Computing the Multiplicity Structure of an Isolated Singular Solution: Case of Breadth One

Nan Li and Lihong Zhi [1]

Key Lab of Mathematics Mechanization, AMSS
Beijing 100190, China

Abstract

We present an explicit algorithm to compute a closed basis of the local dual space of $I = (f_1, \ldots, f_t)$ at a given isolated singular solution $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_s)$ when the Jacobian matrix $J(\hat{\mathbf{x}})$ has corank one. The algorithm is efficient both in time and memory use. Moreover, it can be modified to compute an approximate basis if the coefficients of $f_1, \ldots, f_t$ and $\hat{\mathbf{x}}$ are only known with limited accuracy.

1 Introduction

Motivation and problem statement. 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 $P = (x_1 - \hat{x}_1, \ldots, x_s - \hat{x}_s)$. In (Wu and Zhi, 2008), we used symbolic-numeric method based on the geometric jet theory of partial differential equations introduced in (Reid et al., 2003; Zhi and Reid, 2004; Bonasia et al., 2004) to compute the index ρ, the minimal nonnegative integer such that $P^\rho \subseteq Q$, and the multiplicity $\mu = \dim(\mathbb{C}[\mathbf{x}]/Q)$, where $Q = (I, P^\rho)$. A basis for the local dual space of I at $\hat{\mathbf{x}}$ is obtained from the null space of the truncated coefficient matrix of the involutive system. The size of these coefficient matrices is bounded by $t\binom{\rho+s}{s} \times$

Email addresses: `linan08@amss.ac.cn` (Nan Li), `lzhi@mmrc.iss.ac.cn` (Lihong Zhi).

[1] This research is supported by a NKBRPC 2011CB302400 and the Chinese National Natural Science Foundation under Grants 60821002/F02, 60911130369 and 10871194.

Submitted for publication.

$\binom{\rho+s}{s}$ which will be very big when ρ or s is large. In general $\rho \leq \mu$, however, when the corank of the Jacobian matrix is one, then $\rho = \mu$, which is also called the breadth one case in (Dayton and Zeng, 2005; Dayton et al., 2009), the size of the matrices grows extremely fast with the multiplicity μ. As pointed out in (Zeng, 2009), the matrix size becomes the main bottleneck that slows down the overall computation. This is the main motivation for us to consider whether we can compute the multiplicity structure of $\hat{\mathbf{x}}$ efficiently in this worst case.

In (Dayton and Zeng, 2005; Dayton et al., 2009), they presented an efficient algorithm for computing a dual basis for the breadth one case by solving a deflated system of size roughly $(\mu t) \times (\mu s)$. A general construction of a Gauss basis of differential conditions at a multiple point was also given in (Marinari et al., 1996, Section 4.3), the breath one case is just a special case. The size of linear systems they constructed is bounded by $(\mu t) \times (\mu s)$, and they assumed that I is a zero dimensional system. In (Stetter, 2004, Section 8.5), an algorithmic approach for determining a basis of the local dual space incrementally was stated and some examples were given to show that only a sizeable number of free parameters are needed when we compute the k-th order differential condition.

Main contribution. In the breadth one case, following Stetter's arguments and smart strategies given in (Stetter, 2004, Section 8.5), we prove that the number of free parameters used in computing each order of the differential condition of I at $\hat{\mathbf{x}}$ can be reduced to $s-1$. So that we can compute the multiplicity structure of an isolated multiple zero $\hat{\mathbf{x}}$ very efficiently by solving $\mu - 2$ linear systems with size bounded by $t \times (s-1)$. Moreover, during the computation, we only need to store polynomials, the LU decomposition of the last $s-1$ columns of the Jacobian matrix and the computed differential operators. Therefore, in the breadth one case, both storage space and execution time for computing a closed basis of the local dual space are reduced significantly. Furthermore, we modify the algorithm for computing an approximate basis when singular solutions and polynomials are only known approximately.

Structure of the paper. Section 2 is devoted to recalling some notations and well-known facts. In Section 3, we prove that for the breadth one case, a closed basis of the local dual space of I at $\hat{\mathbf{x}}$ can be constructed incrementally by checking whether a differential operator parameterized by $s-1$ variables is consistent with polynomials in I. In Section 4, we describe an algorithm for computing a closed basis of the local dual space of I at $\hat{\mathbf{x}}$ and the multiplicity μ. If I and $\hat{\mathbf{x}}$ are only known with limited accuracy, then we modify the symbolic algorithm by introducing one more parameter and using singular value decomposition or LU decomposition with pivoting to ensure the numeric stability of the algorithm. Three examples are given to demonstrate that our algorithms are applicable to positive dimensional systems, analytic systems and polynomial systems with irrational or approximate coefficients. The complexity analysis and experiments are done in Section 5. We mention some ongoing research in Section 6.

2 Preliminaries

Suppose we are given an isolated multiple root $\hat{\mathbf{x}}$ of the polynomial system $F = \{f_1, \ldots, f_t\}$ with multiplicity μ and index ρ.

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 $\mathrm{Span}_{\mathbb{C}}(\mathfrak{D})$ the $\mathbb{C}$-vector space generated by $\mathfrak{D}$ and introduce a morphism on $\mathfrak{D}$ that acts as "integral":

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

As a counterpart of the anti-differentiation operator Φ_j, we define the differentiation operator Ψ_j as

$$\Psi_j(D(\alpha)) := D(\alpha_1, \ldots, \alpha_j + 1, \ldots, \alpha_s).$$

Definition 1 *Given a zero $\hat{\mathbf{x}} = (\hat{x}_1, \ldots, \hat{x}_s)$ of an ideal $I = (f_1, \ldots, f_t)$, we define the local dual space of I at $\hat{\mathbf{x}}$ as*

$$\triangle_{\hat{\mathbf{x}}}(I) := \{L \in \mathrm{Span}_{\mathbb{C}}(\mathfrak{D}) \mid L(f)|_{\mathbf{x}=\hat{\mathbf{x}}} = 0,\ \forall f \in I\}. \tag{1}$$

The vector space $\triangle_{\hat{\mathbf{x}}}(I)$ and conditions equivalent to $L(f)|_{\mathbf{x}=\hat{\mathbf{x}}} = 0,\ \forall L \in \triangle_{\hat{\mathbf{x}}}(I)$ are also called Max Noether space and Max Noether conditions in Möller and Tenberg (2001) respectively.

For a non-negative integer k, $\triangle_{\hat{\mathbf{x}}}^{(k)}(I)$ consists of differential operators in $\triangle_{\hat{\mathbf{x}}}(I)$ with the differential order bounded by k. We have that $\dim_{\mathbb{C}}(\triangle_{\hat{\mathbf{x}}}(I)) = \mu$, where μ is the multiplicity of the zero $\hat{\mathbf{x}}$.

Definition 2 *A subspace $\triangle_{\hat{\mathbf{x}}}$ of $\mathrm{Span}_{\mathbb{C}}(\mathfrak{D})$ is said to be closed, if its dimension is finite, and if*

$$L \in \triangle_{\hat{\mathbf{x}}} \Longrightarrow \Phi_j(L) \in \triangle_{\hat{\mathbf{x}}},\ j = 1, \ldots, s. \tag{2}$$

Suppose $\mathrm{Span}(L_0, L_1, \ldots, L_{\mu-1})$ is closed and $L_0, \ldots, L_{\mu-1}$ are linearly independent differential operators satisfy that $L_i(f_j)|_{\mathbf{x}=\hat{\mathbf{x}}} = 0,\ j = 1, \ldots, t,\ i = 0, \ldots, \mu - 1$, then due to the closedness, $L_i(q \cdot f_j)|_{\mathbf{x}=\hat{\mathbf{x}}} = 0,\ \forall q \in \mathbb{C}[x_1, \ldots, x_s]$. Hence, $\triangle_{\hat{\mathbf{x}}}(I) = \mathrm{Span}(L_0, L_1, \ldots, L_{\mu-1})$.

Remark 2.1 *Suppose* $\triangle_{\hat{\mathbf{x}}}(I) = \mathrm{Span}(L_0, L_1, \ldots, L_{\mu-1})$*, then* $\{L_{0,\hat{\mathbf{x}}}, \ldots, L_{\mu-1,\hat{\mathbf{x}}}\}$ *is a dual basis of the local dual space of* I *at* $\hat{\mathbf{x}}$*, where* $L_{i,\hat{\mathbf{x}}}(f) := L_i(f)|_{\mathbf{x}=\hat{\mathbf{x}}}$*. Hence, for simplicity, in the following context, we only show how to compute a closed basis of the local dual space of* I *at* $\hat{\mathbf{x}}$*.*

Lemma 2.2 *Let* $J(\hat{\mathbf{x}})$ *be the Jacobian matrix of a polynomial system* $F = \{f_1, \ldots, f_t\}$ *evaluated at* $\hat{\mathbf{x}}$*. Suppose the corank of* $J(\hat{\mathbf{x}})$ *is one, i.e., the dimension of its null space is one, then* $\dim(\triangle_{\hat{\mathbf{x}}}^{(k)}(I)) = \dim(\triangle_{\hat{\mathbf{x}}}^{(k-1)}(I)) + 1$ *for* $1 \le k \le \mu - 1$ *and* $\dim(\triangle_{\hat{\mathbf{x}}}^{(k)}(I)) = \dim(\triangle_{\hat{\mathbf{x}}}^{(\mu-1)}(I))$*, for* $k \ge \mu$*. Hence* $\mu = \rho$*.*

Proof. Lemma 2.2 is an immediate consequence of (Stanley, 1973, Theorem 2.2) and (Dayton and Zeng, 2005, Lemma 1). □

3 The Local Dual Space of Breadth One

In this section, we are mainly interested in computing a closed basis of the local dual space $\triangle_{\hat{\mathbf{x}}}(I)$ when the corank of the Jacobian matrix $J(\hat{\mathbf{x}})$ is one. In (Stetter, 2004, Section 8.5), an algorithmic approach for determining a basis of the local dual space incrementally was stated and some examples were given to show that only a sizeable number of free parameters are needed when we compute the k-th order differential condition. It is very interesting to see that in the breadth one case, the number of free parameters used in computing the k-th order differential condition following Stetter's strategies can be reduced to $s - 1$. We state below our main theorem.

Theorem 3.1 *Suppose we are given an isolated multiple root* $\hat{\mathbf{x}}$ *of the polynomial system* $F = \{f_1, \ldots, f_t\}$ *with multiplicity* μ *and the corank of the Jacobian matrix* $J(\hat{\mathbf{x}})$ *is one, and* $L_1 = D(1, 0, \ldots, 0) \in \triangle_{\hat{\mathbf{x}}}^{(1)}(I)$*. We can construct the* k*-th order differential condition incrementally for* k *from* 2 *to* $\mu - 1$ *by the following formula:*

$$L_k = P_k + a_{k,2}D(0, 1, \ldots, 0) + \cdots + a_{k,s}D(0, \ldots, 1), \tag{3}$$

where P_k *has no free parameters and is obtained from the computed basis* $\{L_1, \ldots, L_{k-1}\}$ *by the following formula:*

$$P_k = \Psi_1(Q_1) + \Psi_2((Q_2)_{i_1=0}) + \cdots + \Psi_s((Q_s)_{i_1=i_2=\cdots=i_{s-1}=0}), \tag{4}$$

where

$$Q_1 = L_{k-1}, \quad Q_j = a_{2,j}L_{k-2} + \cdots + a_{k-1,j}L_1, \; 2 \le j \le s. \tag{5}$$

Here $i_1 = \cdots = i_{j-1} = 0$ means that we only pick up terms which do not contain derivatives in $\partial x_1, \ldots, \partial x_{j-1}$, and $a_{i,j}$ are known parameters appearing in L_i for $2 \le i \le k-1$ and $2 \le j \le s$.

The parameters $a_{k,j}, \; j = 2, \ldots, s$ in (3) are determined by checking whether

$$[P_k(f_1)|_{\mathbf{x}=\hat{\mathbf{x}}}, \ldots, P_k(f_t)|_{\mathbf{x}=\hat{\mathbf{x}}}]^T$$

can be written as a linear combination of the last $s-1$ linearly independent columns of the Jacobian matrix $J(\hat{\mathbf{x}})$.

Remark 3.2 *It has been pointed out in (Stetter, 2004, Section 8.5) that if $L_1 \in \triangle_{\hat{\mathbf{x}}}^{(1)}(I)$ is not $D(1, 0, \ldots, 0)$ but a linear combination of $\partial x_1, \ldots, \partial x_s$, then we can perform linear transformation of the variables which takes the vector of the linear combination into a unit vector $(1, \ldots, 0)^T$ and reduces the situation to the one where $L_1 = D(1, 0, \ldots, 0)$. However, the change of variables usually will destroy the sparsity structure of input polynomials and might be avoided by using directional derivative (Apostol, 1974; Stetter, 2004).*

Let us suppose now that the given isolated multiple root $\hat{\mathbf{x}}$ of an ideal $I = (f_1, \ldots, f_t)$ has multiplicity μ and the corank of its Jacobian matrix $J(\hat{\mathbf{x}})$ is one, and $L_0 = D(0, \ldots, 0), L_1 = D(1, 0, \ldots, 0) \in \triangle_{\hat{\mathbf{x}}}^{(1)}(I)$. In the following, we show how to compute incrementally from L_0, L_1, a closed set of linearly independent differential operators $L_2, \ldots, L_{\mu-1}$ of derivative order $2, \ldots, \mu-1$ respectively, and $\triangle_{\hat{\mathbf{x}}}(I) = \mathrm{Span}(L_0, L_1, L_2, \ldots, L_{\mu-1})$.

Lemma 3.3 *Suppose $\{L_0, \ldots, L_{\mu-1}\}$ is a closed set of μ linearly independent differential operators which form a basis of the local dual space $\triangle_{\hat{\mathbf{x}}}(I)$, where the highest order derivative of L_k is k, then $D(k, 0, \ldots, 0)$ is the only term in L_k consisting of the k-th derivative.*

Proof. The proof is done by induction on k. It is clear that Lemma 3.3 is true for $k = 0, 1$. Our inductive assumption is that, L_{k-1} has only one term $D(k-1, 0, \ldots, 0)$ as the $(k-1)$-th derivative, therefore, the k-th order differential operator which retains closedness can only be $\Psi_j(D(k-1, 0, \ldots, 0))$ for $1 \le j \le s$. However, when $j \ne 1$, ${\Phi_1}^{k-1}(\Psi_j(D(k-1, 0, \ldots, 0))) = \Psi_j(D(0, \ldots, 0))$ which does not belong to the subspace generated by $\{L_0, L_1\}$ and violate the closedness condition. Hence, $j = 1$ and the only k-th order derivative in L_k is $D(k, 0, \ldots, 0)$. □

According to Lemma 3.3, in the following, we suppose that

$$L_k = D(k, 0, \ldots, 0) + \{\text{derivatives of order bounded by } k-1\}.$$

Moreover, we assume that there are no terms $D(i, 0, \ldots, 0)$ for $i < k$ appear in L_k, otherwise, we can reduce it by L_i.

Lemma 3.4 *Under the assumptions above, we have*

$$\begin{aligned} \Phi_1(L_k) &= L_{k-1}, \\ \Phi_j(L_k) &= c_{k-2,j}L_{k-2} + \cdots + c_{0,j}L_0, \ 2 \le j \le s. \end{aligned} \tag{6}$$

Proof. Suppose

$$\Phi_1(L_k) = L_{k-1} + c_{k-2,1}L_{k-2} + \cdots + c_{0,1}L_0.$$

If $c_{i,1} \neq 0,\ 0 \le i \le k-2$ then $\Phi_1(L_k)$ must have the term $D(i, 0, \ldots, 0)$. Hence L_k has the term $D(i+1, 0, \ldots, 0)$ for $i \le k-2$ which contradicts the assumptions. Our claim follows for the first equation.

The second equation is clear since the only k-th order derivative in L_k is $D(k, 0, \ldots, 0)$. We will prove later that $c_{i,j}$ for $1 \le i \le k-2$ are determined by $\{L_0, \ldots, L_{k-1}\}$. $\square$

Proof of Theorem 3.1. Since

$$\begin{aligned} P_k &= \Psi_1(\Phi_1(P_k)) + \{\text{derivatives in } P_k \text{ do not contain } \ \partial_{x_1}^{i_1} \text{ for } i_1 > 0\} \\ &= \Psi_1(\Phi_1(P_k)) + \Psi_2(\Phi_2(P_k))_{i_1=0} \\ &\quad + \{\text{derivatives in } P_k \text{ do not contain } \ \partial_{x_1}^{i_1}\partial_{x_2}^{i_2}, \text{ for } i_1, i_2 > 0\} \\ &= \Psi_1(\Phi_1(P_k)) + \Psi_2(\Phi_2(P_k))_{i_1=0} + \cdots + \Psi_s(\Phi_s(P_k))_{i_1=i_2=\cdots=i_{s-1}=0}, \end{aligned}$$

we prove the theorem inductively by showing that

$$\Phi_1(P_k) = L_{k-1},\ \Phi_j(P_k) = a_{2,j}L_{k-2} + \cdots + a_{k-1,j}L_1,\ 2 \le j \le s. \tag{7}$$

Therefore, formulas (4) and (5) are correct by setting $Q_j = \Phi_j(P_k),\ 1 \le j \le s$.

- For $k = 2$, it is clear that $P_2 = D(2, 0, \ldots, 0)$ and (7) is correct.
- For $k = 3$, suppose $L_3 = P_3 + a_{3,2}D(0, 1, 0, \ldots, 0) + \cdots + a_{3,s}D(0, \ldots, 1)$, where P_3 consists of derivatives of order at least two. By formula (6),

  $$\Phi_1(P_3) = \Phi_1(L_3) = L_2,\ \Phi_j(P_3) = c_{1,j}L_1,\ 2 \le j \le s.$$

  If $c_{1,j} \neq 0$, then the term $D(1, 0, \ldots, 1, \ldots, 0)$ with 1 at positions 1 and j must appear in P_3, moreover, due to the closedness, the term must be obtained by applying Ψ_1 to $L_2 = D(2, 0, \ldots, 0) + a_{2,2}D(0, 1, 0, \ldots, 0) + \cdots + a_{2,s}D(0, \ldots, 1)$ since L_2 does not include the term $D(1, 0, \ldots, 0)$. Therefore

  $$c_{1,j} = a_{2,j},\ \text{for } 2 \le j \le s,$$

and (7) is correct for $k = 3$.

- For $k > 3$, we assume the formula (7) is correct up to $k-1$. According to (6), it is clear that

$$\Phi_1(P_k) = \Phi_1(L_k) = L_{k-1},\ \Phi_j(P_k) = c_{k-2,j}L_{k-2} + \cdots + c_{1,j}L_1,\ 2 \le j \le s.$$

  Similarly, if $c_{i,j} \neq 0$, then P_k must have a term $c_{i,j}D(i, 0, \ldots, 1, 0, \ldots, 0)$ which has 1 at the position j, for $2 \le j \le s$. Moreover, to retain closedness, this term should come from $\Psi_1(L_{k-1})$ since there is no $D(i, 0, \ldots, 0)$ term in L_{k-1} for $1 \le i \le k-2$. Hence the term $c_{i,j}D(i-1, 0, \ldots, 1, 0, \ldots, 0)$ appears in L_{k-1}. If $i = 1$, then $c_{i,j} = a_{k-1,j} = a_{k-i,j}$, otherwise, it must appear in $\Psi_1(L_{k-2})$ according to (4), which implies that $c_{i,j}D(i-2, 0, \ldots, 1, 0, \ldots, 0)$ should appear in L_{k-2}. In the same way, we can proceed further until L_{k-i} and get

$$c_{i,j} = a_{k-i,j},\ \text{for } 2 \le j \le s.$$

  Therefore, the formula (7) is correct for $\Phi_j(P_k)$, $1 \le j \le s$.

- The differential operator L_k defined by formulas (3, 4, 5) retains closedness and $L_k \in \triangle_{\hat{\mathbf{x}}}^{(k)}(I)$ if and only if the vector $[P_k(f_1)|_{\mathbf{x}=\hat{\mathbf{x}}}, \ldots, P_k(f_t)|_{\mathbf{x}=\hat{\mathbf{x}}}]^T$ can be written as a linear combination of the last $s-1$ linear independent columns of the Jacobian matrix $J(\hat{\mathbf{x}})$. The values for the parameters $a_{k,j}$, $j = 2, \ldots, s$ can be determined if the linear combination does exist. Otherwise, we are finished and the multiplicity of the root $\hat{\mathbf{x}}$ is k.

□

4 Algorithms for Computing a Basis of the Local Dual Space

The routine MultiplicityStructureBreadthOneSymbolic below takes as input exact polynomials $F = \{f_1, \ldots, f_t\}$ which generate an ideal I, an exact isolated solution $\hat{\mathbf{x}}$ and the Jacobian matrix of F evaluated at $\hat{\mathbf{x}}$ has corank one, and returns the multiplicity μ and a closed basis $L = \{L_0, \ldots, L_{\mu-1}\}$ of the local dual space of I at $\hat{\mathbf{x}}$.

Algorithm 1 MultiplicityStructureBreadthOneSymbolic

Input: *An isolated singular solution $\hat{\mathbf{x}}$ of a polynomial system $F = \{f_1, \ldots, f_t\}$, and the Jacobian matrix of F evaluated at $\hat{\mathbf{x}}$ has corank one, $L_0 = D(0, 0, \ldots, 0)$, $L_1 = D(1, 0, \ldots, 0) \in \triangle_{\hat{\mathbf{x}}}^{(1)}(I)$.*

Output: *A closed basis $L = \{L_0, \ldots, L_{\mu-1}\}$ of the local dual space of I at $\hat{\mathbf{x}}$ and the multiplicity μ.*

(1) Set $k = 2$ and $P_2 = D(2, 0, \ldots, 0)$. Compute the LU decomposition of N which consists of the last $s-1$ columns of $J(\hat{\mathbf{x}})$. Suppose $N = L \cdot U$.

(2) *Compute* $\mathbf{p}_k = [P_k(f_1)|_{\mathbf{x}=\hat{\mathbf{x}}}, \ldots, P_k(f_t)|_{\mathbf{x}=\hat{\mathbf{x}}}]^T$. *If the triangular system* $L \cdot \mathbf{b}_k = -\mathbf{p}_k$ *is solvable then solve the triangular system* $U \cdot \mathbf{a}_k = \mathbf{b}_k$ *to get* $\mathbf{a}_k = [a_{k,2}, \ldots, a_{k,s}]^T$, *set* $L_k = P_k + a_{k,2}D(0,1,0,\ldots,0) + \cdots + a_{k,s}D(0,\ldots,0,1)$, *and go to Step (3). Otherwise, go to Step (4).*

(3) *Set* $k := k+1$, $P_k = \Psi_1(L_{k-1}) + \Psi_2((Q_2)_{i_1=0}) + \cdots + \Psi_s((Q_s)_{i_1=i_2=\cdots=i_{s-1}=0})$, *where* $Q_j = a_{2,j}L_{k-2} + \cdots + a_{k-1,j}L_1$, *for* $2 \leq j \leq s$, *and go back to Step (2).*

(4) *The algorithm returns* $\{L_0, L_1, \ldots, L_{\mu-1}\}$ *as a basis of the local dual space of* I *at* $\hat{\mathbf{x}}$ *and the multiplicity* $\mu = k$.

Remark 4.1 *If* L_1 *is not* $D(1,0,\ldots,0)$, *we compute a null vector of* $F'(\hat{\mathbf{x}})$, *denoted by* $\mathbf{r}_1$, *and then form a regular matrix* $R = [\mathbf{r}_1, \ldots, \mathbf{r}_s]$. *By mapping* $\mathbf{x}$ *to* $R\mathbf{z}$, *we generate a new system* $H(\mathbf{z}) = F(R\mathbf{z})$, *and apply* MultiplicityStructureBreadthOneSymbolic *to* H *and* $\hat{\mathbf{z}} = R^{-1}\hat{\mathbf{x}}$ *to get a closed basis. We map it back to a closed basis of* $\triangle_{\hat{\mathbf{x}}}(I)$ *by the following formula:*

$$
\begin{aligned}
D(\alpha) &= \frac{1}{\alpha_1! \cdots \alpha_s!} \partial z_1^{\alpha_1} \cdots \partial z_s^{\alpha_s} \\
&= \frac{1}{\alpha_1! \cdots \alpha_s!} \partial (\mathbf{r}_1^T \cdot \mathbf{x})^{\alpha_1} \cdots \partial (\mathbf{r}_s^T \cdot \mathbf{x})^{\alpha_s} \\
&= \frac{1}{\alpha_1! \cdots \alpha_s!} \sum_{|\beta|=|\alpha|} c_\beta \cdot \beta_1! \cdots \beta_s! \cdot D(\beta).
\end{aligned}
$$

In Maple implementation of MultiplicityStructureBreadthOneSymbolic, *we associate polynomials with the differential operators and this allows* Ψ_j *to be implemented as multiplication by* x_j. *For example, we store* $L_1 = D(1,0,\ldots,0)$ *as the polynomial* x_1 *and store* $\Psi_j(L_1) = D(1,0,\ldots,0,1,0,\ldots,0)$ *as* x_1x_j.

EXAMPLE 4.1 *(Dayton, 2007) Consider a polynomial system*

$$F = \{2x^2 - x - x^3 + z^3,\ x - y - x^2 + xy + z^2, xy^2z - x^2z - y^2z + x^3z.\}$$

The system F *has* $(0,0,0)$ *as a* 5*-fold isolated solution, and there are also two other simple isolated zeros but the ideal* I *defined by polynomials in* F *is not zero dimensional since the entire line* $\{z = 0, x = 1\}$ *is a solution of* F *(Dayton, 2007).*

Set $\hat{\mathbf{x}} = [0,0,0]^T$ and $L_0 = D(0,0,0)$. The Jacobian matrix of F evaluated at $\hat{\mathbf{x}}$ is

$$J(\hat{\mathbf{x}}) = \begin{bmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ which is annihilated by } \mathbf{r}_1 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

We complete this column by $\mathbf{r}_2 = [0, 1, 0]^T$, $\mathbf{r}_3 = [1, 0, 0]^T$ to form a regular 3×3-matrix R and generate a new polynomial system $H(\mathbf{z}) = F(R\mathbf{z})$:

$$H = \{2z^2 - z - z^3 + x^3,\ z - y - z^2 + yz + x^2,\ xy^2z - xz^2 - xy^2 + xz^3\}.$$

The Jacobian matrix of H evaluated at $\hat{\mathbf{z}} = R^{-1}\hat{\mathbf{x}}$ is

$$J(\hat{\mathbf{z}}) = \begin{bmatrix} 0 & 0 & -1 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Initialize $L_1 = D(1, 0, 0)$, $P_2 = D(2, 0, 0)$, then we get $\mathbf{p}_2 = [0, 1, 0]^T$. Solving

$$N \begin{bmatrix} a_{2,2} \\ a_{2,3} \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ -1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} a_{2,2} \\ a_{2,3} \end{bmatrix} = - \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

gives $a_{2,2} = 1$, $a_{2,3} = 0$. Hence

$$L_2 = D(2, 0, 0) + D(0, 1, 0).$$

From the data above, iteration $k = 3$ proceeds

$$\begin{aligned} Q_1 &= L_2, \\ Q_2 &= a_{2,2}(L_1)_{i_1=0} = 0, \\ Q_3 &= a_{2,3}(L_1)_{i_1=0, i_2=0} = 0, \end{aligned}$$

so $P_3 = \Psi_1(L_2) = \Psi_1(D(2, 0, 0) + D(0, 1, 0)) = D(3, 0, 0) + D(1, 1, 0)$, then $\mathbf{p}_3 = [1, 0, 0]^T$. Solving $N[a_{3,2}, a_{3,3}]^T = -\mathbf{p}_3$ gives $a_{3,2} = 1$, $a_{3,3} = 1$. Hence

$$L_3 = D(3, 0, 0) + D(1, 1, 0) + D(0, 1, 0) + D(0, 0, 1).$$

Now we continue with $k = 4$ to obtain

$$L_4 = D(4, 0, 0) + D(2, 1, 0) + D(1, 1, 0) + D(1, 0, 1) + D(0, 2, 0).$$

For $k = 5$, we have

$$\Phi_1(P_5) = L_4,\ \Phi_2(P_5) = L_3 + L_2,\ \Phi_3(P_5) = L_2.$$

Hence

$$P_5 = \Psi_1(L_4) + 2D(0, 2, 0) + D(0, 1, 1),$$

and $\mathbf{p}_5 = [0, 0, -1]^T$. The fifth order differential operator consistent with closedness is

$$L_5 = P_5 + a_{5,2}D(0, 1, 0) + a_{5,3}D(0, 0, 1).$$

Since the last entry of $\mathbf{p}_5$ is nonzero, there are no parameters $a_{5,2}, a_{5,3}$ exist such that L_5 is consistent with H. So that we transform $\{L_0, \ldots, L_4\}$ back to a basis of the local dual space of I at $\hat{\mathbf{x}}$:

$$
\begin{aligned}
&L_0 = D(0,0,0),\ L_1 = D(0,0,1),\ L_2 = D(0,0,2) + D(0,1,0),\\
&L_3 = D(0,0,3) + D(0,1,1) + D(0,1,0) + D(1,0,0),\\
&L_4 = D(0,0,4) + D(0,1,2) + D(0,1,1) + D(1,0,1) + D(0,2,0).
\end{aligned}
$$

Notice that the matrix R only maps variables $[x, y, z]$ to $[z, y, x]$.

If I and $\hat{\mathbf{x}}$ are only known approximately, in order to compute an approximate closed basis of $\triangle_{\hat{\mathbf{x}}}(I)$, for ensuring the numerical stability, we need to add a free parameter to P_k and solve the resulted linear system using the singular value decomposition or LU decomposition with pivoting.

Algorithm 2 MultiplicityStructureBreadthOneNumeric

Input: *An isolated singular solution* $\hat{\mathbf{x}}$ *of a polynomial system* $F = \{f_1, \ldots, f_t\}$*, and the Jacobian matrix of* F *evaluated at* $\hat{\mathbf{x}}$ *has corank one with respect to a given tolerance* τ*, an approximate basis* $L_0 = D(0,0,\ldots,0)$*,* $L_1 = D(1,0,\ldots,0)$ *of* $\triangle^{(1)}_{\hat{\mathbf{x}}}(I)$*.*

Output: *A closed approximate basis* $L = \{L_0, \ldots, L_{\mu-1}\}$ *of the local dual space of* I *at* $\hat{\mathbf{x}}$ *and the multiplicity* μ*.*

(1) Set $k = 2$*,* $P_2 = D(2,0,\ldots,0)$*, and* N *consists of the last* $s-1$ *columns of* $J(\hat{\mathbf{x}})$*.*

(2) Compute $\mathbf{p}_k = [P_k(f_1)|_{\mathbf{x}=\hat{\mathbf{x}}}, \ldots, P_k(f_t)|_{\mathbf{x}=\hat{\mathbf{x}}}]^T$*. For the given tolerance* τ*, if the linear system* $[\mathbf{p}_k, N] \cdot \mathbf{a}_k = 0$ *is solvable, we get* $\mathbf{a}_k = [a_{k,1}, \ldots, a_{k,s}]^T$*, set* $L_k = a_{k,1}P_k + a_{k,2}D(0,1,0,\ldots,0) + \cdots + a_{k,s}D(0,\ldots,0,1)$*, and go to Step (3). Otherwise, go to Step (4).*

(3) Set $k := k+1$*,* $P_k = \Psi_1(L_{k-1}) + \Psi_2((Q_2)_{i_1=0}) + \cdots + \Psi_s((Q_s)_{i_1=i_2=\cdots=i_{s-1}=0})$*, where* $Q_j = \frac{b_{k-2,j}}{l_{k-2}}L_{k-2} + \cdots + \frac{b_{1,j}}{l_1}L_1$*. For* $1 \le i \le k-2$ *and* $2 \le j \le s$*,* $b_{i,j}$ *is the coefficient of* $D(i,0,\ldots,0,1,0,\ldots,0)$ *in* $\Psi_1(L_{k-1})$*, which has 1 at the position* j*, and* l_i *is the coefficient of* $D(i,0,\ldots,0)$ *in* L_i*. Go back to Step (2).*

(4) The algorithm returns $\{L_0, L_1, \ldots, L_{\mu-1}\}$ *as an approximate basis of the local dual space of* I *at* $\hat{\mathbf{x}}$ *and the multiplicity* $\mu = k$*.*

Remark 4.2 *In order to show the correctness of the algorithm* MultiplicityStructureBreadthOneNumeric*, we need to check whether* Q_j *in Step (3) is defined properly. Suppose* $D(i,0,\ldots,0,1,0,\ldots,0)$ *is a term in* $\Psi_1(L_{k-1})$ *which has 1 at the position* j *for* $1 \le i \le k-2$ *and* $2 \le j \le s$*, then* $D(i,0,\ldots,0)$ *must be a term in* $\Phi_j(P_k)$ *with the same coefficient, which is* $b_{i,j}$*. On the other hand,* *by the formula (6) and lemma 3.3, we have*

$$\Phi_j(P_k) = c_{k-2,j}L_{k-2} + \cdots + c_{1,j}L_1,\ 2 \le j \le s.$$

Hence, the coefficient of $D(i, 0, \ldots, 0)$ in $\Phi_j(P_k)$ is $c_{i,j} \cdot l_i$. Therefore, from

$$b_{i,j} = c_{i,j} \cdot l_i,$$

we derive that $c_{i,j} = \frac{b_{i,j}}{l_i}$, for $1 \le i \le k-2$ and $2 \le j \le s$.

In Step (2), suppose $\mathbf{a}_k = [a_{k,1}, \ldots, a_{k,s}]^T$ is a null vector of $[\mathbf{p}_k, N]$ with respect to the give tolerance τ, then we have

$$|L_k(f_i)|_{\mathbf{x}=\hat{\mathbf{x}}}| \le \tau,\ \text{for } 0 \le k \le \mu - 1 \text{ and } 1 \le i \le t.$$

Moreover, according to our construction, all these computed L_k, $0 \le k \le \mu - 1$ satisfy the closedness condition, hence, $\{L_0, L_1, \ldots, L_{\mu-1}\}$ is a closed approximate basis of the local dual space of I at $\hat{\mathbf{x}}$.

EXAMPLE 4.2 *(Dayton and Zeng, 2005) Consider the polynomial system*

$$\begin{aligned} F = \{&14\,x + 33\,y - 3\,\sqrt{5}\left(x^2 + 4\,xy + 4\,y^2 + 2\right) + \sqrt{7} + x^3 + 6\,x^2y \\ &+12\,xy^2 + 8\,y^3, 41\,x - 18\,y - \sqrt{5} + 8\,x^3 - 12\,x^2y + 6\,xy^2 - y^3 \\ &+3\,\sqrt{7}\left(4\,xy - 4\,x^2 - y^2 - 2\right)\}. \end{aligned}$$

The system F has $(\frac{2\sqrt{7}}{5} + \frac{\sqrt{5}}{5}, -\frac{\sqrt{7}}{5} + \frac{2\sqrt{5}}{5})$ as a 5-fold isolated solution.

Unlike algorithms based on Gröbner basis, we can use MultiplicityStructureBreadthOneSymbolic to compute an exact dual basis of F at $\hat{\mathbf{x}}$, despite of irrational numbers $\sqrt{5}, \sqrt{7}$ in F and $\hat{\mathbf{x}}$. In order to comparing with MultiplicityStructureBreadthOneNumeric, we normalize the differential operators with respect to the highest order derivative in x, and obtain:

$$\begin{aligned} &L_0 = D(0,0),\ L_1 = D(1,0) + \frac{1}{3}D(0,1), \\ &L_2 = D(2,0) + \frac{1}{3}D(1,1) + \frac{1}{9}D(0,2), \\ &L_3 = D(3,0) + \frac{1}{3}D(2,1) + \frac{1}{9}D(1,2) + \frac{1}{27}D(0,3) + \frac{25}{54}D(1,0) - \frac{25}{18}D(0,1), \\ &L_4 = D(4,0) + \frac{1}{3}D(3,1) + \frac{1}{9}D(2,2) + \frac{1}{27}D(1,3) + \frac{1}{81}D(0,4) + \frac{25}{27}D(2,0) \\ &\qquad - \frac{100}{81}D(1,1) - \frac{25}{27}D(0,2). \end{aligned}$$

In (Dayton and Zeng, 2005), the coefficients of F and $\hat{\mathbf{x}}$ are rounded to five digits. Hence, choosing tolerance $\tau = 0.002$, we apply MultiplicityStructureBreadthOneNumeric to the rounded system and the approximate singular root.

After normalizing and cutting off coefficients with absolute values less than τ, we obtain an approximate basis:

$$
\begin{aligned}
&L_0 = D(0,0),\ \ L_1 = D(1,0) + 0.33341\, D(0,1),\\
&L_2 = D(2,0) + 0.33343\, D(1,1) + 0.11116\, D(0,2),\\
&L_3 = D(3,0) + 0.33343\, D(2,1) + 0.11117\, D(1,2) + 0.037065\, D(0,3)\\
&\qquad +0.46313\, D(1,0) - 1.3891\, D(0,1),\\
&L_4 = D(4,0) + 0.33343\, D(3,1) + 0.11117\, D(2,2) + 0.037065\, D(1,3)\\
&\qquad +0.012358\, D(0,4) + 0.92629\, D(2,0) - 1.2347\, D(1,1) - 0.92629\, D(0,2).
\end{aligned}
$$

The values of $L_i(F)|_{\mathbf{x}=\hat{\mathbf{x}}}$ for $0 \le i \le 4$ are

$$
\begin{aligned}
&L_0(F)|_{\mathbf{x}=\hat{\mathbf{x}}} = (-0.00066377, -0.00039331)^T,\\
&L_1(F)|_{\mathbf{x}=\hat{\mathbf{x}}} = (-0.00023342, 0.00023341)^T,\\
&L_2(F)|_{\mathbf{x}=\hat{\mathbf{x}}} = (-0.0000099698, 0.0000099694)^T,\\
&L_3(F)|_{\mathbf{x}=\hat{\mathbf{x}}} = (-0.00060593, 0.00060608)^T,\\
&L_4(F)|_{\mathbf{x}=\hat{\mathbf{x}}} = (0.00080432, -0.00080428)^T.
\end{aligned}
$$

An example of an analytic system. The method introduced in this paper can also be applied to systems of analytic equations, since the construction of the system of linear equations only relies on the existence of the partial derivatives of the analytic system up to the order μ.

EXAMPLE 4.3 *(Dayton et al., 2009, Example 6) Consider the analytic system*

$$F = \{x^2 \sin(y),\ y - z^2,\ z + \sin(x^n)\}.$$

The system F has $(0,0,0)$ as an $2(n+1)$-fold isolated solution.

The Jacobian matrix of F evaluated at $\hat{\mathbf{x}} = [0,0,0]^T$ is:

$$J(\hat{\mathbf{x}}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The rank deficiency of $J(\hat{\mathbf{x}})$ is one and its null vector is $[1,0,0]^T$. Hence, $L_1 = D(1,0,0)$. For $k \ge 2$, in order to compute L_k, we only need to check whether the vector $\mathbf{p}_k = [P_k(f_1)|_{\mathbf{x}=\hat{\mathbf{x}}}, P_k(f_2)|_{\mathbf{x}=\hat{\mathbf{x}}}, P_k(f_3)|_{\mathbf{x}=\hat{\mathbf{x}}}]^T$ can be written as linear combination of the last two columns of $J(\hat{\mathbf{x}})$, which is equivalent to check whether the first entry of $\mathbf{p}_k$ is zero. The dominant cost is the evaluation of $P_k(F)$ at $\hat{\mathbf{x}}$. This can be done very efficiently since each polynomial in F only consists of one or two terms. Therefore, for this example, our algorithm MultiplicityStructureBreadthOneSymbolic is significantly faster and more powerful than the algorithm presented in (Dayton et al., 2009).

n	5	50	100	200	300	400	500
multiplicity	12	102	202	402	602	802	1002
time	0.056	0.608	2.077	9.596	35.415	105.060	232.490

Table 1. Algorithm Performance of Example 4.3

Remark 4.3 *The reviewer pointed out that for this analytic system, the local ring at* $(0,0,0)$ *has basis* $\{x^2y, y-z^2, z+x^n\}$ *and the standard basis* $\{y-z^2, z+x^n, x^{2n+2}\}$ *can be computed by Singular from the algebraic basis in negligible amount of time. From the degree of the variable* x*, we know that the multiplicity of* $(0,0,0)$ *is* $2n+2$.

5 Complexity and Experiments

The complexity of algorithms MultiplicityStructureBreadthOneSymbolic and MultiplicityStructureBreadthOneNumeric is dominated by solving $\mu - 2$ linear systems with size bounded by $t \times s - 1$ or $t \times s$ respectively, and the evaluations of

$$\mathbf{p}_k = [P_k(f_1)|_{\mathbf{x}=\hat{\mathbf{x}}}, \ldots, P_k(f_t)|_{\mathbf{x}=\hat{\mathbf{x}}}]^T.$$

Although we only need to store polynomials and the computed differential conditions during the computation, similar to other any algorithm designed to calculate and store the dual basis in memory, our algorithm suffers too when polynomials or the differential operators are not sparse. The following example is kindly provided by the reviewer.

EXAMPLE 5.1 *Consider a system* $F = \{f_1, \ldots, f_s\}$ *given by*

$$f_i = x_i^3 + x_i^2 - x_{i+1}, \text{ if } i < s,$$
$$f_s = x_s^2$$

with zero $(0,0,\ldots,0)$ *of multiplicity* 2^s.

In the following table, we show the time needed for computing the differential conditions for s from 2 to 6.

For $s = 6$, about 17MB of memory is used to store the differential operators $\{L_0, \ldots, L_{63}\}$ and takes about 3 hours. For $s \geq 7$, we are not able to obtain all differential conditions after running the algorithm for 2 days. A new algorithm has been proposed in (Li, 2011) to deal with this kind of problems efficiently.

s	2	3	4	5	6
multiplicity	4	8	16	32	64
time	0.023	0.059	0.510	23.093	11061.269

Table 2. Algorithm Performance of Example 5.1

EXAMPLE 5.2 *Consider a system* $F = \{f_1, \ldots, f_s\}$ *given by*

$$f_i = x_i^2 + x_i - x_{i+1}, \text{ if } i < s,$$
$$f_s = x_s^3$$

with zero $(0, 0, \ldots, 0)$ *of fixed multiplicity* 3.

s	10	20	40	100	200
time	0.071	0.166	1.126	21.528	270.735

Table 3. Algorithm Performance of Example 5.2

For this example, the computational time only increases almost cubically with respect to the number of variables since the polynomials have very few terms and the multiplicity is fixed.

The Maple code and all test results, including examples from the PHCpack demos, are available at `http://www.mmrc.iss.ac.cn/~lzhi/Research/hybrid/`.

6 Conclusion

The multiplicity structure of a singular solution has been studied extensively in (Dayton et al., 2009; Zeng, 2009; Pope and Szanto, 2009; Wu and Zhi, 2008; Damiano et al., 2007; Dayton, 2007; Bates et al., 2006; Dayton and Zeng, 2005; Stetter, 2004; Möller and Tenberg, 2001; Kobayashi et al., 1998; Marinari et al., 1996; Mourrain, 1996; Möller and Stetter, 1995; Marinari et al., 1995). In this paper, we present an algorithm MultiplicityStructureBreadthOneSymbolic based on Stetter's strategies (Stetter, 2004) for computing a closed basis of the local dual space of $I = (f_1, \ldots, f_t)$ at $\hat{\mathbf{x}}$ efficiently in the breadth one case. The number of parameters used in computing each order of the differential condition is $s - 1$, which does not increase along with the multiplicity. The algorithm has also been extended to deal with approximately known systems and multiple roots. We are going to investigate the minimum number of parameters needed in computing a closed basis for $\triangle_{\hat{\mathbf{x}}}(I)$ if the breadth is not one.

It is still a challenge problem to compute the multiple solutions of polynomial systems accurately. Various methods have been proposed for refining an approximate singular solution to high accuracy (Wu and Zhi, 2008; Leykin et al., 2006a,b; Giusti et al., 2007; Lecerf, 2002; Corless et al., 1997; Ojika, 1987; Ojika et al., 1983). The breadth one case root refinement has been studied in (Dayton et al., 2009; Dayton and Zeng, 2005; Giusti et al., 2007). We have started to investigate how to apply the strategies in our paper to reduce the matrices appeared in the (Wu and Zhi, 2008; Dayton and Zeng, 2005) to obtain a more efficient algorithm for refining an approximately known multiple root for this special case.

Acknowledgments

We greatly thank Zhonggang Zeng for fruitful discussions during his visit to our lab in July, 2009. The authors wish to thank the reviewers for valuable suggestions and comments that have improved the presentation of the paper.

References

Apostol, Tom M. *Mathematical Analysis.* Addison Wesley Publishing Company, 1974.

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*, 22:475–489, 2006.

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

Corless, R., Gianni, P., and Trager, B. A reordered Schur factorization method for zero-dimensional polynomial systems with multiple roots. In Küchlin, editor, *Proc. 1997 Internat. Symp. Symbolic Algebraic Comput. ISSAC'97*, pages 133–140, New York, 1997. ACM Press. ISBN 0-89791-875-4.

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

Dayton, B. Numerical local rings and local solutions of nonlinear systems. In Verschelde, Jan and Watt, Stephen M., editors, *SNC'07 Proc. 2007 Internat. Workshop on Symbolic-Numeric Comput.*, pages 79–86, New York, N. Y., 2007. ACM Press. ISBN 978-1-59593-744-5.

Dayton, B., Li, T., and Zeng, Z. Multiple zeros of nonlinear systems. `http://orion.neiu.edu/~zzeng/Papers/MultipleZeros.pdf`, 2009.

Dayton, B. and Zeng, Z. Computing the multiplicity structure in solving polynomial systems. In Kauers, Manuel, editor, *ISSAC'05 Proc. 2005 Internat. Symp. Symbolic Algebraic Comput.*, pages 116–123, New York, N. Y., 2005. ACM Press. ISBN 1-59593-095-7.

Giusti, M., Lecerf, G., Salvy, B., and Yakoubsohn, J.-C. On location and approximation of clusters of zeros: Case of embedding dimension one. *Found. Comput. Math.*, 7(1):1–58, 2007. ISSN 1615-3375.

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

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

Leykin, Anton, Verschelde, Jan, and Zhao, Ailing. Newton's method with deflation for isolated singularities of polynomial systems. *Theor. Comput. Sci.*, 359(1):111–122, 2006a. ISSN 0304-3975.

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

Li, N. An improved method for evaluating Max Noether conditions: case of breadth one. in Maza, M. M., editor, *SNC'11 Proc. 2011 Internat. Workshop on Symbolic-Numeric Comput.*, pages 102–103, New York, N. Y., 2011. ACM Press.

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

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

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

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

Mourrain, B. Isolated points, duality and residues. *J. of Pure and Applied Algebra*, 117 & 118:469–493, 1996.

Ojika, T. Modified deflation algorithm for the solution of singular problems. *J. Math. Anal. Appl.*, 123:199–221, 1987.

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

Pope, Scott R. and Szanto, Agnes. Nearest multivariate system with given root multiplicities. *J. Symbolic Comput.*, 44(6):606–625, 2009. ISSN 0747-7171.

Reid, G., Tang, J., and Zhi, L. A complete symbolic-numeric linear method for camera pose determination. In Sendra, J., editor, *Proc. 2003 Internat. Symp. Symbolic Algebraic Comput. ISSAC'03*, pages 215–223, New York, 2003. ACM Press. ISBN 1-58113-641-2.

Stanley, R.P. Hibert function of graded algebras. *Advances in Math.*, 28: 57–83, 1973.

Stetter, H. *Numerical Polynomial Algebra*. SIAM, Philadelphia, 2004.

Wu, X. and Zhi, L. Computing the multiplicity structure from geometric involutive form. In Jeffrey, David, editor, *Proc. 2008 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'08)*, pages 325–332, New York, N. Y., 2008. ACM Press. ISBN 978-1-59593-904-3.

Zeng, Z. The closedness subspace method for computing the multiplicity structure of a polynomial system. http://orion.neiu.edu/~zzeng/Papers/csdual.pdf, 2009.

Zhi, L. and Reid, G. Solving nonlinear polynomial system via symbolic-numeric elimination method. In Faugére, J. and Rouillier, F., editors, *Proc. International conference on polynomial system solving*, pages 50–53, 2004. Full version of the paper in J. Symoblic Comput. 44(3), 280-291.