\Lambda_{i,t}(f_r)(\mathbf{z}_i)|^2$, compared to the non-simplified versions.

We also remark that our results have straightforward extension to a formulation of Problem 2 where $f_1, \ldots, f_N$ are analytic functions in n variables and the sets $E_1, \ldots, E_N$ are finite sets of analytic functions so that the differences $f_j - \tilde{f}_j$ lie in $\operatorname{span}_{\mathbb{C}}(E_j)$ for all $j = 1, \ldots, N$ (see Ruatta et al. (2004) for a similar extension).

Also, our results can be easily extended to the problem of finding the closest system that satisfies a much broader range of constraints. We could find the closest system that share a common root, but each polynomial has a different multiplicity for the root. We could also find the closest system which has a point that obtains specified values on each polynomial (ex. $f_1(x) = 1$ and $f_2(x) = 45$) or has specific derivative values at each point (ex. $f_1'(x) = 2$, $f_2''(x) = -1$).

Finally, we would like to note that the iteration methods presented in this paper involve the computation of interpolants, i.e. the computation of the inverse or the pseudo inverse of multivariate Vandermonde matrices. However, even in the univariate case, Vandermonde matrices are ill-conditioned, especially when one cannot choose the location of the interpolation points. For larger problems, it could very well be that this ill-conditioning leads to numerical instabilities in the generalized Weierstrass iteration. We note that even though we do not have a choice in the location of the interpolant points, but we can choose freely the basis for the support of the polynomials we interpolate, which might allow improvement in the accuracy (see for example results on using orthogonal polynomials in the univariate case in Demmel and Koev (2006)). Also, using generalizations of the Newton interpolation formula to multivariate Hermite interpolation in Sauer and Xu (1995) may improve accuracy. We would also like to remark, following the suggestions of an anonymous referee, that one could interpret these distances as condition numbers and then it also might suffice in most applications if one could just estimate the magnitude of the distances to get a reliable impression of the conditioning of the given problem.

1.1. Related Work

The literature on solving polynomial systems and systems of analytic functions near singular systems is very rich. Below we give a brief summary of results that are the most related to our method. Without trying to give a complete survey of other results tackling related problems, instead we point to papers that contain good surveys and extended bibliography on these topics.

Our method is similar in spirit to a sequence of results computing the nearest systems which have certain prescribed properties: Hitz and Kaltofen (1998) computes the nearest univariate polynomial with constrained roots, Hitz et al. (1999) computes the nearest polynomial with real root, Karmarkar and Lakshman (1996, 1998) computes the nearest pair of polynomials with non-trivial GCD, Zhi and Wu (1998) computes the nearest polynomial with a singular root, Zeng (2005) computes the nearest polynomial with several singular roots.

In the univariate case, the least squares formulation given here is an extension of the ones in Zhi and Wu (1998); Zeng (2005) to the more general Problem 1. In Zhi and Wu (1998) the authors consider the $m = 1$ case. They formulate the distance of the input to the set of monic degree d polynomials with a root of multiplicity l as a least squares problem, and they give explicit matrix formulation for the objective function and its gradient. In Zeng (2005) the author considers the case when $\sum_{i=1}^{m} l_i = \deg(f)$, and the set E is the support of f. He also formulates the problem as a least squares problem, although the objective function is not given explicitly. The minimum is computed using Gauss-Newton iteration. He also discusses the question of computing the multiplicity structure, which we do not consider in the present paper. Our subject in the univariate case can also be viewed as a special case of the approximate GCD problem (for a survey see (Grabmeier et al., 2002, Section 2.12.3)), and in particular the present approach is an extension of the works in Corless et al. (1995); Karmarkar and Lakshman (1998); Ruatta et al. (2004) where an approximate GCD is sought without multiple roots, while here we seek for a GCD which has roots with given multiplicities.

Another approach handling univariate polynomials near ones with multiple roots is the computation of zero clusters. In Hribernig and Stetter (1997); Stetter (1996) the authors compute the individual zeroes of the cluster if a nearby system with multiple roots is given: our method can be used to compute the nearby system with multiplicities. In the papers Yakoubsohn (2000); Giusti et al. (2005) a complete complexity and convergence analysis is given for computing the center of a small disc containing the cluster. These papers also contain surveys and extended bibliography of the topic. Methods that do not require an initial approximation of the clusters include Sasaki and Noda (1989); Sasaki (2003); Kaltofen and May (2003); Zeng (2005); Janovitz-Freireich et al. (2006); these methods are adaptations of the exact square-free factorization to the approximate case.

Considering the multivariate case, we are not aware of any work which considers computing the nearest multivariate system with given root multiplicity structure. In the works Ojika et al. (1983); Ojika (1987a,b); Lecerf (2002); Dayton and Zeng (2005); Leykin et al. (2006, 2007); Giusti et al. (2007) an incremental "deflation" algorithm is proposed to replace the original system which had a multiple root with another one which has the same root with multiplicity one. Lecerf (2002) shows that the resulting iterator converges quadratically to the root in an appropriate non-Archimedean metric. Recently Giusti et al. (2007) gives a complete convergence analysis for the codimension one case in terms of α-theory of the deflation method, as well as criteria as to where to stop the iteration near the cluster. This paper also contains a good survey of numerical techniques to handle roots with clusters of roots. Global techniques – techniques where all roots of a polynomial system are encountered – include the homotopy continuation method of Morgan et al. (1992); Sommese

and Wampler (1996); Sommese et al. (2002), or the approximate radical computation in Janovitz-Freireich et al. (2006, 2007), where an approximation to the center of gravity of every cluster is computed, without initial approximations.

The method presented here has the property that it can compute several roots simultaneously, but does not need to compute all the roots. Mourrain and Ruatta (2002) define a generalization of the Weierstrass iteration which computes several roots simultaneously with given multiplicity structure. As we mentioned earlier, our iteration is an adaptation of the generalized Weierstrass iteration in Mourrain and Ruatta (2002) for the case when the input might not have multiple roots but a nearby system has.

2. Notation

We will use the following notation throughout this paper:

(1) Vectors will be denoted by **bold**, and considered column vectors. Ex.

$$\mathbf{v} := (v_1, \ldots, v_n)^T \in \mathbb{C}^n.$$

The 2-norm of a vector is denoted by

$$\|\mathbf{v}\| = \|\mathbf{v}\|_2 = \sum_{i=1}^{n} |v_i|^2.$$

(2) Monomials will be expressed as a vector to a vector power. Ex. $\mathbf{x}^\alpha := \prod_{i=1}^n x_i^{\alpha_i}$ for $\alpha \in \mathbb{N}^n$.

(3) Conjugate of a complex number z will be denoted by $\overline{z}$. The adjoint of a matrix A (i.e. conjugate transpose matrices) will be denoted by A^*.

(4) A support is a set of exponents $E := \{\alpha_1, \ldots, \alpha_n\}$ where $\alpha_i \in \mathbb{N}^n$. We denote by $\mathbb{C}[x_1, \ldots, x_n]_E$ to be the set of all polynomials in $\mathbb{C}[x_1, \ldots, x_n]$ of the form $\sum_{\alpha \in E} c_\alpha \mathbf{x}^\alpha$.

(5) For a complex matrix A, we will denote the Moore-Penrose pseudoinverse by $A^\dagger$. If A has more columns than rows and maximal rank, then $A^\dagger = A^*(AA^*)^{-1}$. If A has more rows than columns and maximal rank, then $A^\dagger = (A^*A)^{-1}A^*$.

(6) If A is a matrix/vector with symbolic entries, then we will denote by $\frac{\partial A}{\partial z}$ the matrix/vector obtained by applying $\frac{\partial}{\partial z}$ to each entry of A.

3. Univariate Hermite Interpolation

We give a description of the Hermite interpolation method following the approach in Mourrain and Ruatta (2002). While the univariate case is a special case of the multivariate case which we describe later in the paper, it is presented here for better understanding.

Root multiplicity structures are given in the form $\mathbf{l} = (l_1, \ldots, l_m)$ where l_i is the multiplicity of the ith root. For example, $\mathbf{l} = (3, 1)$ represents 2 roots, the first of which has multiplicity 3 and the second is a simple root. In Zhi and Wu (1998), they consider the case of the nearest polynomial with one multiple root. In our notation, a single root of multiplicity l would be represented by (l).

In the next definition we introduce the ingredients of the univariate Hermite interpolation.

Definition 3. Let $\mathbf{z} := (z_1, \ldots, z_m) \in \mathbb{C}^m$ fixed. For a given multiplicity structure $\mathbf{l} := (l_1, \ldots, l_m)$ and a support $E := \{\alpha_1, \ldots, \alpha_t\} \subset \mathbb{N}$, we define a *generalized Vandermonde matrix*

$$V_{E,\mathbf{l}}(\mathbf{z}) := \begin{bmatrix} z_1^{\alpha_1} & \cdots & z_1^{\alpha_t} \\ \alpha_1 z_1^{\alpha_1-1} & \cdots & \alpha_t z_1^{\alpha_t-1} \\ \vdots & & \vdots \\ \frac{\alpha_1!}{(\alpha_1-l_1+1)!} z_1^{\alpha_1-l_1+1} & \cdots & \frac{\alpha_t!}{(\alpha_t-l_1+1)!} z_1^{\alpha_t-l_1+1} \\ \vdots & & \vdots \\ z_m^{\alpha_1} & \cdots & z_m^{\alpha_t} \\ \vdots & & \vdots \\ \frac{\alpha_1!}{(\alpha_1-l_m+1)!} z_m^{\alpha_1-l_m+1} & \cdots & \frac{\alpha_t!}{(\alpha_t-l_m+1)!} z_m^{\alpha_t-l_m+1} \end{bmatrix}.$$

If we denote

$$d := \sum_{i=1}^{m} l_i$$

then $V_{E,\mathbf{l}}(\mathbf{z})$ has size $d \times t$. We assume that $t \geq d$ for the rest of the paper.

We define the set

$$\Delta_{E,\mathbf{l}} := \{\mathbf{z} \in \mathbb{C}^m \mid \operatorname{rank} V_{E,\mathbf{l}}(\mathbf{z}) < d\}.$$

If it is clear from the context, we will omit $\mathbf{z}$, E and $\mathbf{l}$ from $V := V_{E,\mathbf{l}}(\mathbf{z})$ and $\Delta := \Delta_{E,\mathbf{l}}$.

Note that if $\mathbf{z} \in \mathbb{C}^m \setminus \Delta$ then the Moore-Penrose pseudoinverse of V is given by $V^\dagger = V^*(VV^*)^{-1}$ (see also in the Notation section).

We define a set of unit column vectors $\mathbf{u_{i,j}}$ as follows:

$$\mathbf{u}_{i,j} := [\underbrace{0 \cdots 0}_{l_1} \cdots \underbrace{0 \cdots \overset{j}{\widehat{1}} \cdots 0}_{l_i} \cdots \underbrace{0 \cdots 0}_{l_m}]^T$$

Each $\mathbf{u}_{i,j}$ consists of m blocks. The kth block of $\mathbf{u}_{i,j}$ has l_k rows. The ith block has a 1 in the jth position.

Define the vector of monomials $\mathbf{x}_E := (x^{\alpha_1}, \ldots, x^{\alpha_t})$.

We define the *Hermite basis polynomials* by

$$h_{i,j}(\mathbf{z}, x) := \mathbf{x}_E \, V^\dagger \mathbf{u}_{i,j}. \tag{1}$$

for all $1 \leq i \leq m$ and $1 \leq j \leq l_i$.

The next proposition is a straightforward consequence of the definition of the Hermite basis polynomials.

Proposition 4. *Given $\mathbf{z} = (z_1, \ldots, z_m) \in \mathbb{C}^m$, $\mathbf{l} = (l_1, \ldots, l_m)$, $E = (\alpha_1, \ldots, \alpha_t)$ and $V = V_{E,\mathbf{l}}(\mathbf{z})$ as in Definition 3. Assume that the matrix V has full rank. Then the Hermite basis polynomials defined in (1) satisfy:*

$$\frac{\partial^k h_{i,j}}{\partial x^k}(\mathbf{z}, z_r) = \begin{cases} 1, & \text{if } i = r \text{ and } j = k-1; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

for all $1 \leq i, r \leq m$, $1 \leq j \leq l_i$ *and* $1 \leq k \leq l_r$.

Next we define the Hermite interpolation of a polynomial $f \in \mathbb{C}[x]$ with respect to a fixed multiplicity structure $\mathbf{l}$ and support E.

Definition 5. Let $\mathbf{l} = (l_1, \ldots, l_m)$, $E = (\alpha_1, \ldots, \alpha_t)$ and $\mathbf{z} = (z_1, \ldots, z_m) \in \mathbb{C}^m \setminus \Delta_{E,\mathbf{l}}$ be given as in Definition 3. Let $h_{i,j}(\mathbf{z}, x)$ be as in (1). For a given $f \in \mathbb{C}[x]$ we define the *Hermite interpolation polynomial*

$$F(\mathbf{z}, x) := \sum_{i=1}^{m} \sum_{j=1}^{l_i} f^{(j-1)}(z_i) h_{i,j}(\mathbf{z}, x).$$

The following proposition gives the main properties of the Hermite interpolation polynomials.

Proposition 6. *With* f, m, $\mathbf{l}$, E, $\mathbf{z}$, *and* $F(\mathbf{z}, x)$ *defined as above, the following properties hold:*
(1) $F^{(j)}(\mathbf{z}, z_i) = f^{(j)}(z_i)$ *for all* $1 \leq i \leq m$ *and* $0 \leq j \leq l_i - 1$.
(2) $F(\mathbf{z}, x)$ *is a polynomial with support* E.
(3) $\|F(\mathbf{z}, x)\|_2$ *is minimal among all polynomials with properties 1 and 2.*
(4) A polynomial having properties 1, 2 and 3 is uniquely determined and is equal to $F(\mathbf{z}, x)$.

Proof. 1. Let $\mathbf{F} \in \mathbb{C}^t$ be the coefficient vector of F in the monomial basis $\{x^{\alpha_1}, \ldots, x^{\alpha_t}\}$, i.e.

$$F = \mathbf{x}_E \mathbf{F}, \tag{3}$$

where $\mathbf{x}_E = (x^{\alpha_1}, \ldots, x^{\alpha_t})$ as before. Define also the evaluation vector $\mathbf{f}$ for f, namely

$$\mathbf{f} := (f(z_1), \ldots, f^{(l_1-1)}(z_1), f(z_2), \ldots, f^{(l_m-1)}(z_m))^T. \tag{4}$$

It suffices to note that $V_{E,\mathbf{l}}(\mathbf{z})\mathbf{F} = \mathbf{f}$, so F has the desired property.

2. Since each $h_{i,j}$ has support E, F also has support E.

3. Note that $\mathbf{F} = V^{\dagger}\mathbf{f}$, and the claim is a basic property of the Moore-Penrose pseudoinverse (see for example Meyer (2000)).

4. The uniqueness of least-squares solutions is proved for example in Meyer (2000). □

4. Univariate Weierstrass Map

In this section, we introduce the univariate Weierstrass map, originally defined in Mourrain and Ruatta (2002), using the Hermite interpolation polynomials defined in the previous section. We prove that the minimal 2-norm of the Weierstrass map equals the distance of the input polynomial from the set of polynomials with the given root multiplicity structure. As we mentioned in the introduction, the Weierstrass map considered here is a generalization of the map defined in Zeng (2005) in that we allow for the perturbation of coefficients corresponding to an arbitrary set of monomials, while in Zeng (2005) only perturbation of all the non-leading coefficients of f is allowed.

Definition 7. Let f, m, $\mathbf{l}$, E, F, and Δ be as in Definition 5. We define the *univariate Weierstrass map* as

$$\begin{aligned} \mathcal{W} : \mathbb{C}^m \setminus \Delta &\to \mathbb{C}[x]_E \\ \mathbf{z} &\mapsto F(\mathbf{z}, x) \end{aligned}$$

The following proposition gives the main properties of the Weierstrass map.

Proposition 8. *Let f, m, $\mathbf{l}$, E, F, and Δ be as above. The univariate Weierstrass map $\mathcal{W}$ defined in Definition 7 has the following properties:*

(1) $\mathcal{W}(\mathbf{z}) = \mathbf{0}$ for some $\mathbf{z} = (z_1, \ldots, z_m) \in \mathbb{C}^m \setminus \Delta$ if and only if $z_1, \ldots, z_m$ are roots of f with multiplicities $l_1, \ldots, l_m$, respectively.

(2) $\|\mathcal{W}(\mathbf{z})\|_2^2 = \mathbf{f}^* M^{-1} \mathbf{f}$ where $\|\mathcal{W}(\mathbf{z})\|_2$ is the 2-norm of the coefficient vector $\mathbf{F}$ of F, $M := VV^*$ and $\mathbf{f}$ is defined as in (4) by

$$\mathbf{f} := (f(z_1), \ldots, f^{(l_1-1)}(z_1), f(z_2), \ldots, f^{(l_m-1)}(z_m))^T \quad \in \mathbb{C}^d.$$

(3) If $\min_{\mathbf{z} \in \mathbb{C}^m \setminus \Delta} \|\mathcal{W}(\mathbf{z})\|_2$ exists then it is equal to the distance of f from the set of polynomials $\tilde{f}$ such that $\tilde{f}$ has the multiplicity structure $\mathbf{l}$ and is obtained by perturbation of the coefficients of f corresponding to the support E.

Proof. We prove this in a more general setting in Proposition 26 below. □

We follow the approach of Zhi and Wu Zhi and Wu (1998) and express the roots of the gradient of the function

$$\|\mathcal{W}\|^2 : \mathbb{C}^m - \Delta \to \mathbb{R}.$$

If $z_i = a_i + ib_i$ for $i = 1, \ldots, m$, then $\|\mathcal{W}\|^2$ can be considered as a real function $\|\mathcal{W}\|^2 : (\mathbb{R}^2)^m \setminus \Delta \to \mathbb{R}$, and its gradient is a $1 \times 2m$ vector. However, since one can formally define

$$\frac{\partial \|\mathcal{W}\|^2}{\partial z_i} = \frac{1}{2}\left(\frac{\partial \|\mathcal{W}\|^2}{\partial a_i} - i\frac{\partial \|\mathcal{W}\|^2}{\partial b_i}\right),$$
$$\frac{\partial \|\mathcal{W}\|^2}{\partial \bar{z}_i} = \frac{1}{2}\left(\frac{\partial \|\mathcal{W}\|^2}{\partial a_i} + i\frac{\partial \|\mathcal{W}\|^2}{\partial b_i}\right)$$

we can also consider the roots of $\frac{\partial \|\mathcal{W}\|^2}{\partial z_i}$ for $i = 1, \ldots, m$.

The next theorem and corollaries are needed to give explicit formulas for the gradient of $\|\mathcal{W}\|^2$, as well as for the Jacobian matrix of the Weierstrass map, which we will use later.

Theorem 9. *Let f, V and F be as in Definitions 3 and 5. Let $\mathbf{F}$ be the coefficient vector of F, and $\mathbf{f}$ be the evaluation vector of f as in (3) and (4). If V has maximal rank, then*

$$\frac{\partial \mathbf{F}}{\partial z_i} = V^\dagger \left(\frac{\partial \mathbf{f}}{\partial z_i} - \frac{\partial V}{\partial z_i}\mathbf{F}\right)$$

Proof. This follows immediately from the following lemma and the fact that $\frac{\partial V^*}{\partial z_i} = 0$ since V is analytic in z_i. □

Lemma 10. *Let f, V, F, $\mathbf{F}$ and $\mathbf{f}$ be as above. If V is of maximal rank, then*

$$\frac{\partial \mathbf{F}}{\partial z_i} = V^\dagger \left(\frac{\partial \mathbf{f}}{\partial z_i} - \frac{\partial V}{\partial z_i}\mathbf{F}\right) + \left(I - V^\dagger V\right)\left(\frac{\partial V^*}{\partial z_i}\left(VV^*\right)^{-1}\mathbf{f}\right). \tag{5}$$

Proof. By definition, $\mathbf{F}$ is the least squares solution to

$$V\mathbf{F} = \mathbf{f}. \tag{6}$$

We can solve this system using the Moore-Penrose pseudoinverse of V, namely $V^\dagger = V^* (VV^*)^{-1}$. So equation (6) is actually solved in two steps. First we find $\mathbf{g}$ such that

$$VV^*\mathbf{g} = \mathbf{f} \tag{7}$$

then we compute $\mathbf{F}$ as

$$\mathbf{F} = V^*\mathbf{g}. \tag{8}$$

From equation (8) we have

$$\frac{\partial \mathbf{F}}{\partial z_i} = \left(\frac{\partial V^*}{\partial z_i}\right)\mathbf{g} + V^*\left(\frac{\partial \mathbf{g}}{\partial z_i}\right) \tag{9}$$

From equation (7) we have

$$\frac{\partial \mathbf{g}}{\partial z_i} = (VV^*)^{-1}\left(\frac{\partial \mathbf{f}}{\partial z_i} - \left(\frac{\partial V}{\partial z_i}\right)V^*\mathbf{g} - V\left(\frac{\partial V^*}{\partial z_i}\right)\mathbf{g}\right) \tag{10}$$

Combining equations (10) and (9) and simplifying gives (5). □

Corollary 11. Let f, E, m, $\mathbf{l}$, d, $\mathbf{z}$ $h_{i,j}$ and F be as in Definitions 3 and 5. Let π be the projection

$$\pi : \mathbb{C}[x]_E \to \mathrm{span}_{\mathbb{C}}\{h_{i,j}\}_{i=1\,j=1}^{m\quad l_i}$$

Then

$$\pi\frac{\partial F}{\partial z_i} = \left(f^{(l_i)}(z_i) - F^{(l_i)}(\mathbf{z}, z_i)\right)h_{i,l_i}(\mathbf{z}, x). \tag{11}$$

Proof. By Theorem 9 we have

$$V\frac{\partial \mathbf{F}}{\partial z_i} = \frac{\partial \mathbf{f}}{\partial z_i} - \frac{\partial V}{\partial z_i}\mathbf{F}.$$

Because of property 1 of Proposition (6) we get that

$$\frac{\partial \mathbf{f}}{\partial z_i} - \frac{\partial V}{\partial z_i}\mathbf{F} = \left(f^{(l_i)}(z_i) - F^{(l_i)}(\mathbf{z}, z_i)\right)\mathbf{u}_{i,l_i},$$

where $\mathbf{u}_{i,j}$ as defined in Definition 3. Equivalently,

$$\pi\frac{\partial F}{\partial z_i} = \left(f^{(l_i)}(z_i) - F^{(l_i)}(\mathbf{z}, z_i)\right)h_{i,l_i}(\mathbf{z}, x).$$

□

The next proposition extends the results of (Zhi and Wu, 1998, Theorem 1) to the case when $m > 1$. Before stating the proposition we need some definitions.

Definition 12. Let $\mathbf{z} = (z_1, \ldots, z_m)$, $\mathbf{l} = (l_1, \ldots, l_m)$, $E = \{\alpha_1, \ldots, \alpha_t\}$, f, V, F, $\mathbf{F}$ and $\mathbf{f}$ be as above. For $i = 1, \ldots, m$ we define

- $\hat{V}_i$ to be the matrix obtained from V by deleting the row $\frac{\partial^{l_i-1}}{\partial z_i^{l_i-1}}(z_i^{\alpha_1}, \ldots, z_i^{\alpha_t})$, i.e. $\hat{V}_i$ is the Vandermonde matrix for the multiplicity structure $(l_1, \ldots, l_i - 1, \ldots, l_m)$;
- $\hat{\mathbf{f}}_i$ to be the vector obtained from $\mathbf{f}$ by deleting the entry $f^{(l_i-1)}(z_i)$;
- $\hat{M}_i := \hat{V}_i\hat{V}_i^*$;

- $\hat{F}_i(\mathbf{z}, x)$ to be the polynomial with coefficient vector $\hat{\mathbf{F}}_i := \hat{V}_i^{\dagger}\hat{\mathbf{f}}_i$, i.e. $\hat{F}_i$ interpolates f in $\mathbf{z}$ with respect to the multiplicity structure $(l_1, \ldots, l_i - 1, \ldots, l_m)$.

Proposition 13. *Using the notation of Definition 12, for each $i = 1, \ldots, m$ the partial derivative $\frac{\partial \|\mathcal{W}\|^2}{\partial z_i}$ is equal to*

$$\frac{\partial \|\mathcal{W}\|^2}{\partial z_i} = \frac{\det \hat{M}_i}{\det M} \left(f^{(l_i)}(z_i) - F^{(l_i)}(z_i) \right) \left(\overline{f^{(l_i-1)}(z_i) - \hat{F}_i^{(l_i-1)}(z_i)} \right). \tag{12}$$

Proof. The proof is similar to the one in (Zhi and Wu, 1998, Theorem 1), so we will skip some of the details. By Proposition 8 we have that $\|\mathcal{W}\|^2 = \mathbf{f}^* M^{-1} \mathbf{f}$. Let D be the $d \times m$ matrix with entries defined by

$$D_{j,i} = \begin{cases} f^{(l_i)}(z_i) - F^{(l_i)}(\mathbf{z}, z_i) & \text{if } j = \sum_{t=1}^{i} l_t \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

and let $D_{*,i}$ be the i-th column of D. Let $\mathbf{g} = M^{-1}\mathbf{f}$ be defined as in (7) in the proof of Lemma 10. We have

$$\frac{\partial \|\mathcal{W}\|^2}{\partial z_i} = \frac{\partial \mathbf{f}^* M^{-1} \mathbf{f}}{\partial z_i} = \frac{\partial \mathbf{f}^* \mathbf{g}}{\partial z_i}. \tag{14}$$

Since $\mathbf{f}$ is analytic, $\frac{\partial \mathbf{f}^*}{\partial z_i} = 0$, and by (10) we have that (14) is equal to

$$\frac{\partial \|\mathcal{W}\|^2}{\partial z_i} = \mathbf{f}^* \frac{\partial \mathbf{g}}{\partial z_i} = \mathbf{f}^* (VV^*)^{-1} \left(\frac{\partial \mathbf{f}}{\partial z_i} - \frac{\partial V}{\partial z_i} \mathbf{F} \right) \tag{15}$$

or equivalently

$$\frac{\partial\, \mathbf{f}^* M^{-1} \mathbf{f}}{\partial z_i} = \mathbf{f}^* M^{-1} D_{*,i}.$$

Since the only non-zero entry in $D_{*,i}$ is in row $j_i := \sum_{r=1}^{i} l_r$ and is equal to $f^{(l_i)}(z_i) - F^{(l_i)}(z_i)$, it suffices to show that the j_i-th entry in $\mathbf{f}^* M^{-1}$ is equal to the conjugate of

$$\frac{\det \hat{M}_i}{\det M} (f^{(l_i-1)}(z_i) - \hat{F}_i^{(l_i-1)}(z_i)).$$

This follows from the repeated application of Cramer's rule to get

$$\mathbf{u}_{i,l_i}^T M^{-1} \mathbf{f} = \pm \frac{1}{\det M} \left(f^{l_i}(z_i) \det(\hat{M}_i) - (m_{j_i} \hat{M}_i^{-1} \hat{\mathbf{f}}_i) \det(\hat{M}_i) \right),$$

and from the the fact that

$$\hat{F}_i^{(l_i-1)}(z_i) = v_{j_i} \hat{V}_i^* \hat{M}_i^{-1} \hat{\mathbf{f}}_i = m_{j_i} \hat{M}_i^{-1} \hat{\mathbf{f}}_i$$

where v_{j_i} is the j_i-th row of V and m_{j_i} is the j_i-th row of $\hat{M}_i$. □

Remark 14. If we assume that V has full rank at $\mathbf{z}$, then the Jacobian of $\mathcal{W}$ has deficient rank at $\mathbf{z}$ if and only if $f^{(l_i)}(z_i) - F^{(l_i)}(z_i) = 0$ for some $i = 1, \ldots, m$. This means that the common roots of the system $\left\{ \frac{\partial \|\mathcal{W}\|^2}{\partial z_i} = 0 \right\}_{i=1}^{m}$ that we are going to find using Gauss-Newton method are the common roots of $\left\{ f^{(l_i-1)}(z_i) - \hat{F}_i^{(l_i-1)}(z_i) = 0 \right\}_{i=1}^{m}$.

Note that if $d = |E|$ then the condition $\mathbf{z} \in \mathbb{C}^m \setminus \Delta$ implies that $V = V_{E,\mathbf{l}}(\mathbf{z})$ is a square invertible matrix. In this case the Weierstrass map is a complex analytic function, and we can use the approach of Zeng (2005) applying the Gauss-Newton iteration to get an algorithm which computes a local minimum of the Weierstrass map.

In the general case, when $|E| > d$ and $\mathbf{z} \in \mathbb{C}^m \setminus \Delta$, then V has more columns than rows and has full rank, but the Weierstrass map is not a complex analytic function. However, next we make an observation that is useful in determining the local convergence property of a Gauss-Newton map applied to $\mathcal{W}$ in the general case.

Let J be the Jacobian of $\mathcal{W}$ and let $J_{*,i}$ be the i-th column of J. Then using Theorem 9 and (15) we have that

$$J_{*,i}^* \mathcal{W} = \left(\frac{\partial \mathbf{f}}{\partial z_i} - \frac{\partial V}{\partial z_i} \mathbf{F} \right)^* (VV^*)^{-1} V\mathbf{F} \tag{16}$$

$$= \left(\frac{\partial \mathbf{f}}{\partial z_i} - \frac{\partial V}{\partial z_i} \mathbf{F} \right)^* (VV^*)^{-1} \mathbf{f} \tag{17}$$

$$= \left(\frac{\partial \|\mathcal{W}\|^2}{\partial z_i} \right)^* . \tag{18}$$

If $\tilde{\mathbf{z}}$ is a local minimum of $\|\mathcal{W}(\mathbf{z})\|^2$ then $\frac{\partial \|\mathcal{W}\|^2}{\partial z_i}(\tilde{\mathbf{z}}) = 0$ for $i = 1, \ldots, m$. Therefore $J(\tilde{\mathbf{z}})^* \mathcal{W}(\tilde{\mathbf{z}}) = 0$ (see also (Zeng, 2005, Lemma 2.5)). The Gauss-Newton iteration for $\mathcal{W}$ is

$$\mathbf{z}^{new} = \mathbf{z} - J(\mathbf{z})^\dagger \mathcal{W}(\mathbf{z}) \tag{19}$$

and locally converges to $\tilde{\mathbf{z}}$ satisfying $J(\tilde{\mathbf{z}})^* \mathcal{W}(\tilde{\mathbf{z}}) = 0$, assuming that $J(\tilde{\mathbf{z}})$ has full rank (see Dedieu and Shub (2000)). Note that by Proposition 13 we have that the entries of $J(\tilde{\mathbf{z}})^* \mathcal{W}(\tilde{\mathbf{z}})$ are equal to the conjugate of the expression in (12), and if $J(\tilde{\mathbf{z}})$ has full rank then $f^{(l_i)}(\tilde{z}_i) - F^{(l_i)}(\tilde{z}_i) \neq 0$ for all $i = 1, \ldots, m$.

The next theorem gives a component-wise expression for the Gauss-Newton iteration in (19).

Theorem 15. *Let f, m, $\mathbf{l} = (l_1, \ldots, l_m)$, Δ, F, M and $\mathcal{W}$ be as above. For $\mathbf{z} = (z_1, \ldots, z_m) \in C^m \setminus \Delta$ assume that the Jacobian J of $\mathcal{W}$ at $\mathbf{z}$ has full rank. Then the Gauss-Newton iteration for $\mathcal{W}$ has the following component-wise expression:*

$$z_j^{new} = z_j - \frac{\det(N_j)}{\det(M)\det(N)(f^{(l_j)}(z_j) - F^{(l_j)}(z_j))}$$

where N is the $m \times m$ submatrix of M^{-1} with rows and columns indexed by $\sum_{s=1}^{i} l_s$ for $i = 1, \ldots, m$, and N_j is the matrix obtained from N by replacing its j-th column by the vector

$$P := \left[\det(\hat{M}_i) \left(f^{(l_i - 1)}(z_i) - \hat{F}_i^{(l_i - 1)}(z_i) \right) \right]_{i=1,\ldots,m}^T .$$

Here $\hat{M}_i$ and $\hat{F}_i$ are defined in Definition 12.

Proof. Using the definition of M and F and Corollary 11 and also the matrix D defined in equation (13) we get that (19) is equal to

$$\mathbf{z}^{new} = \mathbf{z} - (D^* M^{-1} D)^{-1} D^* M^{-1} \mathbf{f}$$

where D is the matrix defined in (13) and $\mathbf{f}$ was defined (4). Let $\tilde{D}$ be the $m \times m$ diagonal matrix consisting of the non-zero rows of D. Then we have

$$D^* M^{-1} D = \tilde{D}^* N \tilde{D}$$

and the condition that the Jacobian $J(\mathbf{z})$ has full rank is equivalent to both $\tilde{D}$ and N being invertible. Therefore,

$$(D^* M^{-1} D)^{-1} D^* M^{-1} \mathbf{f} = \frac{1}{\det(M)} \tilde{D}^{-1} N^{-1} P. \tag{20}$$

Here we used the fact that the entry of $M^{-1}\mathbf{f}$ indexed by $j_i := \sum_{s=1}^{i} l_s$ is equal to $\frac{\det(\hat{M}_i)}{\det(M)}(f^{(l_i-1)}(z_i) - \hat{F}_i^{(l_i-1)}(z_i))$ (see the proof of Proposition 13). The claim follows from Cramer's rule applied for the j-th entry of $\tilde{D}^{-1} N^{-1} P$. □

Remark 16. Because of the multiple determinate calculations in the component-wise formula, it is actually more efficient to compute the iteration function using Theorem 9, or to use (20). If we use the assumption that the number of support elements and constraints are equal, Corollary 11 gives us an even more efficient way to construct the Jacobian.

5. Multivariate Interpolation

In the multivariate case the multiplicity structure of roots of a system can be described by the vanishing of certain partial derivatives of the defining polynomials in the given roots. A theory of the multiplicity structure in the language of dual algebras can be found in Marinari et al. (1995); Stetter (2004); Mourrain and Ruatta (2002). Here we adopt the notation in Mourrain and Ruatta (2002).

Notation 17. Let $\mathbf{x} = (x_1, \ldots, x_n)$ be n variables.

- We use Λ or $\Lambda(\partial)$ to denote a polynomial in the partial derivatives ∂_{x_i}, i.e. $\Lambda \in \mathbb{C}[\partial_{x_1} \ldots \partial_{x_n}]$.
- For $\Lambda \in \mathbb{C}[\partial_{x_1} \ldots \partial_{x_n}]$, $p \in \mathbb{C}[x_1, \ldots, x_n]$ and $\mathbf{z} \in \mathbb{C}^n$, we use $\Lambda(p)(\mathbf{z})$ to denote Λ applied to the polynomial p, and the result evaluated at $\mathbf{z}$.

In the first definition we give the main ingredients of a multivariate Hermite interpolation.

Definition 18. Given

- Support $E = \{\alpha_1, \ldots, \alpha_t\} \subset \mathbb{N}^n$;
- $\vec{\mathbf{z}} = (\mathbf{z}_1, \ldots, \mathbf{z}_m) \in (\mathbb{C}^n)^m$;
- The *tangential conditions* $\mathbf{\Lambda} = (\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_m)$, where each $\mathbf{\Lambda}_i = \{\Lambda_{i,1}, \ldots, \Lambda_{i,l_i}\} \subset \mathbb{C}[\partial_{x_1} \ldots \partial_{x_n}]$ has cardinality l_i. We denote $d := \sum_{i=1}^{m} l_i$.

The generalized Vandermonde matrix corresponding to E and $\mathbf{\Lambda}$ is the following $d \times t$ matrix:

$$V_{E,\mathbf{\Lambda}}(\vec{\mathbf{z}}) := \begin{bmatrix} \Lambda_{1,1}(\mathbf{x}^{\alpha_1})(\mathbf{z}_1) & \cdots & \Lambda_{1,1}(\mathbf{x}^{\alpha_t})(\mathbf{z}_1) \\ \vdots & & \vdots \\ \Lambda_{1,l_1}(\mathbf{x}^{\alpha_1})(\mathbf{z}_1) & \cdots & \Lambda_{1,l_1}(\mathbf{x}^{\alpha_t})(\mathbf{z}_1) \\ \vdots & & \vdots \\ \Lambda_{m,1}(\mathbf{x}^{\alpha_1})(\mathbf{z}_m) & \cdots & \Lambda_{m,1}(\mathbf{x}^{\alpha_t})(\mathbf{z}_m) \\ \vdots & & \vdots \\ \Lambda_{m,l_m}(\mathbf{x}^{\alpha_1})(\mathbf{z}_m) & \cdots & \Lambda_{m,l_m}(\mathbf{x}^{\alpha_t})(z_m) \end{bmatrix}.$$

We assume that $t \geq d$ and define the set

$$\Delta_{E,\mathbf{\Lambda}} := \{\vec{\mathbf{z}} \in (\mathbb{C}^n)^m \ : \ \operatorname{rank} V_{E,\mathbf{\Lambda}}(\vec{\mathbf{z}}) < d\}.$$

We may omit E, $\mathbf{\Lambda}$ and $\vec{\mathbf{z}}$ from the notation of $\Delta := \Delta_{E,\mathbf{\Lambda}}$ and $V := V_{E,\mathbf{\Lambda}}(\vec{\mathbf{z}})$.

We define $\mathbf{u}_{i,j}$ to be the unit vector associated with $\Lambda_{i,j}$ by

$$\mathbf{u}_{i,j} := [\underbrace{0 \cdots 0}_{l_1} \ \cdots \ \underbrace{0 \cdots \overset{j}{\widehat{1}} \cdots 0}_{l_i} \ \cdots \ \underbrace{0 \cdots 0}_{l_m}]^T.$$

For fixed $\vec{\mathbf{z}} \in (\mathbb{C}^n)^m \setminus \Delta$ the multivariate *Hermite basis polynomials* are defined by

$$h_{i,j}(\vec{\mathbf{z}}, \mathbf{x}) := \vec{\mathbf{x}}_E V^{\dagger} \mathbf{u}_{i,j} \tag{21}$$

for $i = 1, \ldots, m$, $j = 1, \ldots, l_i$, and $\vec{\mathbf{x}}_E := (\mathbf{x}^{\alpha_1}, \ldots, \mathbf{x}^{\alpha_t})$.

Remark 19. See for example Möller and Sauer (2000) for an investigation on what conditions on the support elements in E ensure that V has maximal rank.

The next proposition is a straightforward generalization of the univariate case, so we omit the proof.

Proposition 20. *Let E, $\mathbf{\Lambda}$, $\vec{\mathbf{z}} \in (\mathbb{C}^n)^m \setminus \Delta$ and $h_{i,j}$ be as in definition 18. Then we have*

- $h_{i,j} \in \mathbb{C}[x_1, \ldots, x_t]_E$.
- $$\Lambda_{r,s}(h_{i,j})(\mathbf{z}_r) = \begin{cases} 1, & \text{if } i = r \text{ and } j = s; \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

for all $i, r = 1, \ldots, m$, $j = 1, \ldots l_i$ and $\Lambda_{r,s} \in \mathbf{\Lambda}_r$.

Next we define the Hermite interpolation of a polynomial $f \in \mathbb{C}[\mathbf{x}]$.

Definition 21. Fix $\vec{\mathbf{z}} = (\mathbf{z}_1, \ldots, \mathbf{z}_m) \in (\mathbb{C}^n)^m \setminus \Delta$, $f \in \mathbb{C}[\mathbf{x}]$, support $E \subset \mathbb{N}^n$ and multiplicity structure $\mathbf{\Lambda} = (\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_m)$. We define the multivariate *Hermite interpolation polynomial*

$$F(\vec{\mathbf{z}}, \mathbf{x}) := \sum_{r=1}^{m} \sum_{s=1}^{l_r} \Lambda_{r,s}(f)(\mathbf{z}_r)\, h_{r,s}(\vec{\mathbf{z}}, \mathbf{x}) \tag{23}$$

where $\Lambda_{r,s} \in \mathbf{\Lambda}_r$ and $h_{r,s}$ was defined in Definition 18.

The next proposition is a straightforward consequence of the definition, so we omit the proof.

Proposition 22. *Let $\vec{\mathbf{z}}$, f, E, $\mathbf{\Lambda}$ and F be as in Definition 21. Then*
(1) $F \in \mathbb{C}[\mathbf{x}]_E$;
(2) $\Lambda_{r,s}(f-F)(\mathbf{z}_r) = 0$ for all $r = 1, \ldots, m$ and $\Lambda_{r,s} \in \mathbf{\Lambda}_r$;
(3) F is the minimal 2-norm polynomial satisfying 1 and 2.
(4) $\|F\|_2^2 = \mathbf{f}^* M^{-1} \mathbf{f}$ where $M := VV^*$ and

$$\mathbf{f} = (\Lambda_{1,1}(f)(\mathbf{z}_1), \ldots, \Lambda_{m,l_m}(f)(\mathbf{z}_m))^T \in \mathbb{C}^d. \tag{24}$$

We finish this section with a generalization of Theorem (9) and Corollary (11) to the multivariate case. The proof is the same as the univariate case, and we omit it.

Theorem 23. *Let f, V and $F(\vec{\mathbf{z}}, \mathbf{x})$ be as in Definition 21. Let $\mathbf{F}$ be the coefficient vector of F, and $\mathbf{f}$ be the evaluation vector of f defined in (24). If V is of maximal rank then*

$$\frac{\partial \mathbf{F}}{\partial z_{i,j}} = V^\dagger \left(\frac{\partial \mathbf{f}}{\partial z_{i,j}} - \frac{\partial V}{\partial z_{i,j}} \mathbf{F} \right).$$

Corollary 24. Let f, $\mathbf{\Lambda}$, E and $F(\vec{\mathbf{z}}, \mathbf{x})$ be as in Definition 21. Let $\{h_{i,s}\}$ be the Hermite polynomials defined in Definition 18 and define the projection

$$\pi \ : \ \mathbb{C}[\mathbf{x}]_E \to \operatorname{span}_{\mathbb{C}} \{h_{i,j}\}_{i=1 \, j=1}^{m \ \ l_i}.$$

Denote $\vec{\mathbf{z}} = (\mathbf{z}_1, \ldots, \mathbf{z}_m)$ and $\mathbf{z}_i = (z_{i,1}, \ldots, z_{i,n})$. Then

$$\pi \frac{\partial F(\vec{\mathbf{z}}, \mathbf{x})}{\partial z_{i,j}} = \sum_{s=1}^{l_i} (\partial_{x_j} \Lambda_{i,s})(f-F)(\mathbf{z}_i) \ h_{i,s}(\vec{\mathbf{z}}, \mathbf{x})$$

where $\partial_{x_j} \Lambda_{i,s}$ is the multiple of ∂_{x_j} and $\Lambda_{i,s}$ in $\mathbb{C}[\partial_{x_1} \ldots \partial_{x_n}]$.

6. Multivariate Weierstrass Map

In this section we generalize the univariate Weierstrass map to the case when we are given $N \geq 1$ polynomials $f_1, \ldots, f_N \in \mathbb{C}[x_1, \ldots, x_n]$. Together with the polynomials we are also given N sets of supports $E_1, \ldots, E_N$. Furthermore, we also have m systems of tangential conditions $\mathbf{\Lambda} = \{\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_m\}$, one for each desired common root, with cardinality $|\mathbf{\Lambda}_i| =: l_i$ as in Definition 18.

For the rest of the paper we assume that

$$d < \min_i(|E_i|),$$

where $d := \sum_{i=1}^m l_i$ as in Definition 18. This implies that we have to solve an under-constrained system of equations in the interpolation problem, so it makes sense to talk about the nearest system which satisfies the tangential conditions.

Definition 25. Let $f_1, \ldots, f_N$, $\mathbf{\Lambda}$, and $E_1, \ldots, E_N$ be as above. We define the multivariate Weierstrass map as

$$\begin{aligned}\mathcal{W} : (\mathbb{C}^n)^m \setminus \Delta &\to \bigoplus_{i=1}^{N} \mathbb{C}[\mathbf{x}]_{E_i} \\ \vec{\mathbf{z}} &\mapsto (F_1, \ldots, F_N)\end{aligned}$$

where F_i is the Hermite interpolation polynomial corresponding to f_i, $\mathbf{\Lambda}$, E_i and $\vec{\mathbf{z}}$, as defined in Definition 21. Here

$$\Delta := \bigcup_{i=1}^{N} \Delta_{E_i,\mathbf{\Lambda}}$$

using the notation of Definition 18.

The next proposition summarizes the main properties of the Weierstrass map.

Proposition 26. *The Weierstrass map $\mathcal{W}$, as defined in Definition 25, has the following properties:*

(1) $\mathcal{W}(\vec{\mathbf{z}}) = \mathbf{0}$ for some $\vec{\mathbf{z}} = (\mathbf{z}_1, \ldots, \mathbf{z}_m) \in (\mathbb{C}^t)^m \setminus \Delta$ if and only if

$$\Lambda_{j,k}(f_i)(\mathbf{z}_j) = 0 \quad \text{for all} \quad 1 \le i \le N,\ 1 \le j \le m,\ 1 \le k \le l_j.$$

(2)

$$\|W(\vec{\mathbf{z}})\|_2^2 = \sum_{i=1}^{N} \mathbf{f_i}^* M_i^{-1} \mathbf{f_i}$$

where $\mathbf{f}_i$ is the evaluation vector of f_i and $M_i := V_i V_i^*$ for $V_i := V_{E_i,\mathbf{\Lambda}}(\vec{\mathbf{z}})$.

(3) If the minimum

$$\min_{\vec{\mathbf{z}} \in (\mathbb{C}^n)^m \setminus \Delta} \|\mathcal{W}(\vec{\mathbf{z}})\|_2^2$$

exists and is reached in $(\mathbb{C}^n)^m \setminus \Delta$ then it is equal to the distance of $(f_1, \ldots, f_N)$ from the set of systems $(\tilde{f}_1, \ldots \tilde{f}_N)$ such that for some $\vec{\zeta} \in (\mathbb{C}^n)^m \setminus \Delta$ each $\tilde{f}_i$ satisfies the tangential conditions $\mathbf{\Lambda}$, and is obtained by a perturbation of the coefficients of f_i corresponding to the support E_i.

Proof. Properties (1) and (2) follow from Proposition 22. Property (3) follows from the fact that if the minimum is taken at $\vec{\zeta} \in (\mathbb{C}^n)^m \setminus \Delta$ then the corresponding Weierstrass image $\mathcal{W}(\vec{\zeta}) = (F_1(\vec{\zeta}, \mathbf{x}), \ldots, F_N(\vec{\zeta}, \mathbf{x}))$ will be the perturbations to obtain the nearest system

$$\left(\tilde{f}_1, \ldots \tilde{f}_N\right) := \left(f_1 - F_1(\vec{\zeta}, \mathbf{x}), \ldots f_N - F_N(\vec{\zeta}, \mathbf{x})\right)$$

satisfying the conditions in the claim. □

First we give explicitly the gradient of the map

$$\|\mathcal{W}\|^2 : (\mathbb{C}^n)^m \setminus \Delta \to \mathbb{R}.$$

Similarly as in the univariate case we can consider the vanishing of the partial derivatives $\frac{\partial \|\mathcal{W}\|^2}{\partial z_{i,j}}$ instead of the partial derivatives by the real and imaginary part of $z_{i,j}$.

In order to give explicitly the gradient of the map $\|\mathcal{W}\|^2$, we use Corollary 24 and the following definition:

Definition 27. Given $f_1, \ldots, f_N$, $\vec{\mathbf{z}} = (\mathbf{z}_1, \ldots, \mathbf{z}_m)$, $\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_m$ and $E_1, \ldots, E_N$ as above. Let $F_1, \ldots, F_N$ be as in Definition 25. For each $i = 1, \ldots, N$ define the $d \times mn$ block diagonal matrix

D_i

$$D_i := \begin{bmatrix} D_{i,1} & & & \\ & D_{i,2} & & \\ & & \ddots & \\ & & & D_{i,m} \end{bmatrix}$$

where each block $D_{i,j}$ has size $l_j \times n$ and defined as

$$D_{i,j} := [\ (\partial_{x_k}\Lambda_{j,s})(f_i - F_i)(\mathbf{z}_j)]_{1\le s\le l_j, 1\le k\le n}\,. \tag{25}$$

Let $M_1, \ldots, M_N$ be as in Proposition 26. For each $i = 1, \ldots N$, $j = 1, \ldots m$ and $s = 1, \ldots, l_{i,j}$ define $\hat{F}_{i,j,s}$ and $\hat{M}_{i,j,s}$ to be the modification of F_i and M_i using the modified multiplicity structure

$$(\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_j - \{\Lambda_{j,s}\}, \ldots, \mathbf{\Lambda}_m).$$

Now we are ready to give an explicit formula for the gradient of the map $\|W\|^2$, which is a generalization of (Zhi and Wu, 1998, Theorem 1) and Proposition 13 to the multivariate case.

Proposition 28. *Using the notation of Definition 27, for $j = 1, \ldots, m$ and $k = 1, \ldots, n$ the partial derivatives $\frac{\partial\|\mathcal{W}\|^2}{\partial z_{j,k}}$ are equal to*

$$\frac{\partial\|\mathcal{W}\|^2}{\partial z_{j,k}} = \sum_{\substack{i=1,\ldots,N \\ s=1,\ldots l_{i,j}}} \frac{\det \hat{M}_{i,j,s}}{\det M_i}\ \ (\partial_{x_k}\Lambda_{j,s})(\ f_i - F_i)(\mathbf{z}_j)\ \ \overline{\Lambda_{j,s}(\ f_i - \hat{F}_{i,j,s})(\mathbf{z}_j)}.$$

Proof. Similarly to the univariate case, Corollary 24 implies that

$$M_i \frac{\partial M_i^{-1}\mathbf{f}_i}{\partial z_{j,k}} = V_i \frac{\partial \mathbf{F}_i}{\partial z_{j,k}} = [D_{i,j}]_{*,k}$$

where $\mathbf{f}_i$ was defined in Proposition 26(2), $\mathbf{F}_i$ is the coefficient vector of F_i, $V_i := V_{E_i,\mathbf{\Lambda}}(\vec{\mathbf{z}})$ is the Vandermonde matrix and $[D_{i,j}]_{*,k}$ denotes the k-th column of the matrix $D_{i,j}$ defined in (25). Therefore

$$\frac{\partial \sum_{i=1}^N \mathbf{f}_i^* M_i^{-1}\mathbf{f}_i}{\partial z_{j,k}} = \sum_{i=1}^N \mathbf{f}_i^* M_i^{-1}[D_{i,j}]_{*,k}$$

and the rest of the argument is the same as in the univariate case. □

Next we apply the Gauss-Newton method to find the least squares solution of the multivariate Weierstrass map. Since the structure of the matrices D_i is not as simple as in the univariate case, we cannot give component-wise formula for the Gauss-Newton iteration of the multivariate Weierstrass map. Therefore, we only give an explicit formula for the Gauss-Newton iteration in terms of the matrices D_i and M_i.

Proposition 29. *Let $f_1, \ldots f_N$ and $\mathcal{W}$ be as above. Using the notation above, for $\vec{\mathbf{z}} \in (\mathbb{C}^n)^m \setminus \Delta$ the Gauss-Newton iteration for $\mathcal{W}$ is given by*

$$\vec{\mathbf{z}}^{new} = \vec{\mathbf{z}} - \left(\sum_{i=1}^N D_i^* M_i^{-1} D_i\right)^{-1} \left(\sum_{i=1}^N D_i^* M_i^{-1}\mathbf{f}_i\right) \tag{26}$$

where M_i and $\mathbf{f}_i$ are defined in 26(2) and D_i is defined in Definition 27.

Proof. Note that, using Corollary 24, we get that the Jacobian matrix of $\mathcal{W}$ is given by

$$J := \begin{bmatrix} V_1^\dagger D_1 \\ \vdots \\ V_N^\dagger D_N \end{bmatrix} \tag{27}$$

where $V_i = V_{E_i,\mathbf{\Lambda}}(\vec{\mathbf{z}})$ is the Vandermonde matrix for $i = 1, \ldots, N$. Then the claim follows from the definitions of $J^\dagger$, $\mathcal{W}$, M_i and F_i. □

7. Simplified Iteration

In this section we consider a simplification of the iteration function in (26). The simplification we propose is analogous to the idea used in the classical univariate Weierstrass iteration, which we briefly describe first. The classical univariate Weierstrass iteration finds simultaneously all roots of a given monic univariate polynomial f of degree m, and has the following simple and elegant component-wise iteration function:

$$z_k^{new} = z_k - \frac{f(z_k)}{\prod_{j \neq k}(z_k - z_j)} \quad k = 1, \ldots, m.$$

One can derive this formula by applying the Newton method to the corresponding Weierstrass map, and then expressing the result in terms of the standard Lagrange polynomial basis at the iteration point: the Jacobian of the Weierstrass map is diagonal in the Lagrange basis, which results in the simple, component-wise iteration formula. Generalization of this to finding the roots of multivariate systems were proposed in Ruatta (2001).

Now we explore an analogue of the above simplification to our problem of solving approximate systems with given root multiplicities. Let N, $f_1, \ldots, f_N$, $E_1, \ldots, E_N$ and $\mathbf{\Lambda} = (\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_m)$ with cardinality $|\mathbf{\Lambda}_i| = l_i$ be as in Section 6. We will need the assumption that we choose the number of support elements in E_i so that the Vandermonde matrix $V_i = V_{E_i,\mathbf{\Lambda}}(\vec{\mathbf{z}})$ in Definition 18 is square and invertible, i.e.

$$|E_i| = d := \sum_{j=1}^{m} l_j \quad \text{for all} \quad i = 1, \ldots, N.$$

In this case the Hermite interpolation polynomials are defined from V_i^{-1}, and form a basis for the space of polynomials with support E_i. Then we can use Corollary 24 to give a simple expression for the Jacobian of $\mathcal{W}$ in the basis of Hermite interpolation polynomials. Using (27) the Jacobian of $\mathcal{W}$ written in the Hermite interpolation basis is the matrix

$$D := \begin{bmatrix} D_1 \\ \vdots \\ D_N \end{bmatrix}$$

where D_i is the block matrix defined in Definition 27. Also, the coefficient vector $\mathbf{F}_i$ of F_i written in the basis of Hermite polynomials is equal to the evaluation vector $\mathbf{f}_i$ of f_i (since $V_i \mathbf{F}_i = \mathbf{f}_i$).

Thus, the Gauss-Newton iteration written in terms of the basis of Hermite polynomials at each step of the iteration yields

$$\vec{\mathbf{z}}^{new} = \vec{\mathbf{z}} - D^{\dagger}\mathbf{f} \quad \text{where} \quad \mathbf{f} := \begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_N \end{bmatrix}. \tag{28}$$

Using the simple block structure of the matrices D_i we can rearrange the rows of D to have a block structure. This way we can get the following component-wise iteration function for the simplified Gauss-Newton iteration:

$$\mathbf{z}_i^{new} = \mathbf{z}_i - D_{\mathbf{z}_i}^{\dagger}\mathbf{f}_{\mathbf{z}_i} \quad i = 1, \ldots, m, \tag{29}$$

where

$$D_{\mathbf{z}_i} = \begin{bmatrix} D_{1,i} \\ \vdots \\ D_{N,i} \end{bmatrix} \in \mathbb{C}^{Nl_i \times n}, \qquad \mathbf{f}_{\mathbf{z}_i} = \begin{bmatrix} \Lambda_{i,1}(f_1)(\mathbf{z}_i) \\ \vdots \\ \Lambda_{i,l_i}(f_1)(\mathbf{z}_i) \\ \vdots \\ \Lambda_{i,l_i}(f_N)(\mathbf{z}_i) \end{bmatrix} \in \mathbb{C}^{Nl_i}, \tag{30}$$

and $D_{j,i}$ is defined in Definition 27.

We must note that by changing the polynomial basis, we have changed the norm that we are minimizing. In general, the iteration in (29) will converge to a different point than the iteration of (26). In the exact case however, i.e. when the input system has roots with prescribed multiplicity, the fixed points of (29) are the same as the fixed points of (26) since $\|\mathcal{W}(\mathbf{z})\|^2 = 0$ if and only if all entries of $\mathcal{W}$ are zero, regardless of the norm used.

The following proposition describes some of the fixed points of the simplified iteration. Informally, it asserts that if $\vec{\mathbf{z}}^* = (\mathbf{z}_1^*, \ldots \mathbf{z}_m^*) \in (\mathbb{C}^n)^m$ is such that the corresponding vector of perturbation functions $(F_1(\vec{\mathbf{z}}^*, \mathbf{x}), \ldots, F_N(\vec{\mathbf{z}}^*, \mathbf{x}))$ is locally pointwise minimal around all $\mathbf{z}_i^*$, then the simplified iteration (29) will have a fixed point in $\vec{\mathbf{z}}^*$. Thus, instead of finding the 2-norm minimum of the coefficient vector of the perturbation functions, the simplified iteration finds locally pointwise minimal perturbation functions. As our numerical experiments demonstrate, the simplified iteration, when compared to other methods, will return the output with the smallest residual, which is defined as the square root of

$$\sum_{r=1}^{N}\sum_{i=1}^{m}\sum_{t=1}^{l_i} |\Lambda_{i,t}(f_r)(\mathbf{z}_i^*)|^2 = \sum_{r=1}^{N}\sum_{i=1}^{m}\sum_{t=1}^{l_i} |\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}^*, \mathbf{z}_i^*)|^2 .$$

Proposition 30. *Let $f_1, \ldots, f_N$, $E_1, \ldots, E_N$ and $F_1, \ldots F_N$ be as above, with $|E_r| = d$ for all $r = 1 \ldots, N$. Let $\vec{\mathbf{z}}^* = (\mathbf{z}_1^*, \ldots \mathbf{z}_m^*) \in (\mathbb{C}^n)^m$. If there exists an open neighborhood $U \subset \mathbb{C}^{mn} - \Delta$ about the point $\vec{\mathbf{z}}^*$ such that for all $\vec{\mathbf{z}} = (\mathbf{z}_1, \ldots, \mathbf{z}_m) \in U$ and all $\vec{\mathbf{z}}' = (\mathbf{z}_1', \ldots, \mathbf{z}_m') \in U$*

$$\sum_{r=1}^{N}\sum_{t=1}^{l_i} |\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}^*, \mathbf{z}_i')|^2 \;\leq\; \sum_{r=1}^{N}\sum_{t=1}^{l_i} |\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}, \mathbf{z}_i')|^2 \quad \forall\, i = 1, \ldots, m, \tag{31}$$

then the iteration function (29) has a fixed point at $\vec{\mathbf{z}}^*$.

Proof. First note that $\vec{\mathbf{z}}^*$ is a fixed point of (29) if and only if $D^*_{\mathbf{z}_i}\mathbf{f}_{\mathbf{z}_i} = 0$ at $\vec{\mathbf{z}}^*$ for all $i = 1, \ldots, m$. On the other hand, if $\vec{\mathbf{z}}^*$ satisfies (31) then the partial derivatives

$$\frac{\partial}{\partial z_{i,j}} \left(\sum_{r=1}^{N} \sum_{t=1}^{l_i} |\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}, \mathbf{z}'_i)|^2 \right) \Bigg|_{\vec{\mathbf{z}}=\vec{\mathbf{z}}^*} = 0 \tag{32}$$

for all $i = 1, \ldots, m$, $j = 1, \ldots, n$ and $\vec{\mathbf{z}}' = (\mathbf{z}'_1, \ldots, \mathbf{z}'_m) \in U$. Fix some $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, n\}$. Next we will simplify the partial derivatives in (32). First fix also $r \in \{1, \ldots, N\}$ and $t \in \{1, \ldots, l_i\}$. By Corollary 24 and our assumption on $|E_r|$ we have that

$$\frac{\partial F_r(\vec{\mathbf{z}}, \mathbf{x})}{\partial z_{i,j}} = \sum_{s=1}^{l_i} (\partial_{x_j} \Lambda_{i,s})(f_r - F_r)(\vec{\mathbf{z}}, \mathbf{z}_i)\ h_{i,s}(\vec{\mathbf{z}}, \mathbf{x}).$$

Since the differential operators $\partial/\partial z_{i,j}$ and $\Lambda_{i,t}$ commute, it is easy to see that

$$\frac{\partial}{\partial z_{i,j}} \Lambda_{i,t}(F_r)(\vec{\mathbf{z}}, \mathbf{x}) = \sum_{s=1}^{l_i} (\partial_{x_j} \Lambda_{i,s})(f_r - F_r)(\vec{\mathbf{z}}, \mathbf{z}_i)\ \Lambda_{i,t}(h_{i,s})(\vec{\mathbf{z}}, \mathbf{x}),$$

and

$$\frac{\partial}{\partial z_{i,j}} |\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}, \mathbf{x})|^2 = \left(\sum_{s=1}^{l_i} (\partial_{x_j} \Lambda_{i,s})(f - F_r)(\vec{\mathbf{z}}, \mathbf{z}_i)\ \Lambda_{i,t}(h_{i,s})(\vec{\mathbf{z}}, \mathbf{x}) \right) \overline{\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}, \mathbf{x})}.$$

By the definition of $h_{i,s}$ we have that

$$\Lambda_{i,t}(h_{i,s})(\vec{\mathbf{z}}, \mathbf{z}_i) = \begin{cases} 1 & \text{if } s = t \\ 0 & \text{if } s \neq t \end{cases}.$$

Therefore, we get that

$$\frac{\partial}{\partial z_{i,j}} \left(\sum_{t=1}^{l_i} |\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}, \mathbf{x})|^2 \right) \Bigg|_{\mathbf{x}=\mathbf{z}_i} = \sum_{t=1}^{l_i} (\partial_{x_j} \Lambda_{i,t})(f - F_r)(\vec{\mathbf{z}}, \mathbf{z}_i)\ \overline{\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}, \mathbf{z}_i)}. \tag{33}$$

Now combining (32) and (33) with $\vec{\mathbf{z}} = \vec{\mathbf{z}}' = \vec{\mathbf{z}}^*$ we get that

$$\sum_{r=1}^{N} \sum_{t=1}^{l_i} (\partial_{x_j} \Lambda_{i,t})(f - F_r)(\vec{\mathbf{z}}^*, \mathbf{z}_i^*)\ \overline{\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}^*, \mathbf{z}_i^*)} = 0. \tag{34}$$

Using that $\Lambda_{i,t}(F_r)(\vec{\mathbf{z}}^*, \mathbf{z}_i^*) = \Lambda_{i,t}(f_r)(\mathbf{z}_i^*)$, (34) is equivalent to

$$\mathbf{f}^*_{\mathbf{z}_i} D_{\mathbf{z}_i}|_{\vec{\mathbf{z}}=\vec{\mathbf{z}}^*} = 0,$$

which proves the claim. □

8. Numerical Results

The following three versions of the iterations described in the paper were implemented in Matlab:

(1) The method with complete support. In this case we use the iteration defined in Proposition 29. The support consists of all monomials of total degree less than or equal to the total degree of the input polynomials.

(2) The method with minimal support. In this case we again use the iteration from Proposition 29. However, the number of support elements is equal to the number of constraints, making the Vandermonde matrices square.

(3) The simplified iteration. In this case we use the iteration defined in equation (29). The number of support elements is equal to the number of constraints, making the Vandermonde matrices square.

We generated random test cases in the following way:

- Two hundred random systems of three polynomials in three variables where generated.
- Random roots, $\bar{\mathbf{z}} = (\bar{\mathbf{z}}_1, \ldots, \bar{\mathbf{z}}_m) \in (\mathbb{C}^3)^m$, were generated with $0 \leq Re(\bar{z}_{i,j}) \leq 1$ and $0 \leq Im(\bar{z}_{i,j}) \leq 1$. We considered cases where $m = 2$ and $m = 3$.
- A Vandermonde system was solved with minimal support to ensure the system had the common roots with multiplicity.
- The resulting system was then perturbed by a random polynomial with known 2-norm, $\sqrt{3} \times 10^{-6}$.
- The three iteration methods described above were applied to each of these perturbed systems.
- The initial iterate for the root was chosen to be 0.4 from the original root $\bar{\mathbf{z}}$ of the non-perturbed system.
- If the iteration did not converge, a new initial iterate half the distance from the original root was chosen, and the algorithm was repeated.
- After 10 tries, if the iteration still failed, then that case was labeled divergent.

To explain our results, the following definitions based on the list above are useful:

- $\bar{f} = \{\bar{f}_1, \ldots, \bar{f}_N\}$ is the polynomial system that has the desired root structure.
- $\bar{\mathbf{z}} = (\bar{\mathbf{z}}_1, \ldots, \bar{\mathbf{z}}_m)$ are the common roots of $\bar{f}$ with the given multiplicity structure $\mathbf{\Lambda} = (\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_m)$.
- $f = \{f_1, \ldots, f_N\}$ is the perturbed system.
- $\mathbf{z}^* = (\mathbf{z}_1^*, \ldots, \mathbf{z}_m^*)$ is the m-tuple of the computed roots.
- $f^* = \{f_1^*, \ldots, f_N^*\}$ is the computed polynomial system with roots $\mathbf{z}^*$ which have the multiplicity structure $\mathbf{\Lambda}$.
- $\mathbf{f}(\mathbf{x})$ is the evaluation vector of the polynomial system f at a point $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_m)$, i.e.

$$\mathbf{f}(\mathbf{x}) = (\Lambda_{1,1}(f_1)(\mathbf{x}_1), \ldots, \Lambda_{m,l_m}(f_1)(\mathbf{x}_m), \ldots, \Lambda_{m,l_m}(f_N)(\mathbf{x}_m)).$$

The columns of the Figures 1, 2, 3, 4 are defined as follows:

- Method refers to the type of iteration used.

Method	End Res. (10^{-4})	Distance (10^{-4}) from Original	Avg. Iterations	Avg. Time (sec)
Complete	4.731	1.694	20.82	53.54
Simplified	2.632	13.34	6.53	6.762
Square	3.846	1.200	6.18	6.081

Fig. 1. 3 polynomials, total degree 5, in 3 unknowns. Looking for closest system with 2 roots of multiplicity 3 each. Test cases were generated by perturbing a system with known root structure by 1.74×10^{-4}. The residual of the known roots in the perturbed system was 6.288×10^{-4} for the complete support case and 4.356×10^{-4} for the other cases.

Method	End Res. (10^{-4})	Distance (10^{-4}) from original	Avg. Iterations	Avg. Time (sec)
Complete	6.351	1.719	24.19	105.3
Simplified	3.829	33.03	7.665	13.37
Square	5.832	1.171	6.995	12.41

Fig. 2. 3 polynomials, total degree 5, in 3 unknowns. Looking for closest system with 3 roots of multiplicity 3 each. Test cases were generated by perturbing a system with known root structure by 1.74×10^{-4}. The residual of the known roots in the perturbed system was 7.867×10^{-4} for the complete support case and 6.250×10^{-4} for the other cases.

- The ending residual (End Res.) is defined as $\|\mathbf{f}(\mathbf{z}^*)\|_2$.
- The Distance from Original is defined as $\|f - f^*\|_2$. Note that the polynomials in the difference $f - f^*$ has prescribed support by construction, and the 2-norm is the 2-norm of the coefficient vectors in the given supports.
- The average iterations (Avg. Iterations) is defined as the average number of iterations needed. The iteration terminates when an iteration step fails to reduce the residual by at least 1%.
- The average time (Avg. Time) is the average time in seconds it took the given method to complete.
- In the caption of each figure we note the beginning residual. The beginning residual is defined as $\|\mathbf{f}(\bar{\mathbf{z}})\|_2$.

The only divergent cases observed occurred in the complete support case of figure 2. In this instance 3.5% of cases were divergent. In all the test runs, over 70% of test cases converged with an initial iterate that shared only one significant digit with the roots of the original system.

References

Corless, R. M., Gianni, P. M., Trager, B. M., Watt, S. M., 1995. The singular value decomposition for polynomial systems. In: Proceedings of the International Symposium on Symbolic and Algebraic Computation. pp. 195–207.

Method	End Res. (10^{-4})	Distance (10^{-4}) from original	Avg. Iterations	Avg. Time (sec)
Complete	11.35	1.721	5.84	216.3
Simplified	3.152	13.78	7.575	23.02
Square	4.221	1.016	7.560	22.77

Fig. 3. 3 polynomials, total degree 10, in 3 unknowns. Looking for closest system with 2 roots of multiplicity 3 each. Test cases were generated by perturbing a system with known root structure by 1.74×10^{-4}. The residual of the known roots in the perturbed system was 1.530×10^{-3} for the complete support case and 1.732×10^{-4} for the other cases.

Method	End Res. (10^{-4})	Distance (10^{-4}) from original	Avg. Iterations	Avg. Time (sec)
Complete	13.55	1.716	6.21	414.9
Simplified	4.212	35.39	8.345	48.38
Square	5.983	1.006	8.305	47.69

Fig. 4. 3 polynomials, total degree 10, in 3 unknowns. Looking for closest system with 3 roots of multiplicity 3 each. Test cases were generated by perturbing a system with known root structure by 1.74×10^{-4}. The residual of the known roots in the perturbed system was 1.725×10^{-3} for the complete support case and 6.242×10^{-4} for the other cases.

Dayton, B. H., Zeng, Z., 2005. Computing the multiplicity structure in solving polynomial systems. In: ISSAC '05: Proceedings of the 2005 International Symposium on Symbolic and Algebraic Computation. ACM Press, New York, NY, USA, pp. 116–123.

Dedieu, J. P., Shub, M., 2000. Newton's method for overdetermined systems of equations. Math. Comp. 69 (231), 1099–1115.

Demmel, J., Koev, P., 2006. Accurate SVDs of polynomial Vandermonde matrices involving orthonormal polynomials. Linear Algebra Appl. 417 (2-3), 382–396.

Giusti, M., Lecerf, G., Salvy, B., Yakoubsohn, J.-C., 2005. On location and approximation of clusters of zeros of analytic functions. Found. Comput. Math. 5 (3), 257–311.

Giusti, M., Lecerf, G., Salvy, B., Yakoubsohn, J.-C., 2007. On location and approximation of clusters of zeros: case of embedding dimension one. Found. Comput. Math. 7 (1), 1–49.

Grabmeier, J., Kaltofen, E., Weispfenning, V. (Eds.), 2002. Computer Algebra Handbook: Foundations, Applications, Systems. Springer.

Gröbner, W., 1970. Algebraische Geometrie. II. Teil: Arithmetische Theorie der Polynomringe. Bibliographisches Institut, Mannheim, b. I. Hochschultaschenbücher, 737/737a*.

Hitz, M. A., Kaltofen, E., 1998. Efficient algorithms for computing the nearest polynomial with constrained roots. In: Proceedings of the 1998 International Symposium on Symbolic and Algebraic Computation. ACM Press, pp. 236–243.

Hitz, M. A., Kaltofen, E., Lakshman, Y. N., 1999. Efficient algorithms for computing the nearest polynomial with a real root and related problems. In: Proceedings of the 1999 International Symposium on Symbolic and Algebraic Computation. ACM Press, New York, pp. 205–212.

Hribernig, V., Stetter, H. J., 1997. Detection and validation of clusters of polynomial zeros. Journal of Symbolic Computation 24 (6), 667–681.

Janovitz-Freireich, I., Rónyai, L., Szántó, A., 2006. Approximate radical of ideals with clusters of roots. In: ISSAC '06: Proceedings of the 2006 International Symposium on Symbolic and Algebraic Computation. ACM Press, New York, NY, USA, pp. 146–153.

Janovitz-Freireich, I., Rónyai, L., Szántó, A., 2007. Approximate radical for clusters: a global approach using Gaussian elimination or SVD. Journal of Mathematics in Computer Science 1 (2), pp. 393–425.

Kaltofen, E., May, J., 2003. On approximate irreducibility of polynomials in several variables. In: Proceedings of the 2003 International Symposium on Symbolic and Algebraic Computation. ACM, New York, pp. 161–168.

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

Karmarkar, N. K., Lakshman, Y. N., 1998. On approximate GCDs of univariate polynomials. Journal of Symbolic Computation 26 (6), 653–666.

Lecerf, G., 2002. Quadratic Newton iteration for systems with multiplicity. Foundations of Computational Mathematics (2), 247–293.

Leykin, A., Verschelde, J., Zhao, A., 2006. Newton's method with deflation for isolated singularities of polynomial systems. Theor. Comput. Sci. 359 (1), 111–122.

Leykin, A., Verschelde, J., Zhao, A., 2007. Evaluation of Jacobian matrices for Newton's method with deflation to approximate isolated singular solutions of polynomial systems. In: Wang, D., Zhi, L. (Eds.), Symbolic-Numeric Computation. Trends in Mathematics. Birkhäuser, pp. 269–278.

Macaulay, F. S., 1994. The Algebraic Theory of Modular Systems. Cambridge Mathematical Library. Cambridge University Press, revised reprint of the 1916 original. With an introduction by Paul Roberts.

Marinari, M. G., Mora, T., Möller, H. M., 1995. Gröbner duality and multiplicities in polynomial system solving. In: ISSAC '95: Proceedings of the 1995 International Symposium on Symbolic and Algebraic Computation. ACM Press, New York, NY, USA, pp. 167–179.

Meyer, C. D., 2000. Matrix analysis and applied linear algebra. SIAM.

Möller, H. M., Sauer, T., 2000. H-bases for polynomial interpolation and system solving. Adv. Comput. Math. 12 (4), 335–362, Multivariate polynomial interpolation.

Morgan, A. P., Sommese, A. J., Wampler, C. W., 1992. Computing singular solutions to polynomial systems. Adv. in Appl. Math. 13 (3), 305–327.

Mourrain, B., 1997. Isolated points, duality and residues. J. Pure Appl. Algebra 117/118, 469–493, Algorithms for algebra (Eindhoven, 1996).

Mourrain, B., Ruatta, O., 2002. Relation between roots and coefficients, interpolation and application to system solving. Journal of Symbolic Computation 33 (5), 679–699.

Ojika, T., 1987a. Modified deflation algorithm for the solution of singular problems. I. A system of nonlinear algebraic equations. J. Math. Anal. Appl. 123 (1), 199–221.

Ojika, T., 1987b. Modified deflation algorithm for the solution of singular problems. II. Nonlinear multipoint boundary value problems. J. Math. Anal. Appl. 123 (1), 222–237.

Ojika, T., Watanabe, S., Mitsui, T., 1983. Deflation algorithm for the multiple roots of a system of nonlinear equations. J. Math. Anal. Appl. 96 (2), 463–479.

Ruatta, O., 2001. A multivariate Weierstrass iterative rootfinder. In: Proceedings of the 2001 International Symposium on Symbolic and Algebraic Computation. ACM press, London, Ontario.

Ruatta, O., Sciabica, M., Szanto, A., 2004. Over-constrained Weierstrass iteration and the nearest consistent system, accepted in Journal of Theoretical Computer Science.

Sasaki, T., 2003. The subresultant and clusters of close roots. In: Proceedings of the 2003 International Symposium on Symbolic and Algebraic Computation. pp. 232–239.

Sasaki, T., Noda, M.-T., 1989. Approximate square-free decomposition and root-finding of ill-conditioned algebraic equations. J. Inform. Process. 12 (2), 159–168.

Sauer, T., Xu, Y., 1995. On multivariate Hermite interpolation. Adv. Comput. Math. 4 (3), 207–259.

Sommese, A. J., Verschelde, J., Wampler, C. W., 2002. A method for tracking singular paths with application to the numerical irreducible decomposition. In: Algebraic geometry. de Gruyter, Berlin, pp. 329–345.

Sommese, A. J., Wampler, C. W., 1996. Numerical algebraic geometry. In: The mathematics of numerical analysis (Park City, UT, 1995). Vol. 32 of Lectures in Appl. Math. Amer. Math. Soc., Providence, RI, pp. 749–763.

Stetter, H. J., 1996. Analysis of zero clusters in multivariate polynomial systems. In: Proceedings of the 1996 International Symposium on Symbolic and Algebraic Computation. ACM Press, pp. 127–136.

Stetter, H. J., 2004. Numerical Polynomial Algebra. SIAM.

Wu, X., Zhi, L., 2008. Computing the multiplicity structure from geometric involutive form. In: Proceedings of the 2008 International Symposium on Symbolic and Algebraic Computation. pp. 325–332.

Yakoubsohn, J.-C., 2000. Finding a cluster of zeros of univariate polynomials. Journal of Complexity 16 (3), 603–638.

Zeng, Z., 2005. Computing multiple roots of inexact polynomials. Mathematics of Computation 74, 869–903.

Zhi, L., Wu, W., 1998. Nearest singular polynomials. J. Symbolic Comput. 26 (6), 667–675, Symbolic numeric algebra for polynomials.