Computing the Multiplicity Structure from Geometric Involutive Form *

Xiaoli Wu and Lihong Zhi
Key Laboratory of Mathematics Mechanization
AMSS, Beijing 100190, China
xlwu@amss.ac.cn, lzhi@mmrc.iss.ac.cn
http://www.mmrc.iss.ac.cn/~lzhi/

ABSTRACT

We present a method based on symbolic-numeric reduction to geometric involutive form to compute the primary component and the differential operators for an isolated singular solution of a polynomial ideal. The singular solution can be exact or approximate. If the singular solution is known with limited accuracy, then we propose a new method to refine it to high accuracy.

Categories and Subject Descriptors: G.1.5 [Mathematics of Computing]: Roots of Nonlinear Equations; I.1.2 [Symbolic and Algebraic Manipulation]: Algebraic Algorithms

General Terms: Algorithms, Theory

Keywords: Involutive System, Numerical Linear Algebra, Differential Operator, Index, Multiplicity

1. INTRODUCTION

Consider an ideal I generated by a polynomial system $F = \{f_1, \ldots, f_t\}$, where $f_i \in \mathbb{C}[x_1, \ldots, x_s]$, $i = 1, \ldots, t$. For a given isolated singular solution $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_s)$ of F, suppose Q is the isolated primary component whose associate prime is $P = (x_1 - \hat{x}_1, \ldots, x_s - \hat{x}_s)$, we use symbolic-numeric method based on the geometric jet theory of partial differential equations introduced in [36, 37, 43] to compute the index ρ and multiplicity μ, such that $Q = (I, P^\rho)$ and $\mu = \dim(\mathbb{C}[\mathbf{x}]/Q)$. The multiplication structure of the quotient ring $\mathbb{C}[\mathbf{x}]/Q$ is computed from the null space of the involutive form of Q. The differential operators are determined by computing the normal form of a polynomial with undetermined coefficients up to degree $\rho - 1$. If the singular solution is only known with limited accuracy, then the primary ideal Q has a cluster of solutions. A refined solution with higher accuracy can be obtained by averaging the eigenvalues of each multiplication matrix [5].

*This research was partially supported by NKBRPC (2004CB318000) and the Chinese National Natural Science Foundation under Grant 10401035.

ISSAC'08, July 20–23, 2008, Hagenberg, Austria.

Inspired by recent works in [8, 9], we apply the involutive criterion to the truncated coefficient matrices formulated from the Taylor series expansions of polynomials in prolonged systems of F at $\hat{\mathbf{x}}$ to order k. The number of columns of these coefficient matrices is fixed by $\binom{k+s-1}{s}$. The differential operators can be obtained from the null space of the truncated coefficient matrix of the involutive system. Our algorithm for computing differential operators could be regarded as a primal version of the one given in [9].

Our method for computing index and multiplicity is also related to the one presented in [2]. The algorithm they presented for computing multiplicity is based on Bayer and Stillman's theory on regularity [3]. It was pointed out in [21] that the concept of involutivity of symbol is equivalent to its Mumford regularity. Our criterion for involutivity is similar to their stopping criterion for regularity. However we do not need homogenization procedure while the algorithm in [2] works for homogenous polynomials.

If a singular solution is only known with limited accuracy, by choosing a tolerance, we can compute the index, multiplicity and differential operators for this approximate singular solution. It is well known that numeric computations deeply depend on the choice of tolerance. In order to obtain accurate information about the multiplicity structure, we propose a method to improve the accuracy of the singular root.

Suppose $\hat{\mathbf{x}} = \hat{\mathbf{x}}_{\text{exact}} + \hat{\mathbf{x}}_{\text{error}}$ where $\hat{\mathbf{x}}_{\text{exact}}$ denotes the exact singular solution of F. We observe that a good approximation $\hat{\mathbf{y}}$ of $-\hat{\mathbf{x}}_{\text{error}}$ can be computed from the null vectors of the truncated coefficient matrix of the involutive system. The singular solution $\hat{\mathbf{x}} + \hat{\mathbf{y}}$ has higher accuracy compared with $\hat{\mathbf{x}}$. We can apply our procedure iteratively to $\hat{\mathbf{x}} + \hat{\mathbf{y}}$ with a smaller tolerance. A singular solution accurately to the full machine precision can usually be obtained in less than 3 iterations, as is shown in our experiments. It is still not clear how our refinement procedure related to the methods in [18, 19, 20, 32, 33]. The column dimension of the matrix we used for refining the approximate singular solution is $\binom{\rho+s}{s}$. Our algorithm for refining an approximate singular solution is not efficient when index ρ is big. We notice that the number of deflations of the algorithms in [19, 20] is not closely related to the index, their algorithms can be very efficient for singular solutions with large index.

All algorithms we present in this paper have been implemented in Maple 11. We give two examples to illustrate our methods along the paper. We also show the test results for a set of benchmark problems. All computations are done in Maple 11 with Digits := 14.

2. ISOLATED PRIMARY COMPONENT

2.1 Preliminaries

The following paragraphs give a brief outline of the notations and tools we use throughout this paper. We refer to [6, 41] for detailed introduction.

Definition 1. *Let I be an ideal in the ring of polynomials over complex field denoted by $\mathbb{C}[\mathbf{x}] = \mathbb{C}[x_1, \ldots, x_s]$. Let f, g be arbitrary elements in $\mathbb{C}[\mathbf{x}]$.*

- *I is* prime *if $fg \in I \Longrightarrow f \in I$ or $g \in I$.*
- *I is* primary *if $fg \in I \Longrightarrow f \in I$ or $g^m \in I$ for some positive integer m.*
- *I is* radical *if $f^m \in I \Longrightarrow f \in I$.*
- *The* radical of I *is the set*

$$\sqrt{I} = \{f \mid f^m \in I \text{ for some integer } m \geq 1\}.$$

It should be noted that $\sqrt{I}$ is an ideal. Every prime ideal is a radical ideal and the radical of a primary ideal is a prime ideal. An ideal is finitely generated if there exists a finite list of elements $f_1, f_2, \ldots, f_t \in I$ such that every element in I can be written as a $\mathbb{C}[\mathbf{x}]$-linear combination of $f_1, f_2, \ldots, f_t$, and denoted by $I = (f_1, f_2, \ldots, f_t)$.

Definition 2. *If P and Q are ideals and have the property that (1) $fg \in Q$ and $f \notin Q$ implies $g \in P$, (2) $Q \subseteq P$, (3) $g \in P$ implies $g^\rho \in Q$ for some positive integer ρ, then Q is* primary *and P the* prime ideal belonging to Q.

If Q is a primary ideal then $P = \sqrt{Q}$ is the prime ideal belonging to Q and Q is called P-primary.

Definition 3. *Every polynomial ideal has an irredundant primary decomposition, i.e. $I = \cap_{i=1}^{r} Q_i$, where Q_i are primary, $Q_i \subsetneq \cap_{j \neq i} Q_j$. We call Q_i a* primary component (ideal) *of I. Q_i is said to be isolated if no prime ideal belonging to Q_j, $j \neq i$, is divisible by a prime ideal belonging to Q_i.*

Definition 4. *ρ is called the index of a primary ideal Q if ρ is the minimal nonnegative integer such that $\sqrt{Q}^\rho \subseteq Q$.*

Theorem 1. *[41] Suppose the polynomial ideal I has an isolated primary component Q whose associated prime P is maximal, and ρ is the index of Q.*
If $\sigma < \rho$, then

$$\dim(\mathbb{C}[\mathbf{x}]/(I, P^{\sigma-1})) < \dim(\mathbb{C}[\mathbf{x}]/(I, P^\sigma)). \quad (1)$$

If $\sigma \geq \rho$, then

$$Q = (I, P^\rho) = (I, P^\sigma). \quad (2)$$

Corollary 1. *If a polynomial ideal I has an isolated primary component Q whose associated prime P is maximal, then the index ρ of Q is less than or equal to the multiplicity μ of Q.*

Proof: The multiplicity μ of the isolated primary component Q is equal to the dimension of the quotient algebra $\mathbb{C}[\mathbf{x}]/(I, P^\rho)$. Since the dimension of $\mathbb{C}[\mathbf{x}]/(I, P^\sigma)$ increases strictly until $\sigma = \rho$, the multiplicity μ is bigger than or equal to the index ρ. □

2.2 SNEPSolver

Consider a polynomial system $F = \{f_1, \ldots, f_t\}$, where $f_i \in \mathbb{C}[x_1, \ldots, x_s]$ is of degree d, $i = 1, \ldots, t$ and $s \leq t$. The system can be written as

$$M_d^{(0)} \cdot [x_1^d, x_1^{d-1}x_2, \ldots, x_s^2, x_1, \ldots, x_s, 1]^T = [0, \ldots, 0]^T$$

in terms of its coefficient matrix $M_d^{(0)}$. Here and hereafter, $[...]^T$ means the transposition. Further, $[\xi_1, \xi_2, \ldots, \xi_s]$ is one of the solutions of the polynomial system, if and only if

$$[\xi_1^d, \xi_1^{d-1}\xi_2, \ldots, \xi_s^2, \xi_1, \ldots, \xi_s, 1]^T$$

is a null vector of the coefficient matrix $M_d^{(0)}$.

Since the number of monomials is usually much greater than the number of polynomials, the dimension of the null space can be large. Completion methods for polynomial ideals based on critical pairs [1, 10, 11, 17, 24, 25, 28, 29, 30, 31, 39, 40] aim to include additional polynomials belonging to the ideal generated by F, until the normal form is determined capable of deciding membership of the ideal. The method in [35, 36, 37, 43] focuses on direct methods to calculate and minimize these dimensions by using the criterion of involution for PDE R [15, 34]. Here R is equivalent to polynomial system F by the bijection

$$\phi : x_i \leftrightarrow \frac{\partial}{\partial x_i}, \quad 1 \leq i \leq s.$$

In the following, we briefly explain the symbolic-numeric elimination method in the language of polynomial algebra. We study the variety

$$V(F) = \left\{ [\mathbf{x}^d, \ldots, 1] \in \mathbb{C}^{N_d} \mid M_d^{(0)} \cdot [\mathbf{x}^d, \ldots, 1]^T = \mathbf{0} \right\},$$

where $N_d = \binom{d+s}{s}$, $\mathbf{x}^j$ denotes all monomials of total degree equal to j. All distinct monomials are regarded as independent variables and $V(F)$ is simply the null space of $M_d^{(0)}$.

A single prolongation of the system F is to multiply the polynomials in F by variables, so that the resulting augmented system has degree $d+1$. Successive prolongations of the system yield $F = F^{(0)}, F^{(1)}, F^{(2)}, \ldots$, and a sequence of corresponding linear constant matrix systems:

$$M_d^{(0)} \cdot \mathbf{v_d} = \mathbf{0}, M_d^{(1)} \cdot \mathbf{v_{d+1}} = \mathbf{0}, M_d^{(2)} \cdot \mathbf{v_{d+2}} = \mathbf{0}, \cdots$$

where $\mathbf{v_i} = \left[\mathbf{x}^i, \mathbf{x}^{i-1}, \ldots, \mathbf{x}, 1\right]^T$.

A single geometric projection is defined as

$$\boldsymbol{\pi}(F) = \left\{ [\mathbf{x}^{d-1}, \ldots, 1] \in \mathbb{C}^{N_{d-1}} \mid \exists \mathbf{x}^d, M_d^{(0)} \cdot [\mathbf{x}^d, \ldots, 1]^T = \mathbf{0} \right\}.$$

The projection operator $\boldsymbol{\pi}$ maps a point in $\mathbb{C}^{N_d}$ to one in $\mathbb{C}^{N_{d-1}}$ by eliminating the monomials of the highest degree d. A numeric projection operator $\hat{\boldsymbol{\pi}}$ based on singular value decomposition (SVD) was purposed in [4, 36, 42]. We first find the singular value decomposition $M_d^{(0)} = U \cdot \Sigma \cdot V$. The approximate rank r of $M_d^{(0)}$ is the number of singular values bigger than a fixed tolerance. The tolerance is chosen according to the number of correct digits for the coefficients of the input polynomials. The dimension of F is defined as the dimension of the null space of $M_d^{(0)}$, so we have $\dim F = \dim \text{Nullspace}(M_d^{(0)}) = N_d - r$. Deleting the first r rows of V yields an approximate basis for the null space of $M_d^{(0)}$. To estimate $\dim \hat{\boldsymbol{\pi}}(F)$, the components

of the approximate basis for the null space of $M_d^{(0)}$ corresponding to the monomials of the highest degree d are deleted. This projected basis yields an approximate spanning set for $\hat{\boldsymbol{\pi}}(F)$. Application of the SVD to each of these approximate spanning sets yields the approximate dimensions of $\hat{\boldsymbol{\pi}}(F)$, $\hat{\boldsymbol{\pi}}^2(F)$, $\hat{\boldsymbol{\pi}}^3(F)$, ..., which are required for the approximate involutive form test.

The symbol matrix of polynomials of degree d is simply the submatrix of the coefficient matrix $M_d^{(0)}$ corresponding to the monomials of the highest degree d. One of the most important requirements of involutive systems is that their symbols are involutive. The following criterion of involution for zero dimensional polynomial systems is given in [43].

Theorem 2. *[43] A zero dimensional polynomial system F is involutive at order m and projected order ℓ if and only if $\boldsymbol{\pi}^\ell(F^{(m)})$ satisfies the projected elimination test:*

$$\dim \boldsymbol{\pi}^\ell \left(F^{(m)}\right) = \dim \boldsymbol{\pi}^{\ell+1} \left(F^{(m+1)}\right), \tag{3}$$

and the symbol involutive test:

$$\dim \boldsymbol{\pi}^\ell \left(F^{(m)}\right) = \dim \boldsymbol{\pi}^{\ell+1} \left(F^{(m)}\right). \tag{4}$$

The following algorithm given in [37, 43] solves zero dimensional polynomial systems based on the symbolic-numeric completion method.

Algorithm 1. *SNEPSolver*
Input: *A zero dimensional ideal $I = (f_1, \ldots, f_t)$ where the polynomials are in $\mathbb{C}[\mathbf{x}]$ of degree d and a tolerance τ.*
Output: *Dimension of the quotient ring $\mathbb{C}[\mathbf{x}]/I$ and its multiplication matrices $M_{x_1}, \ldots, M_{x_s}$.*

- *Apply the symbolic-numeric completion method to $F = \{f_1, \ldots, f_t\}$ with tolerance τ, we obtain the table of $\dim \hat{\boldsymbol{\pi}}^\ell(F^{(m)})$.*
- *We seek the smallest m such that there exists an ℓ with $\hat{\boldsymbol{\pi}}^\ell(F^{(m)})$ approximately involutive, i.e., satisfying the conditions (3, 4). If there are several such values for the given m, then choose the largest such ℓ.*
- *The number of solutions of polynomial system F is $d = \dim(\mathbb{C}[\mathbf{x}]/I) = \dim \hat{\boldsymbol{\pi}}^\ell(F^{(m)})$.*
- *The multiplication matrices $M_{x_1}, \ldots, M_{x_s}$ are formed from the null vectors of $\hat{\boldsymbol{\pi}}^\ell(F^{(m)})$ and $\hat{\boldsymbol{\pi}}^{\ell+1}(F^{(m)})$.*

Remark 1. Instead of choosing monomials to form a normal set of size d, we compute the SVD of the approximate basis of the null space of $\hat{\boldsymbol{\pi}}^{\ell+1}(F^{(m)})$. According to (4), the first d left singular vectors permit a stable representation of the other rows in the approximate basis of the null space of $\hat{\boldsymbol{\pi}}^\ell(F^{(m)})$, a polynomial basis formed from these singular vectors leads to a stable representation of multiplicative structure of the quotient ring $\mathbb{C}[\mathbf{x}]/I$. The solutions of F can be obtained by computing eigenvalues and eigenvectors of the multiplication matrices [1, 5, 25].

2.3 Algorithm for Computing Isolated Primary Component

For a given isolated solution of the ideal $I = (f_1, \ldots, f_t)$, suppose Q is the isolated primary component whose associate prime $P = (x_1 - \hat{x}_1, \ldots, x_s - \hat{x}_s)$, we apply SNEPSolver to compute the index ρ, such that $Q = (I, P^\rho)$ and the multiplication structure of the quotient ring $\mathbb{C}[\mathbf{x}]/Q$.

Algorithm 2. *IsolatedPrimaryComponent*
Input: *An isolated multiple solution $\hat{\mathbf{x}}$ of an ideal $I = (f_1, \ldots, f_t)$, a tolerance τ.*
Output: *The multiplicity μ, the index ρ, and multiplication matrices $M_{x_1}, \ldots, M_{x_s}$ of the quotient ring $\mathbb{C}[\mathbf{x}]/Q$ where $Q = (I, P^\rho)$.*

- *Form the prime ideal $P = (x_1 - \hat{x}_1, \ldots, x_s - \hat{x}_s)$.*
- *Compute $d_k = \dim(\mathbb{C}[\mathbf{x}]/(I, P^k))$ as described above by SNEPSolver for the given tolerance τ until $d_k = d_{k-1}$, then set $\rho = k - 1$, $\mu = d_\rho$ and $Q = (I, P^\rho)$.*
- *Compute the multiplication matrices $M_{x_1}, \ldots, M_{x_s}$ of $\mathbb{C}[\mathbf{x}]/Q$ by SNEPSolver.*

Symbolic methods based on the uniqueness of the reduced Gröbner basis are given in [12, 16] to determine the index of Q. However, when the multiple zero is only known with finite precision, their methods are subject to numerical stability problem.

Remark 2. The set made up of these computed multiplication matrices $\{M_{x_1}, \ldots, M_{x_s}\}$ is called numerical local ring in [8] for a given root $\hat{\mathbf{x}}$.

Since the ideal (I, P^k) is generated by polynomials

$$F_k = \{f_1, \ldots, f_t,\ (x_1 - \hat{x}_1)^{\alpha_1} \cdots (x_s - \hat{x}_s)^{\alpha_s},\ \sum_{i=1}^{s} \alpha_i = k\}.$$

Without loss of generality, suppose $d \le k$, we prolong all polynomials f_i to have degree k. Since all monomials of degree $k + j$ appear in the prolonged system $F_k^{(j)}$, the symbol matrices of $F_k^{(j)}$ always have full rank, i.e., the symbols of $F_k^{(j)}$ are involutive. The dimension of the prolonged system, denoted by $\dim F_k^{(j)} = \dim \text{Nullspace}(M_k^{(j)})$, decreases strictly until it is stabilized, where $M_k^{(j)}$ is the coefficient matrix of $F_k^{(j)}$. So we have the following simple criterion of involution for polynomial system F_k.

Theorem 3. *The zero dimensional polynomial system F_k is involutive at order m if and only if*

$$\dim F_k^{(m)} = \dim F_k^{(m+1)}. \tag{5}$$

Example 1. *[32] Consider an ideal I generated by the polynomials*

$$\{f_1 = x_1^2 + x_2 - 3,\ f_2 = x_1 + 0.125x_2^2 - 1.5\}. \tag{6}$$

The system has $(1, 2)$ as a 3-fold solution.

Form the maximal ideal $P = (x_1 - 1, x_2 - 2)$.

- $k = 2$, we consider the system

  $$F_2 = \{f_1,\ f_2,\ (x_1-1)^2,\ (x_1-1)(x_2-2),\ (x_2-2)^2\}.$$

  Since $\dim F_2 = 1$, $\dim F_2^{(1)} = \dim F_2^{(2)} = 2$, we have $\dim(\mathbb{C}[\mathbf{x}]/(I, P^2)) = 2$.

Similarly, we have

- $k = 3$, $\dim F_3 = 1$, $\dim F_3^{(1)} = \dim F_3^{(2)} = 3$, we have $\dim(\mathbb{C}[\mathbf{x}]/(I, P^3)) = 3$.
- $k = 4$, $\dim F_4 = 1$, $\dim F_4^{(1)} = \dim F_4^{(2)} = 3$, we have $\dim(\mathbb{C}[\mathbf{x}]/(I, P^4)) = 3$.

Therefore, the index and multiplicity of the root $(1, 2)$ are: $\rho = 3$, $\mu = 3$. The multiplication matrices with respect to the normal set $\{x_1, x_2, 1\}$ are:

$$M_{x_1} = \begin{bmatrix} 0 & -1 & 3 \\ 6 & 3 & -10 \\ 1 & 0 & 0 \end{bmatrix},\ M_{x_2} = \begin{bmatrix} 6 & 3 & -10 \\ -8 & 0 & 12 \\ 0 & 1 & 0 \end{bmatrix} \tag{7}$$

The triple eigenvalues of M_{x_1} and M_{x_2} are 1 and 2 respectively.

If the singular solution $\hat{\mathbf{x}}$ is only known approximately, then the polynomial system F_k has a cluster of solutions. The Schur factorization of multiplication matrices M_{x_i} consists of only one block. As shown in [5], the average of the cluster eigenvalues of M_{x_i} computed by $\mathrm{Trace}(M_{x_i})/\mu$ gives a refined value for $\hat{x}_i$. We can apply the procedure again for the refined singular solution and obtain singular solution with higher accuracy.

Example 1 (continued) Suppose we are given an approximate singular solution:

$$\hat{\mathbf{x}} = (1 + 2.5428 \times 10^{-4} + 2.4352 \times 10^{-4}\, i,\\ 2 + 8.4071 \times 10^{-4} + 3.6129 \times 10^{-4}\, i).$$

We choose a tolerance $\tau = 10^{-4}$ and apply Algorithm 2 to $\hat{\mathbf{x}}$ and the polynomial system (6). The dimensions computed for the given tolerance are the same as shown above. Therefore, we get the same index and multiplicity for the approximate singular solution with respect to the given tolerance. The refined root computed from the multiplication matrices is:

$$(1 + 9.5829 \times 10^{-8} - 1.2762 \times 10^{-7}\, i,\\ 2 - 2.6679 \times 10^{-6} + 3.5569 \times 10^{-7}\, i).$$

Then use this refined solution as an initial one and set $\tau = 10^{-6}$, run Algorithm 2 again, we obtain:

$$(1 - 1.0000 \times 10^{-15} + 2.5854 \times 10^{-14}\, i, 2 + 8.4457 \times 10^{-14}).$$

3. MODIFIED SNEPSOLVER FOR COMPUTING DIFFERENTIAL OPERATORS

To apprehend the structure of the dual space of an ideal at a multiple zero further, we calculate a basis for the dual space.

Let $D(\alpha) = D(\alpha_1, \ldots, \alpha_s) : \mathbb{C}[\mathbf{x}] \to \mathbb{C}[\mathbf{x}]$ denote the differential operator defined by:

$$D(\alpha_1, \ldots, \alpha_s) = \frac{1}{\alpha_1! \cdots \alpha_s!} \partial x_1^{\alpha_1} \cdots \partial x_s^{\alpha_s},$$

for non-negative integer array $\alpha = [\alpha_1, \ldots, \alpha_s]$. We write $\mathfrak{D} = \{D(\alpha),\ |\alpha| \geq 0\}$ and denote by $Span_{\mathbb{C}}(\mathfrak{D})$ the $\mathbb{C}$-vector space generated by $\mathfrak{D}$ and introduce a morphism on $\mathfrak{D}$ that acts as "integral":

$$\sigma_{x_j}(D(\alpha)) = \begin{cases} D(\alpha_1, \ldots, \alpha_j - 1, \ldots, \alpha_s), & \text{if } \alpha_j > 0, \\ 0, & \text{otherwise.} \end{cases}$$

Definition 5. *A subspace L of $Span_{\mathbb{C}}(\mathfrak{D})$ is said to be closed if*

$$\sigma_{x_j}(L) \subseteq L,\ j = 1, \ldots, s.$$

Definition 6. *Given a zero $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_s)$ of an ideal I, we define the subspace of differential operators associated to I and $\hat{\mathbf{x}}$ as*

$$\triangle_{\hat{\mathbf{x}}} := \{L \in Span_{\mathbb{C}}(\mathfrak{D}) |\, L(f)|_{\mathbf{x}=\hat{\mathbf{x}}} = 0,\ \forall f \in I\}. \tag{8}$$

Theorem 4. *[7] Let M be the maximal ideal $(x_1, \ldots, x_s)$ of $\mathbb{C}[\mathbf{x}]$. There is a bijective correspondence between M-primary ideals of $\mathbb{C}[\mathbf{x}]$ and closed subspaces of $Span_{\mathbb{C}}(\mathfrak{D})$:*

$$\{M\text{-primary ideals in } \mathbb{C}[\mathbf{x}]\}\\ \uparrow\downarrow\\ \{\text{closed subspaces of } \mathrm{Span}_{\mathbb{C}}(\mathfrak{D})\}.$$

Moreover, for a zero dimensional M-primary ideal of $\mathbb{C}[\mathbf{x}]$ whose multiplicity is μ, we have that $\dim_{\mathbb{C}}(\triangle_{\hat{\mathbf{x}}}) = \mu$.

3.1 Algorithm for Computing Differential Operators

The following algorithm computes the differential operators from the output of Algorithm 2. Moreover, the differential operators evaluated at the multiple zero are functionals which constitute a set of bases for the dual space of the ideal at the multiple zero.

Algorithm 3. *DifferentialOperatorsI*
Input: *Multiplication matrices $M_{x_1}, \ldots, M_{x_s}$, multiple zero $\hat{\mathbf{x}}$ and index ρ.*
Output: *$L = \{L_1, \ldots, L_\mu\}$, a basis for the space $\triangle_{\hat{\mathbf{x}}}$.*

- *Write the Taylor expansion at $\hat{\mathbf{x}}$ of a polynomial $h \in \mathbb{C}[\mathbf{x}]$ up to degree $\rho - 1$ with coefficients $c_\alpha \in \mathbb{C}$:*

  $$\mathrm{T}_{\rho-1}(h) = \sum_{\alpha \in \mathbb{N}^s, |\alpha| < \rho} c_\alpha (x_1 - \hat{x}_1)^{\alpha_1} \cdots (x_s - \hat{x}_s)^{\alpha_s}.$$

- *Compute the normal form of h from the multiplication matrices $M_{x_1}, \ldots, M_{x_s}$, and expand it at $\hat{x}$,*

  $$\mathrm{NF}(h(x)) = \sum_\beta d_\beta (\mathbf{x} - \hat{\mathbf{x}})^\beta.$$

- *Find scalars $a_{\alpha\beta} \in \mathbb{C}$ such that $d_\beta = \sum_\alpha a_{\alpha\beta} c_\alpha$. For each β such that $d_\beta \neq 0$, return the operator*

  $$L_\beta = \sum_\alpha a_{\alpha\beta} \frac{1}{\alpha_1! \cdots \alpha_s!} \partial x_1^{\alpha_1} \cdots \partial x_s^{\alpha_s} = \sum_\alpha a_{\alpha\beta} D(\alpha).$$

An alternative procedure for computing these differential operators based on Gröbner basis computation is given in [7]. Our algorithm can be applied to polynomial system with floating point coefficients since it computes the normal form stably from the multiplication matrices computed by

SNEPSolver. Furthermore, in [7], the degree of polynomial $h(\mathbf{x})$ is bounded by the multiplicity μ which is larger or equal to our degree bound ρ according to Corollary 1.

Example 1 (continued) We compute the differential operators at the exact solution $(1, 2)$:

- Write the Taylor expansion of a polynomial at $(1, 2)$ up to degree $\rho - 1 = 2$,

$$h(\mathbf{x}) = c_{0,0} + c_{1,0}(x_1 - 1) + c_{0,1}(x_2 - 2) + c_{2,0}(x_1 - 1)^2 + c_{1,1}(x_1 - 1)(x_2 - 2) + c_{0,2}(x_2 - 2)^2.$$

- From the multiplication matrices (7), we obtain the normal form of h by replacing x_1^2, x_1x_2, x_2^2 with

$$x_1^2 = -x_2 + 3, x_1x_2 = 6x_1 + 3x_2 - 10, x_2^2 = -8x_1 + 12.$$

The differential operators are:

$$\begin{cases} L_1 &= D(0,0), \\ L_2 &= D(0,1) - D(2,0) + 2D(1,1) - 4D(0,2), \\ L_3 &= D(1,0) - 2D(2,0) + 4D(1,1) - 8D(0,2). \end{cases}$$

Corollary 2. *Applying the differential operators output from Algorithm 3 to the polynomials in F, we obtain a new polynomial system*

$$\{L_j(f_i) \mid L_j \in L, f_i \in F,\ 1 \le j \le \mu,\ 1 \le i \le t\}, \quad (9)$$

and $\hat{\mathbf{x}}$ becomes its isolated simple solution.

Proof: If $\hat{\mathbf{x}}$ is a singular solution of system (9), then there exists a non-trivial differential operator L_τ, its differential order $\mid \tau \mid \ge 1$. Take $L_\beta \in L$ that has the highest differential order in L. Then $L_\tau \circ L_\beta$ is a differential operator for F at $\hat{\mathbf{x}}$ which is not included in L. It is a contradiction. □

Suppose the multiple solution $\hat{\mathbf{x}}$ is only known with limited accuracy. Deflation method demonstrated in [19, 20] adds new equations and variables to F to have $\hat{\mathbf{x}}$ become a simple solution, then use Newton iteration to refine it. The new polynomial system (9) also has $\hat{\mathbf{x}}$ as its simple root, however, it is not clear whether we can use the attained information to refine $\hat{\mathbf{x}}$.

3.2 Specialized SNEPSolver

We have applied Algorithms 2 and 3 to a set of examples shown in the Table 1. For some examples, the systems become too large after we add all monomials of high degrees. In this subsection, we propose a modified SNEPSolver. The matrices we used to verify the involutivity are much smaller than the ones used by SNEPSolver.

Suppose $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_s)$ is an isolated zero of a set of multivariate polynomials $F = \{f_1, \ldots, f_t\}$. Let $P = (x_1 - \hat{x}_1, \ldots, x_s - \hat{x}_s)$ and I be an ideal having P-primary isolated component. Let

$$\mathrm{T}_k(F) = \{\mathrm{T}_k(f_1), \ldots, \mathrm{T}_k(f_t)\},$$

where $\mathrm{T}_k(f_i) = \sum_{|\alpha|<k} f_{i,\alpha}(\mathbf{x} - \hat{\mathbf{x}})^\alpha$ denotes truncated Taylor series expansions of the polynomial f_i at $\hat{\mathbf{x}}$ to order k. The zero can be moved to the origin by changing of variables. For simplicity, we suppose $\hat{\mathbf{x}}$ is the origin.

The ideal (I, P^k) is generated by the polynomials $F_k = \mathrm{T}_k(F) \cup P^k$. The lower submatrices of the symbol matrices of F_k and prolonged system $F_k^{(j)}$ are identity matrices corresponding to monomials in P^k and P^{k+j}. Therefore the symbol matrices are of full column rank, the dimensions and null spaces of coefficient matrices of the systems F_k and $F_k^{(j)}$ can be computed from the coefficient matrices generated by the truncated systems $\mathrm{T}_k(F)$ and $\mathrm{T}_k(F^{(j)})$. So we can work with the coefficient matrices of truncated system $\mathrm{T}_k(F)$ and truncated prolonged systems $\mathrm{T}_k(F^{(j)})$ instead of the coefficient matrices of $F \cup P^k$ and their prolongations. For simplicity, we still use $M_k^{(j)}$ to denote the coefficient matrices of the system $\mathrm{T}_k(F^{(j)})$. Let $d_k^{(j)} = \dim \mathrm{Nullspace}(M_k^{(j)})$.

Since all polynomials are truncated by degree k, the coefficient matrix $M_k^{(j)}$ has only $\binom{k+s-1}{s}$ columns. Furthermore, the number of prolongations m has an upper bound: $m \le \max(1, k - \min(\mathrm{ldeg}(f_1), \ldots, \mathrm{ldeg}(f_t)))$, where $\mathrm{ldeg}(f)$ denotes the lowest degree of f.

Example 2. *[19] The following polynomial system has* 15 *regular solutions and three* 4*-fold multiple solutions:*

$$\{f_1 = x_1^3 + x_2^2 + x_3^2 - 1, f_2 = x_1^2 + x_2^3 + x_3^2 - 1, f_3 = x_1^2 + x_2^2 + x_3^3 - 1\}$$

We pick the multiple root $\hat{\mathbf{x}} = (1, 0, 0)$. By changing of variables, we get a new system $\{g_1 = x_1^3 + 3x_1^2 + 3x_1 + x_2^2 + x_3^2, g_2 = x_1^2 + 2x_1 + x_2^3 + x_3^2, g_3 = x_1^2 + 2x_1 + x_2^2 + x_3^3\}$ has the 4-fold multiple solution $\hat{\mathbf{x}} = (0, 0, 0)$. Let $P = (x_1, x_2, x_3)$.

- $k = 2$. We have:

$$[\mathrm{T}_2(g_1), \mathrm{T}_2(g_2), \mathrm{T}_2(g_3)]^T = M_2^{(0)} \cdot [x_1, x_2, x_3, 1]^T,$$

where

$$M_2^{(0)} = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 \end{bmatrix}$$

and $d_2^{(0)} = 3$. The prolonged matrix $M_2^{(1)}$ is obtained by adding zero elements to $M_2^{(0)}$. Hence $d_2^{(1)} = 3$ and

$$d_2 = \dim(\mathbb{C}[\mathbf{x}]/(I, P^2)) = 3.$$

- $k = 3$. We have:

$$[\mathrm{T}_3(g_1), \mathrm{T}_3(g_2), \mathrm{T}_3(g_3)]^T = M_3^{(0)} \cdot \left[x_1^2, \ldots, x_3, 1\right]^T,$$

where

$$M_3^{(0)} = \begin{bmatrix} 3 & 0 & 0 & 1 & 0 & 1 & 3 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & 0 & 0 \end{bmatrix}$$

and $d_3^{(0)} = 7$. After the first prolongation, we have:

$$[\mathrm{T}_3(x_1g_1), \mathrm{T}_3(x_1g_2), \mathrm{T}_3(x_1g_3), \ldots, \mathrm{T}_3(g_3)]^T = M_3^{(1)} \cdot \left[x_1^2, x_1x_2, x_1x_3, x_2^2, x_2x_3, x_3^2, x_1, x_2, x_3, 1\right]^T,$$

where

$$M_3^{(1)} = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 1 & 0 & 1 & 3 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & 0 & 0 \end{bmatrix}.$$

We have $d_3^{(1)} = 4$. The prolonged matrix $M_3^{(2)}$ is obtained by adding zeros to $M_3^{(1)}$, hence $d_3^{(2)} = 4$, and

$$d_3 = \dim(\mathbb{C}[\mathbf{x}]/(I, P^3)) = 4.$$

- $k = 4$. We compute $d_4^{(0)} = 17$, $d_4^{(1)} = 8$, and $d_4^{(2)} = d_4^{(3)} = 4$. Hence,

$$d_4 = \dim(\mathbb{C}[\mathbf{x}]/(I, P^4)) = 4.$$

Since $d_3 = d_4 = 4$, the multiplicity of $\hat{\mathbf{x}} = (0, 0, 0)$ is $\mu = 4$ and the index of the primary ideal is $\rho = 3$, and $Q = (I, P^3)$.

The null space of the matrix $M_3^{(1)}$ has dimension 4 and can be written as

$$N_3^{(1)} = [e_{10}, e_9, e_8, e_5],$$

where e_i is the i-th column of 10×10 identity matrix.

It is interesting to notice that the differential operators can be obtained by multiplying the differential operators of order less than 3 with the 4 null vectors in $N_3^{(1)}$:

$$\{D(0,0,0),\ D(0,0,1),\ D(0,1,0),\ D(0,1,1)\}.$$

It is not a coincidence that the differential operators can be obtained from null vectors of the coefficient matrix $M_3^{(1)}$.

Theorem 5. *Let $Q = (I, P^\rho)$ be the isolated primary component of an ideal $I = (f_1, \ldots, f_t)$ at the multiple solution $\hat{\mathbf{x}}$ and μ be the multiplicity of $\hat{\mathbf{x}}$. Suppose the system $F_\rho = \mathrm{T}_\rho(F) \cup P^\rho$ is involutive at prolongation order m, the null space of the matrix $M_\rho^{(m)}$ is generated by vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_\mu$. Let*

$$\mathbf{L} = [D(\rho-1, 0, \ldots, 0), D(\rho-2, 1, 0, \ldots, 0), \ldots, D(0, \ldots, 0)]$$

denote the vector consists of all differential operator of order less than ρ. Then a basis for the space $\triangle_{\hat{\mathbf{x}}}$ can be computed as

$$L_j = \mathbf{L} \cdot \mathbf{v}_j, \quad \text{for } 1 \le j \le \mu.$$

Proof: Since the system F_ρ is involutive at prolongation order m, for any polynomial $f \in I$, the coefficient vector $\mathbf{f}$ of the truncated polynomial $\mathrm{T}_\rho(f)$ can be expressed as $\mathbf{f} = \mathbf{c} \cdot M_\rho^{(m)}$, where $\mathbf{c}$ is a complex row vector. For $1 \le j \le \mu$, we have

$$L_j(f) \mid_{\mathbf{x}=\hat{\mathbf{x}}} = L_j(\mathrm{T}_k(f)) \mid_{\mathbf{x}=\hat{\mathbf{x}}} = \mathbf{c} \cdot M_\rho^{(m)} \cdot \mathbf{v}_j = 0. \qquad \square$$

Algorithm 4. *DifferentialOperatorsII*
Input: *An isolated multiple solution $\hat{x} = (\hat{x}_1, \ldots, \hat{x}_s)$ of an ideal $I = (f_1, \ldots, f_t)$ and a tolerance τ.*

Output: *The multiplicity μ, the index ρ of the primary component $Q = (I, P^\rho)$ and a set of differential operators $L = \{L_1, \ldots, L_\mu\}$.*

- *Form the coefficient matrix $M_k^{(0)}$ by computing the truncated multivariate Taylor series expansions of $f_1, \ldots, f_t$ at $\hat{x}$ to order k. The prolonged matrix $M_k^{(j)}$ is computed by shifting $M_k^{(0)}$ accordingly.*
- *Compute $d_k^{(j)} = \dim \mathrm{Nullspace}(M_k^{(j)})$ for the given τ. The prolongation is stopped until $d_k^{(m)} = d_k^{(m+1)} = d_k$.*
- *If $d_k = d_{k-1}$, then set $\rho = k - 1$ and $\mu = d_\rho$.*
- *Compute the null vectors of $M_\rho^{(m)}$ denoted by $\mathbf{v}_1, \ldots, \mathbf{v}_\mu$. The differential operators are computed as $L_j = \mathbf{L} \cdot \mathbf{v}_j$, for j from 1 to μ.*

Remark 3. If the zero is not at the origin, we can also compute the matrix $M_k^{(0)}$ by changing of variables $y_i = x_i - \hat{x}_i$ for $1 \le i \le s$ and compute the coefficients of polynomials $f_1(y_1 + \hat{x}_1, \ldots, y_s + \hat{x}_s), \ldots, f_t(y_1 + \hat{x}_1, \ldots, y_s + \hat{x}_s)$ with respect to the variables $y_1, \ldots, y_s$. However, we only need the coefficients of $y_1^{\alpha_1} \cdots y_s^{\alpha_s}$ with total degree less than k. Hence it is more efficient and numerically stable to compute the Taylor expansions at $\hat{\mathbf{x}}$ to order k.

There is an impressive paper written by Dayton and Zeng [9]. They compute the differential operators from the dual space. We obtain the differential operators from studying the primal space directly.

3.3 Algorithm for Refining Approximate Singular Solution

Suppose we are only given an approximate root

$$\hat{\mathbf{x}} = \hat{\mathbf{x}}_{\text{exact}} + \hat{\mathbf{x}}_{\text{error}},$$

where $\hat{\mathbf{x}}_{\text{error}}$ denotes the error in the solution and $\hat{\mathbf{x}}_{\text{exact}}$ denotes the exact solution of the polynomial system $F = \{f_1, \ldots, f_t\}$ with multiplicity μ and index ρ. The output of Algorithm 4 is a set of differential operators satisfying approximately,

$$L_j(f_i) \mid_{\mathbf{x}=\hat{\mathbf{x}}} \approx 0, \quad 1 \le j \le \mu, \quad 1 \le i \le t.$$

According to Remark 3, the matrix $M_k^{(0)}$ in Algorithm 4 is the coefficient matrix of truncated polynomials $\mathrm{T}_k(G) = \{\mathrm{T}_k(g_1), \ldots, \mathrm{T}_k(g_t)\}$ where

$$g_i = f_i(y_1 + \hat{x}_{1,\text{exact}} + \hat{x}_{1,\text{error}}, \ldots, y_s + \hat{x}_{s,\text{exact}} + \hat{x}_{s,\text{error}})$$

obtained by changing of variables $y_i = x_i - \hat{x}_i$, for $1 \le i \le s$. The output of Algorithm 4 can be regarded equivalently as the differential operators of system $G = \{g_1, \ldots, g_t\}$ at its approximate solution $\hat{\mathbf{y}} = 0$,

$$L_j(g_i) \mid_{\mathbf{y}=\hat{\mathbf{y}}} \approx 0, \quad 1 \le j \le \mu, \quad 1 \le i \le t.$$

Suppose Algorithm 4 computes the multiplicity μ and index ρ correctly, and the system $G_{\rho+1} = \mathrm{T}_{\rho+1}(G) \cup P^{\rho+1}$, $P = (y_1, \ldots, y_s)$ is involutive at prolongation order m for a given tolerance. The matrix $M_{\rho+1}^{(m)}$ corresponds to the coefficient matrix of the prolonged system $\mathrm{T}_{\rho+1}(G^{(m)})$ by truncating the monomials $\mathbf{y}^\alpha$ for $\alpha \ge \rho + 1$. Similar to Algorithm 1, we can use the null vectors of the matrix $M_{\rho+1}^{(m)}$ to form the multiplication matrices and compute a solution $\hat{\mathbf{y}}$ which is usually a good approximation for $-\hat{\mathbf{x}}_{\text{error}}$. This is mainly due to the fact that

$$\hat{\mathbf{y}} = -\hat{\mathbf{x}}_{\text{error}} = (-\hat{x}_{1,\text{error}}, \ldots, -\hat{x}_{s,\text{error}})$$

is an exact solution of the system G with the multiplicity μ and index ρ.

Since all computations are done modular the ideal $P^{\rho+1} = (y_1, \ldots, y_s)^{\rho+1}$. If the given approximate solution $\hat{\mathbf{x}}$ is not far away from the exact solution $\hat{\mathbf{x}}_{\text{exact}}$, then we have $\hat{\mathbf{y}} \approx 0$, the terms $\hat{\mathbf{y}}^\alpha$ for $\alpha > \rho$ will be much closer to a zero vector. We are performing a generalized Newton iteration modular $P^{\rho+1}$. The profound theory still need to be investigated in future. We refer to [18] for interesting discussion on quadratic Newton iteration for systems with multiplicity.

Algorithm 5. *MultipleRootRefiner(MRR for short)*
Input: *An isolated approximate singular solution* $\hat{\mathbf{x}}$ *of an ideal* $I = (f_1, \ldots, f_t)$ *and a tolerance* τ.
Output: *Refined solution* $\hat{\mathbf{x}}$, *the multiplicity* μ *and index* ρ *of the primary component* $Q = (I, P^{\rho})$ *and differential operators* $L = \{L_1, \ldots, L_\mu\}$ *for the refined solution.*

- *For given approximate root* $\hat{\mathbf{x}}$ *and tolerance* τ, *applying Algorithm 4 to estimate the multiplicity* μ *and index* ρ.
- *Suppose the truncated system* $G_{\rho+1}$ *is involutive at prolongation order* m, *then form the multiplication matrices* $M_{x_1}, \ldots, M_{x_s}$ *from the null vectors of the matrix* $M_{\rho+1}^{(m)}$. *An approximate solution* $\hat{\mathbf{y}}$ *is obtained by averaging the trace of each multiplication matrix.*
- *Set* $\hat{\mathbf{x}} = \hat{\mathbf{x}} + \hat{\mathbf{y}}$ *and run the first two steps for the refined solution. The tolerance is decreased according to the solution* $\hat{\mathbf{y}}$.
- *If* $\hat{\mathbf{y}}$ *converges to the origin, then we get the refined solution* $\hat{\mathbf{x}}$ *with high accuracy. We apply Algorithm 4 to compute the differential operators with respect to the refined solution. Otherwise, we decrease the tolerance and run the above steps again.*

Example 2 (continued) Suppose we are given an approximate solution $\hat{\mathbf{x}} = (1.001, -0.002, -0.001\,i)$. By changing of variables, we have a perturbed system

$$G = \{f_j(y_1 + 1.001, y_2 - 0.002, y_3 - 0.001\,i),\ j = 1, 2, 3\}.$$

Applying Algorithm 4 to estimate multiplicity and index for a given tolerance $\tau = 10^{-2}$.

- The singular values of $M_3^{(1)}$ are: $\{6.2680, \cdots, 0.2167, 4.6071 \times 10^{-3}, 1.7236 \times 10^{-5}, 5.5960 \times 10^{-7}, 3.5123 \times 10^{-9}\}$ and $d_3^{(1)} = 4$.
- The singular values of $M_3^{(2)}$ are: $\{6.2680, \cdots, 0.2168, 5.8533 \times 10^{-3}, 2.2785 \times 10^{-5}, 8.1321 \times 10^{-6}, 6.4922 \times 10^{-9}\}$ and $d_3^{(2)} = 4$.
- The singular values of $M_4^{(2)}$ are: $\{7.2589, \cdots, 0.07891, 2.0256 \times 10^{-5}, 5.2991 \times 10^{-8}, 7.4807 \times 10^{-9}, 7.4618 \times 10^{-12}\}$ and $d_4^{(2)} = 4$.

So that for the given tolerance $\tau = 10^{-2}$, the multiplicity is $\mu = 4$ and the index is $\rho = 3$. The multiplication matrices are formed from the null space of $M_4^{(2)}$. The solution computed by averaging the eigenvalues of the multiplication matrices is

$$\begin{aligned}\hat{\mathbf{y}} = (&-0.0009994 - 7.5315 \times 10^{-10}\,i,\\ &0.002001 + 2.8002 \times 10^{-8}\,i,\\ &-1.4949 \times 10^{-6} + 0.0010000\,i).\end{aligned}$$

Adding $\hat{\mathbf{y}}$ to $\hat{\mathbf{x}}$, we obtain the refined solution $\hat{\mathbf{x}}$. We apply Algorithm 5 twice to the new singular solution $\hat{\mathbf{x}}$ for tolerance 10^{-5} and 10^{-8} respectively, and get the refined solution:

$$\begin{aligned}\hat{\mathbf{x}} = (&1 + 7.0405 \times 10^{-18} - 7.8146 \times 10^{-19}\,i,\\ &1.0307 \times 10^{-16} - 1.9293 \times 10^{-17}\,i,\\ &1.5694 \times 10^{-16} + 7.9336 \times 10^{-17}\,i).\end{aligned}$$

4. EXPERIMENTS

The following experiments are done for Digits := 14 in Maple 11 under Windows. The systems DZ1 and DZ2 are quoted from [9]. The system D2 [8] is positive dimensional, but we can compute its isolated zero dimensional primary component at the origin. The other examples are cited from the PHCpack demos http://www.math.uic.edu/~jan/. The second column lists the singular solutions $\hat{\mathbf{x}}$, where $Z_2 = (-.7071, .4082, .5774, .2500, -.1443, -.4082)$. We use ρ and μ to represent the index and the multiplicity respectively. The fifth column lists the increase in the number of correct digits from the initial guess to the refined solutions by SNEPSolver. The empty box denotes that SNEPSolver fails after running out of memory. The sixth columns shows the increase in the number of correct digits of the approximate solutions obtained by Algorithm MultipleRootRefiner.

System	Zero	ρ	μ	SNEPSolver	MRR
cmbs1	$(0, 0, 0)$	5	11	$5 \to 14$	$3 \to 11 \to 15$
cmbs2	$(0, 0, 0)$	4	8	$5 \to 15$	$3 \to 13 \to 15$
mth191	$(0, 1, 0)$	3	4	$5 \to 10 \to 15$	$4 \to 9 \to 15$
LVZ	$(0, 0, -1)$	7	18		$5 \to 10 \to 14$
KSS	$(1, 1, 1, 1, 1, 1)$	5	16		$5 \to 11 \to 14$
Caprasse	$(2, -i\sqrt{3}, 2, i\sqrt{3})$	3	4	$5 \to 14$	$4 \to 12 \to 15$
DZ1	$(0, 0, 0, 0)$	11	131	$5 \to 14$	$5 \to 14$
DZ2	$(0, 0, -1)$	8	16		$4 \to 7 \to 14$
tangents1	Z_2	4	4		$3 \to 10 \to 16$
D2	$(0, 0, 0)$	5	5	$5 \to 10 \to 15$	$5 \to 10 \to 15$
Ojika1	$(1, 2)$	3	3	$5 \to 7 \to 14$	$3 \to 6 \to 18$
Ojika2	$(0, 1, 0)$	2	2	$5 \to 10 \to 15$	$5 \to 10 \to 14$
Ojika3	$(0, 0, 1)$	3	4	$5 \to 9 \to 14$	$4 \to 8 \to 15$
Ojika4	$(0, 0, 1)$	3	3	$5 \to 10 \to 14$	$3 \to 7 \to 15$

Table 1: Algorithm Performance

5. CONCLUSION

The multiplicity structure of a singular solution has been studied extensively in [2, 7, 8, 9, 14, 22, 23, 25, 26, 27]. Various methods have also been proposed for computing the singular solutions to high accuracy [5, 18, 19, 20, 32, 33]. In this paper, we describe algorithms based on the geometric involutive form to completely describe the multiplicity structure of an isolated singular solution.

If the polynomial system and the singular solutions are known exactly, the tolerance is set to be zero. We compute the multiplicity, index and differential operators by exact linear algebra computation. If we are given an approximate isolated singular solution of an exact polynomial system, then we refine the singular solution to have high accuracy and obtain accurate multiplicity structure with respect to the refined solution. However, if the polynomials are only known with limited accuracy, the results we computed depend on a properly chosen tolerance. If there is no information about correct digits of the input data, we decide the tolerance by checking the biggest jump in the singular values of the coefficient matrix of the polynomial system F. It is interesting to investigate whether we can find a nearby polynomial system which has an isolated singular solution with given structure, such as multiplicity, index or differential operators.

Suppose the ideal $I = (f_1, \ldots, f_t)$ is zero dimensional, then applying our method to each root, we can compute the primary decomposition of I. However, I need not to be zero dimensional. We only require that the primary component Q is isolated and zero dimensional. Furthermore, the algorithms we present can be generalized to compute an isolated primary component for a maximal ideal represented by polynomials, not only the ones generated by the singular solutions.

Acknowledgments

We greatly thank Greg Reid for fruitful discussions. Our thanks go also to Philippe Trébuchet for interesting discussions during the first Chinese-SALSA workshop.

6. REFERENCES

[1] Auzinger, W., and Stetter, H. An elimination algorithm for the computation of all zeros of a system of multivariate polynomial equations. *Intern. Series in Numer. Math. 86* (1988), 11–30.

[2] Bates, D., Peterson, C., and Sommese, A. A numerical-symbolic algorithm for computing the multiplicity of a component of an algebraic set. *Journal of Complexity 22* (2006), 475–489.

[3] Bayer, D., and Stillman, M. A criterion for detecting m-regularity. *Inventiones Mathematicae 87*, 1 (1987), 1–11.

[4] Bonasia, J., Lemaire, F., Reid, G., Scott, R., and Zhi, L. Determination of approximate symmetries of differential equations. In *CRM Proceedings and Lecture Notes* (2004), pp. 233–249.

[5] Corless, R., Gianni, P., and Trager, B. A reordered Schur factorization method for zero-dimensional polynomial systems with multiple roots. In *Proc. 1997 Internat. Symp. Symbolic Algebraic Comput. ISSAC'97* (New York, 1997), Küchlin, Ed., ACM Press, pp. 133–140.

[6] Cox, D., Little, J., and O'Shea, D. *Ideals, Varieties, and Algorithms*. Springer-Verlag, New York, 1992. Undergraduate Texts in Mathematics.

[7] Damiano, A., Sabadini, I., and Struppa, D. Computational methods for the construction of a class of Noetherian operators. *Experiment. Math. 16* (2007), 41–55.

[8] Dayton, B. Numerical local rings and local solutions of nonlinear systems. In *SNC'07 Proc. 2007 Internat. Workshop on Symbolic-Numeric Comput.* (New York, 2007), J. Verschelde and S. M. Watt, Eds., ACM Press, pp. 79–86.

[9] Dayton, B., and Zeng, Z. Computing the multiplicity structure in solving polynomial systems. In Kauers [13], pp. 116–123.

[10] Faugère, J. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). In *Proc. 2002 Internat. Symp. Symbolic Algebraic Comput. ISSAC'02* (New York, 2002), T. Mora, Ed., ACM Press, pp. 75–83.

[11] Gerdt, V., and Blinkov, Y. Involutive bases of polynomial ideals. *Mathematics and Computers in Simulation 45* (1998), 519–541.

[12] Gianni, P., Trager, B., and Zacharias, G. Gröbner bases and primary decomposition of polynomial ideals. *J. Symbolic Comput. 6* (1988), 149–167.

[13] Kauers, M., Ed. *ISSAC '05 Proc. 2005 Internat. Symp. Symbolic Algebraic Comput.* (New York, 2005), ACM Press.

[14] Kobayashi, H., Suzuki, H., and Sakai, Y. Numerical calculation of the multiplicity of a solution to algebraic equations. *Math. Comp. 67(221)* (1998), 257–270.

[15] Kuranishi, M. On E.Cartan's prolongation theorem of exterior differential systems. *Amer. J. Math 79* (1957), 1–47.

[16] Lakshman, Y. A single exponential bound on the complexity of computing Gröbner bases of zero dimensional ideals. *Effective Methods in Algebraic Geometry, Progress in Mathematics* (1994), 227–234.

[17] Lazard, D. Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In *Proc. European Conf. Comput. Algebra. Lect. Notes in Comp. Sci., 162, Springer* (1983), pp. 146–157.

[18] Lecerf, G. Quadratic Newton iteration for systems with multiplicity. *Foundations of Computational Mathematics 2*, 3 (2002), 247–293.

[19] Leykin, A., Verschelde, J., and Zhao, A. Newton's method with deflation for isolated singularities of polynomial systems. *Theoretical Computer Science 359* (2006), 111–122.

[20] Leykin, A., Verschelde, J., and Zhao, A. Higher-order deflation for polynomial systems with isolated singular solutions. Manuscript, 19 pages, Jan 2007.

[21] Malgrange, B. Cartan involutiveness = Mumford regularity. *Contemp. Math. 331* (2003), 193–205.

[22] Marinari, M., Mora, T., and Möller, H. Gröbner duality and multiplicities in polynomial solving. In *Proc. 1995 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'95)* (New York, 1995), A. Levelt, Ed., ACM Press, pp. 167–179.

[23] Marinari, M., Mora, T., and Möller, H. On multiplicities in polynomial system solving. *Trans. Amer. Math. Soc. 348* (1996), 3283–3321.

[24] Möller, H., and Sauer, T. H-bases for polynomial interpolation and system solving. *Advances Comput. Math. 12* (2000), 23–35.

[25] Möller, H., and Stetter, H. Multivariate polynomial equations with multiple zeros solved by matrix eigenproblems. *Numer. Math. 70* (1995), 311–329.

[26] Möller, H., and Tenberg, R. Multivariate polynomial system solving using intersections of eigenspaces. *J. Symbolic Comput. 30* (2001), 1–19.

[27] Mourrain, B. Isolated points, duality and residues. *J. of Pure and Applied Algebra 117 & 118* (1996), 469–493.

[28] Mourrain, B. A new criterion for normal form algorithms. In *AAECC* (Berlin, 1999), M. Fossorier, H. Imai, S. Lin, and A. Poli, Eds., vol. 1719, Springer, pp. 430–443.

[29] Mourrain, B., and Trébuchet, P. Solving projective complete intersection faster. In *Proc. 2000 Internat. Symp. Symbolic Algebraic Comput. ISSAC'00* (New York, 2000), C. Traverso, Ed., ACM Press, pp. 430–443.

[30] Mourrain, B., and Trébuchet, P. Algebraic methods for numerical solving. In *Proceedings of the 3rd International Workshop on Symbolic and Numeric Algorithms for Scientific Computing* (2002), pp. 42–47.

[31] Mourrain, B., and Trébuchet, P. Generalized normal forms and polynomial system solving. In Kauers [13], pp. 253–260.

[32] Ojika, T. Modified deflation algorithm for the solution of singular problems. *J. Math. Anal. Appl. 123* (1987), 199–221.

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

[34] Pommaret, J. *Systems of Partial Differential Equations and Lie Pseudogroups*. Gordon and Breach Science Publishers, 1978.

[35] Reid, G., Scott, R., Wu, W., and Zhi, L. Algebraic and geometric properties of nearby projectively involutive polynomial systems. Manuscript, 2005.

[36] Reid, G., Tang, J., and Zhi, L. A complete symbolic-numeric linear method for camera pose determination. In *Proc. 2003 Internat. Symp. Symbolic Algebraic Comput. ISSAC'03* (New York, 2003), J. Sendra, Ed., ACM Press, pp. 215–223.

[37] Reid, G., and Zhi, L. Solving polynomial systems via symbolic-numeric elimination method. J. Symbolic Comput. (2008). To appear.

[38] Seiler, W. *Involution - The formal theory of differential equations and its applications in computer algebra and numerical analysis*. Habilitation thesis, Univ. of Mannheim, Germany, 2002.

[39] Stetter, H. *Numerical Polynomial Algebra*. SIAM, Philadelphia, 2004.

[40] Trébuchet, P. *Vers une Résolution Stable et Rapide des Équations Algébriques*. Dissertation, Université Pierre et Marie Curie, France, 2002.

[41] van der Waerden B. L. *Algebra*. Frederick Ungar Pub. Co., 1970.

[42] Wittkopf, A., and Reid, G. Fast differential elimination in C: The CDiffelim Environment. *Comp. Phys. Comm. 139(2)* (2001), 192–217.

[43] Zhi, L., and Reid, G. Solving nonlinear polynomial system via symbolic-numeric elimination method. In *Proc. International conference on polynomial system solving* (2004), J. Faugére and F. Rouillier, Eds., pp. 50–53.