NUMERICAL DECOMPOSITION OF AFFINE ALGEBRAIC VARIETIES

SHAWKI AL-RASHED  GERHARD PFISTER
RASHED@MATHEMATIK.UNI-KL.DE  PFISTER@MATHEMATIK.UNI-KL.DE

DEPARTMENT OF MATHEMATICS,
UNIVERSITY OF KAISERSLAUTERN,
ERWIN-SCHRÖDINGER-STR.,
67663 KAISERSLAUTERN,
GERMANY

Abstract. An irreducible algebraic decomposition \( \bigcup_{i=0}^{d_i} X_i = \bigcup_{i=0}^{d_i} (\bigcup_{j=1}^{d_j} X_{ij}) \) of an affine algebraic variety \( X \) can be represented as an union of finite disjoint sets \( \bigcup_{i=0}^{d_i} W_i = \bigcup_{i=0}^{d_i} (\bigcup_{j=1}^{d_j} W_{ij}) \) called numerical irreducible decomposition (cf. [14], [15], [17], [18], [21], [22], [23]). \( W_i \) corresponds to a pure \( i \)-dimensional \( X_i \), and \( W_{ij} \) presents an \( i \)-dimensional irreducible component \( X_{ij} \). Modifying this concepts by using partially Gröbner bases, local dimension, and the “Zero Sum Relation” we present in this paper an implementation in SINGULAR to compute the numerical irreducible decomposition. We will give some examples and timings, which show that the modified algorithms are more efficient if the number of variables is not too large. For a large number of variables BERTINI is more efficient. Note that each step of the numerical decomposition is parallelizable. For our comparisons we did not use the parallel version of BERTINI.

Keyword: Witness point set, Homotopy function, Gröbner basis, Local dimension, Monodromy action, Zero Sum Relation.

1. Introduction

Given a system of \( n \) polynomials in \( \mathbb{C}^N \),

\[
\begin{pmatrix}
  f_1(x_1, \ldots, x_N) \\
  \vdots \\
  f_n(x_1, \ldots, x_N)
\end{pmatrix}
\]

Let \( X = \text{V}(f) \) be the algebraic variety defined by the system above. \( X \) has an unique algebraic decomposition into \( d \) pure \( i \)-dimensional components \( X_i, X = \bigcup_{i=0}^{d_i} X_i \). Where \( X_i = \bigcup_{j=1}^{d_j} X_{ij} \) is the union of \( d_i \) \( i \)-dimensional irreducible components, \( d_0, d_1, \ldots, d_d \) positive integers.

The numerical irreducible decomposition (cf. [15], [17], [18], [19], [22]) is given as the union \( W = \bigcup_{i=0}^{d_i} W_i = \bigcup_{i=0}^{d_i} (\bigcup_{j=1}^{d_j} W_{ij}) \). The \( W_i \) are called i-Witness point sets and are given as an intersection of the pure i-dimensional component \( X_i \) of \( X \) with a generic linear space \( L \) in \( \mathbb{C}^N \) of dimension \( N-i \), the \( W_{ij} \) are called the irreducible
witness point sets presenting the irreducible components \(X_{ij}\) of dimension \(i\) with the following properties:

- \(W_{ij}\) consists of a finite number of points contained in \(X_{ij}\).
- \(\#(W_{ij}) = \deg(X_{ij})\) for \(i \neq 0\).
- \(W_{ij} \cap W_{il} = \emptyset\) for \(j \neq l\).

Computing of the numerical irreducible decomposition uses numerical polynomial homotopy continuation methods (cf. [24], [25]). This requires that the number \(n\) of a given polynomial system has to be equal to the number \(N\) of the variables. Therefore we reduce the polynomial system which defines \(X\) to a square system of \(N\) polynomials in \(N\) variables (cf. [15], [17], [22]). Numerical irreducible decomposition (cf. [15], [17], [18], [19], [22]) is proceeded in three steps:

1. The first step reduces the polynomial system to a system of \(N\) polynomials in \(N\) variables and computes a finite set \(\hat{W}_i\) called witness point super set for a pure \(i\)-dimensional component \(X_i\) for \(i=0, \ldots, N-1\). \(\hat{W}_i\) consists of points on \(X_i\) and \(J_i\) a set of points on components of larger dimension the so-called Junk point set (cf. [15], [22]).
2. The second step removes the points of \(J_i\) from \(\hat{W}_i\) to obtain a subset \(W_i\) of the pure \(i\)-dimensional component \(X_i\) (cf. [22]).
3. The third step breakup \(W_i\) into irreducible witness point sets representing the \(i\)-dimensional irreducible components of \(X\) using two algorithms. The first algorithm finds points on the same irreducible component in the Witness point set connected by path tracking techniques applying the idea of monodromy. The second algorithm computes a linear trace for each component, which certifies the decomposition (cf. [14], [21]).

In the second section we present a modified algorithm to compute \(\hat{W}_i\) using the cascade algorithm (cf. [15], [17], [22]) and Gröbner bases in the zero-dimensional case. Then we use the homotopy function (cf. [22]), local dimension and Gröbner bases in the zero-dimensional case to remove Junk points from \(\hat{W}_i\) to obtain the i-Witness point set \(W_i\). This is explained in third section. In the fourth section we explain how to use the “Zero Sum Relation” and monodromy action on the algebraic variety to breakup \(W_i\) into irreducible witness point sets. In the fifth section some examples are tested and timings are given on the basis of our SINGULAR implementation.

### 2. Witness Point Super Set

**Definition 2.1.** Let \(Z=V(f)\) be an affine algebraic variety in \(\mathbb{C}^N\) of dimension \(d\), and \(X\) be a pure \(i\)-dimensional component of \(Z\). Let \(L_i\) be a generic linear space in \(\mathbb{C}^N\) of dimension \(N-i\). A set \(\hat{W}_i \subset \mathbb{C}^N\) is called \(i\)-Witness Point Super Set for \(X\) if it has the following properties:

- \(\hat{W}_i\) is a finite set of points.
- \(X \cap L_i \subset \hat{W}_i \subset Z \cap L_i\).

The union \(\hat{W}\) of all \(i\)-Witness Point Super Sets is called a Witness Point Super Set for \(Z\).

In (cf. [15], [17], [22]) the cascade algorithm is used to compute \(\hat{W}_i\). It starts with \(i=N-1\) to compute the Witness point Super Sets \(\hat{W}_i\). It needs to define a start
system \( G(x) = 0 \) for the homotopy continuation method (cf. [24], [25]) and needs to know its solutions. We use a Gröbner basis to compute the dimension \( d \) of \( Z \), then use the cascade algorithm (cf. [15], [17], [22]) which starts with \( i = d - 1 \). We show below that we do not need to define a start system.

**Algorithm 1 WitnessPointSuperSet**

**Input:** \( F_1, ..., F_n \in \mathbb{C}[x_1, ..., x_N] \).

**Output:** \( \{f_1, ..., f_N\}, \{\hat{W}_r, ..., \hat{W}_d\}, L \). \( \{f_1, ..., f_N\} \) square system, \( \hat{W}_i \) Witness point super sets corresponding to a pure \( i \)-dimensional component of \( V(f_1, ..., f_N) \), \( L \) a set of linear polynomials defining a linear space of dimension \( N - d \).

\[ f = \{f_1, ..., f_N\} \] reduction of \( F = \{F_1, ..., F_n\} \) to a square system (cf. [15], [17], [22]);

\[ d = \dim(V(f_1, ..., f_N)) \], using Gröbner basis (cf. [8], [11]);

\[ r = \text{N-rank}(f) \] the rank of the Jacobian matrix of the system \( f \) at a generic point;

\( L = \{l_1, ..., l_d\} \) a set of \( d \) generic linear polynomials;

if \( d = r \) then

compute \( T_d = V(f_1, ..., f_N, l_1, ..., l_d) \), using triangular sets, (cf. [8], [11]);

set \( \hat{W}_d = \{(x_1, ..., x_N) \mid (x_1, ..., x_N) \in T_d, (x_1, ..., x_N) \in V(F)\} \);

return \( \{f_1, ..., f_N\}, \{\hat{W}_d\}, L \);

else

for \( i = r \) to \( d \) do

if \( i = 0 \) then

\[ \Omega_i(f)(x) = f; \]

else

\[ \Omega_i(f)(x, z_1, ..., z_i) =: \left( \begin{array}{c} f_1(x) + \sum_{j=1}^{i} \lambda_{i1} z_j \\ \vdots \\ f_N(x) + \sum_{j=1}^{i} \lambda_{Nj} z_j \\ l_1 + z_1 \\ \vdots \\ l_i + z_i \end{array} \right) \]

\( \lambda_{kj} \in \mathbb{C} \) generic, \( k = 1, ..., N, j = 1, ..., i; \)

for \( i = d \) to \( r \) do

if \( i = d \) then

compute \( T_i = V(\Omega_i(f)(x, z_1, ..., z_i)) \), using triangular sets, (cf. [8], [11]);

set \( \hat{W}_i = \{(x_1, ..., x_N) \mid (x_1, ..., x_N, 0, ..., 0) \in T_i, (x_1, ..., x_N) \in V(F)\} \);

\( S_i = T_i \setminus \{(x_1, ..., x_N, z_1, ..., z_i) \in T_i \mid z_1 = ... = z_i = 0\} \);

else

compute \( T_i = V(\Omega_i(f)(x, z_1, ..., z_i)) \), using homotopy function with \( \Omega_{i+1}(f)(x, z_1, ..., z_i) \) as start system and \( S_{i+1} \) as start solution set (cf. [15], [17], [22]);

return \( \{f_1, ..., f_N\}, \{\hat{W}_r, ..., \hat{W}_d\}, L \);
3. Computation Witness Point Set

The witness point super set \( \hat{W}_i \) is an union of an i-Witness point set \( W_i \) and a junk point set \( J_i \) (cf. \cite{15,17,22}),

\[
\hat{W}_i = W_i \cup J_i, \text{ where } W_i \subset X_i \text{ and } J_i \subset \cup_{j>i} X_j \text{ for } i = 0, 1, ..., d.
\]

We use Gröbner bases for the 0-dimensional ideal, local dimension and homotopy continuation method to remove the points of \( J_i \) from \( \hat{W}_i \) as follows:

**Algorithm 2 WitnessPointSet**

**Input:** \( \{f_1, ..., f_N\} \subset \mathbb{C}[x_1, ..., x_N], \{\hat{W}_r, ..., \hat{W}_d\} \) a list of witness point superset sets, \( L = \{l_1, ..., l_d\} \) a set of generic linear polynomials (Output of Algorithm 1).

**Output:** \( \{f_1, ..., f_N\}, \{W_r, ..., W_d\}, \{l_1, ..., l_d\} \). \( W_i \) a Witness point set corresponding to a pure i-dimensional component of \( V(f_1, ..., f_N) \).

\( W_d = \hat{W}_d, s_d = \sharp W_d; \)
for \( i = d - 1 \) to \( r \) do
\( W_i = W_i; \)
for each point \( w \in W_i \) do
\( t = \text{dim}_w Z \) for \( Z = V(f_1 - f_1(w), ..., f_N - f_N(w)) \), using Gröbner basis (cf. \cite{8,11});
if \( t > i \) then
\( W_i = W_i \setminus \{w\}; \)
for each point \( w \in W_i \) do
if \( i = 0 \) then
choose \( A \subset \mathbb{C}^{d \times N} \) a generic matrix and a generic \( \epsilon \in \mathbb{C}^N, \|\epsilon\| \) small;
compute \( S = V(\{f_1, ..., f_N, A(x - w)\}), T = V(\{f_1, ..., f_N, A(x - w - \epsilon)\}) \),
using triangular sets (cf. \cite{8,11});
if \( \sharp S = \sharp T \) then
\( W_i = W_i \setminus \{w\}; \)
else
for \( j = i + 1 \) to \( d \) do
choose \( A \subset \mathbb{C}^{j \times N} \) a generic matrix;
if \( j = d \) then
compute \( S = V(\{f_1, ..., f_N, A(x - w)\}) \), using triangular sets, (cf. \cite{8,11});
if \( \sharp S = s_d \) then
\( W_i = W_i \setminus \{w\}; \)
else
compute \( S = V(\{f_1, ..., f_N, A(x - w)\}) \), using homotopy function with start system \( \{f_1, ..., f_N, l_1, ..., l_j\} \) and start solution \( W_j \) (cf. \cite{22});
if \( w \in S \) then
\( W_i = W_i \setminus \{w\}; \)
return \( \{f_1, ..., f_N\}, \{W_r, ..., W_d\}, L \);

---

\( t = \text{dim}_w Z \) is less or equal to the local dimension of a point \( v \in V(f_1, ..., f_N) \) (cf. \cite{9}) with \( \|w - v\| \) small.
4. Partition Witness Point Sets

In this section we show that the monodromy action on an algebraic variety $Z$ and the Zero Sum Relation are sufficient to find the breakup of the $k$-Witness point set $W_k$ into irreducible $k$-Witness point sets. We present here a modified version of the algorithms described in (cf. [14], [21]).

Let $Z$ be a pure $k$-dimensional algebraic variety in $\mathbb{C}^N$, and $Z = \bigcup_{i=1}^r Z_i$ be the irreducible decomposition of $Z$. Let $\pi : \mathbb{C}^N \to \mathbb{C}^k$ be a generic projection and let $l \subset \mathbb{C}^k$ be a general line.

Set

- $W_l := \pi^{-1}(l) \cap Z$ a set of $r$ different curves in $\mathbb{C}^N$.
- $U$ is a non-empty open subset of $l$ consisting of all points $x \in l$ with $\pi^{-1}(x)$ transverse to $Z$.
- $W := \pi^{-1}(x) \cap Z$ for a generic element $x \in U$, and $V_x := V$ a subset of $W$.
- $W_i := \pi^{-1}(x) \cap Z_i$ for an irreducible $k$-dimensional component $Z_i$ of $Z$.
- $\lambda : \mathbb{C}^N \to \mathbb{C}$ a linear function one-to-one on $W$.

Define the function $s : U \to \mathbb{C}$ by

$$s(y) = \sum_{z \in V_y} \lambda(z).$$

Where $V_y$ is a subset of $\pi^{-1}(y) \cap Z$ defined by.

$$V_y := \{z \mid z \text{ on a curve through a point of } V\}.$$

\textbf{Theorem 4.1.} Let $l$, $U$, $W_i$ for $i=1,\ldots,r$, and the functions $\lambda$, $s$ be as above. If the function $s$ is continuous and $V \cap W_i \neq \emptyset$ for some $i \in \{1,\ldots,r\}$, then $W_i \subseteq V$.

\textbf{Example 4.1.} Before proving the theorem we illustrate it by an example.

- Let $Z$ be the algebraic variety of dimension one in $\mathbb{C}^2$ defined by the polynomial $f(x,y) = (x^2 + y^2 - 5)(x - 2y - 3)$. Let $L_1$ be the linear space of dimension one in $\mathbb{C}^2$ defined by the polynomial $l_1 = x + y - 3$.
- Define a homotopy function :

$$h(t, x(t), y(t)) := \begin{pmatrix} \alpha(t) \\ f(x(t), y(t)) \end{pmatrix},$$

where $\alpha : [0,1] \to \mathbb{R}$ given by $\alpha(t) = (1-t)l_0 + tl_1 = x + y - 2t - 1$.

Let $L_0$ be the 1-dimensional linear space defined by the polynomial $l_0 = x + y - 1$, $L_0 \cap Z \neq \emptyset$. Let $G(N-k,N)$ be the Grassmanian and $R :=$
\{(L_{N-k}, x) \in G(N - k, N) \times \overline{Z} \mid x \in L_{N-k} \cap \overline{Z}\} be the family of the intersections $L_{N-k} \cap \overline{Z}$, $L_{N-k} \subset \mathbb{P}^N$ $k$-dimensional linear spaces and \( \overline{Z} \subset \mathbb{P}^N \) the closure of \( Z \). Let \( p : R \rightarrow G(N - k, N) \) be the canonical projection.

Then with conditions above \( \alpha(t) \) maps a point in \( L_1 \cap \overline{Z} \) to a point in \( L_0 \cap \overline{Z} \) as \( t \) goes from 1 to 0.

**Proof.** (of theorem 4.1) Assume that \( W_i \not\subset V \). Since \( W_i \cap V \neq \emptyset \), then there are \( a, b \in W_i \) such that \( a \) is not in \( V \) and \( b \in V \). Let \( a_1, ..., a_r \) denote the points of the set \( V \setminus \{b\} \). By (Corollary 3.5 in \[14\]) there is a loop \( \lambda \) in the fundamental group \( \pi_1(U, \pi^{-1}(x)) \) with \( \alpha(0) = \alpha(1) \) which takes \( a_j \) to \( a_j \) for all \( j=1, ..., r \), and interchanges \( a \) and \( b \).

Since \( \alpha \) is a continuous loop and \( s : U \rightarrow \mathbb{C} \) is continuous, the composition \( s \circ \alpha : [0, 1] \rightarrow \mathbb{C} \) is continuous and

\[
\lambda(a) + \sum_{j=1}^{r} \lambda(a_j) = \lambda(b) + \sum_{j=1}^{r} \lambda(a_j),
\]

as \( t \) goes from 1 to 0. This implies that \( \lambda(a) = \lambda(b) \). But this contradicts the fact that \( \lambda \) is one-to-one on \( W \). Thus \( W_i \subseteq V \).

**Example 4.2.** Let \( Z \) be a pure 1-dimensional component in \( \mathbb{C}^2 \) defined by the polynomial \( f(x, y) = (y-x)(y-2x)(y-3x) \), and \( Z = Z_1 \cup Z_2 \cup Z_3 \) be the irreducible decomposition. Let \( \pi : \mathbb{C}^2 \rightarrow \mathbb{C} \) be the projection given by \( \pi(x, y) = x \), and \( \lambda : \mathbb{C}^2 \rightarrow \mathbb{C} \), \( \lambda(x, y) = y \).

Note that the restriction of \( \pi \) to \( Z \), \( \pi_Z \) is proper and generically three-to-one with degree 3 equal to the degree of \( Z \). \( \lambda \) is one-to-one on the fiber \( \pi_Z^{-1}(y) = \{(x, x), (x, 2x), (x, 3x)\} \). Let \( L \) be the linear space in the Grassmannian \( G(1,2) \) defined by the linear polynomial \( l(x, y) = x + y - 2 \). \( L \) intersects \( Z \) in the finite set \( W := \{(1,1), (\frac{2}{3}, \frac{4}{3}), (\frac{3}{2}, \frac{5}{2})\} \).

Let \( V := \{(1,1), (\frac{2}{3}, \frac{4}{3})\} \subset W \). The function \( \sum_{v \in V} \lambda(v) \) given by \( \lambda(x, x) + \lambda(x, 2x) = x + 2x = 3x \) is continuous. Then by the theorem above for an irreducible 1-Witness point set \( W_1 \subset V \) it contains \( \{(1,1)\} \).

**Preparation of the algorithm(IrrWitnessPointSet):**
Let \( Z_k = \cup_{i=1}^{r} Z_{k_i} \) be the union of the irreducible \( k \)-dimensional components of the algebraic variety \( Z \) and \( L_k \) be the generic linear space in \( \mathbb{C}^N \) defined by \( k \) linear equations

\[
l_j = c_{j0} + c_{j1}x_1 + ... + c_{jN}x_N.
\]

for \( j=1, ..., k \) and \( i=0,1,...,N \), \( c_{ij} \in \mathbb{C} \).

We use the generic linear space \( L_k \) to define the generic projection \( \pi : \mathbb{C}^N \rightarrow \mathbb{C}^{k+1} \), \( \pi(x_1, ..., x_N) := (z_1, ..., z_{k+1}) \) as follows:

\[
\begin{pmatrix}
x_1 \\
\vdots \\
x_N
\end{pmatrix} \rightarrow
\begin{pmatrix}
z_1 \\
\vdots \\
z_{k+1}
\end{pmatrix} :=
\begin{pmatrix}
c_{11} & c_{12} & \cdots & c_{1N} \\
c_{21} & c_{22} & \cdots & c_{2N} \\
\vdots & \vdots & \ddots & \vdots \\
c_{k1} & c_{k2} & \cdots & c_{kN} \\
p_1 & p_2 & \cdots & p_N
\end{pmatrix}
\begin{pmatrix}
x_1 \\
\vdots \\
x_N
\end{pmatrix},
\]
$p_1, ..., p_N \in \mathbb{C}$ randomly chosen.
Set $\lambda(x_1, ..., x_N) := z_{k+1}$ and $l := z_k \subset \mathbb{C}^k$ as in the theorem above. Let $y := c_0$
 vary and fix the other $c_1, ..., c_{(k-1)0}$.

- Let $L_{k,y}$ be the linear spaces defined by the linear equations $l_1, ..., l_{k-1}$
  above and $l_{k,y} := y + c_k \cdot x_1 + ... + c_N \cdot x_N$.

- For the subset $V_y$ of $W_y := L_{K,y} \cap Z_k$ the function $s$ is given as a function

$$s_y : Z_k \cap L_{k,y} \rightarrow \mathbb{C},$$

$$s_y(x_1, ..., x_N) := \sum_{(x_1, ..., x_N) \in V_y} \lambda(x_1, ..., x_N).$$

It is convenient to test the linearity of $s_y$.

- Since $y = -(c_k \cdot x_1 + ... + c_N \cdot x_N)$, then we can define a function $s$

$$s : \mathbb{C} \rightarrow \mathbb{C}, s(y) := s_y(x_1, ..., x_N) = \sum_{(x_1, ..., x_N) \in V_y} \lambda(x_1, ..., x_N).$$

- $s_y$ is a linear function in $x = (x_1, ..., x_N)$ if and only if $s$ is linear in $y$.
- To test the linearity of $s$, we take three values of $y$ in $\mathbb{C}$, say $a, b, c$.

If there exist $A, B \in \mathbb{C}$ such that:

$$(s(a) = Aa + B, s(b) = Ab + B) \implies s(c) = Ac + B. \quad (4.1)$$

Then $s$ is linear in $y$.

Here $s(a), s(b)$ and $s(c)$ correspond to the subsets $V_a \subset W_a = Z_k \cap L_{k,a}$,
$V_b \subset W_b = Z_k \cap L_{k,b}$ and $V_c \subset W_c = Z_k \cap L_{k,c}$ respectively with $\sharp V_a = \sharp V_b = \sharp V_c = m$.

So far this is the approach which can be found in [14]. Now we give some modifications.

- The condition (4.1) of the linearity above is equivalent to the following equation

$$s(a)(b - c) + s(b)(c - a) + s(c)(a - b) = 0. \quad (4.2)$$

- From Theorem 4.1 we obtain: If $W_{kj} \cap V_a \neq \emptyset$ for some $j \in \{1, ..., r\}$ and
  the condition (4.1) of the linearity above is true, then $W_{kj} \subseteq V_a$.

- Let $Z(y) := \{z = \sum_{t=1}^{N} p_t v_t | v = (v_1, ..., v_N) \in V_y, p = (p_1, ..., p_N) \in \mathbb{C}^N\}$. Then

$$s(y) = \sum_{v \in V_y} \lambda(v) = \sum_{v \in V_y} \left(\sum_{t=1}^{N} p_t v_t\right) = \sum_{z \in Z(y)} z.$$

- To compute the sets $V_b$ and $V_c$ we use the homotopy function as $t$ goes from 1 to 0 using $V_a$ as start set. In particular

$$V_b := ((1 - t) L_{k,b} + t L_{k,a}) \cap Z, V_c := ((1 - t) L_{k,c} + t L_{k,a}) \cap Z, \text{ as } t \text{ goes } 1 \rightarrow 0.$$
From (4.2) we get a condition equivalent to the condition (4.1) of the linearity

\[(b - c) \sum_{i=1}^{m} a_i + (c - a) \sum_{i=1}^{m} b_i + (a - b) \sum_{i=1}^{m} c_i = 0.\]  

(4.3)

The condition (4.3) is called **Zero Sum Relation** (cf. [7]) of a given subset \(V_a \subseteq W\) denoted by \(ZSR(V_a)\).

- The sets \(V_a, V_b\) and \(V_c\) have distinct points and the same cardinality \(m\), then obviously

\[ZSR(V_a) = \sum_{a_i \in V_a} ZSR(\{a_i\}).\]  

(4.4)

where \(ZSR(\{a_i\}) = (b - c)a_i + (c - a)b_i + (a - b)c_i\) is defined as a Zero Sum Relation of a given point in \(V_a\).

---

**Algorithm 3**  
**IRRWITNESSPOINTSET**

**Input:** \(\{f_1, ..., f_N\} \subset \mathbb{C}[x_1, ..., x_N]\), \(\{W_r, ..., W_d\}\), a list of witness point sets, \(L = \{l_1, ..., l_d\}\) a set of generic linear polynomials (Output of Algorithm 2). Where \(W_k = \{w_1, ..., w_{mk}\}\) are witness point sets for a pure \(k\)-dimensional component \(Z_k\) of \(Z = V(f_1, ..., f_N)\), \(k = r, ..., d\).

**Output:** \(\{\{W_{r1}, ..., W_{rt_i}\}, ..., \{W_{d1}, ..., W_{d_{t_k}}\}\}\), \(W_{kr_k}\) irreducible Witness point sets corresponding to \(k\)-dimensional irreducible component \(Z_{kr_k}\) of \(Z_k\).

```plaintext
for k = r to d
    a:=c_k0, define \(L_{ka}\) to be the linear space defined by the subset \(\{l_1, ..., l_k\} \subset L\).
    choose \(b, c \in \mathbb{C}\) generic, define \(L_{kb}, L_{kc}\) as above;
    \(W_a = W_k, W_b = \emptyset, W_c = \emptyset, R = \emptyset\);
    choose \(p_1, ..., p_N \in \mathbb{C}\);
    for i = 1 to \(mk\) do
        compute \(\{v_i\} \subset Z \cap L_{kb}\) and \(\{\hat{v}_i\} \subset Z \cap L_{kc}\) using the homotopy function
        with \(\{f_1, ..., f_N, l_{i-1}, l_{k,a}\}\) as start system and \(\{v_i\}\) as start solution;
        compute the Zero Sum Relation of \(\{v_i\}\):
        \[r_i = (a - b)(\sum_{j=1}^{N} p_j \hat{v}_{ij}) + (b - c)(\sum_{j=1}^{N} p_j w_{ij}) + (c - a)(\sum_{j=1}^{N} p_j v_{ij});\]
        int \(t_k = 0\);
        while \(R \neq \emptyset\) do
            if \(\sum_{t \in T} t = 0\) and \(T\) is a smallest subset\(^3\) of \(R\) then
                \(t_k = t_k + 1;\)
                \(W_{kr_k} \subset W_a\) consists of the points corresponding of the points of \(T\);
                \(R = R \setminus T;\)
            return \(\{\{W_{r1}, ..., W_{rt_i}\}, ..., \{W_{d1}, ..., W_{d_{t_k}}\}\}\);\(^4\)
```

\(^3\)the \(i\) – th point in \(R\) corresponds to the \(i\) – th point in \(W_a\);

\(^4\)smallest subset with respect to the cardinality.
We now illustrate\footnote{Note that the values of $w_i$ are approximate values. The following equalities are therefore to interpret as approximations of the points $w_i$.} interpret as approximations of the points \{w_0, w_1, w_2, w_3\}. $Z_1$ should contain three points $W_1 := \{w_1, w_2, w_3\}$ and the remaining points $W_2 := \{w_4, w_5\}$ are on $Z_2$. The algorithms (cf. \[14\], \[21\]) use the homotopy function at least nine times to breakup $W$ into $W_1$ and $W_2$. We will show below that we do not need more than five times to use the homotopy function to breakup $W$ into $W_1$ and $W_2$.

**Example 4.3.** Let $Z$ be the algebraic variety of dimension two in $\mathbb{C}^3$ defined by the polynomial $f(x, y, z) = (x^3 + z)(x^2 - y)$. Let $L$ be the linear space of dimension one in $\mathbb{C}^3$ defined by the linear equations $l_1 = 4x + 7y + 2z + 6$, $l_2 = 5x + 7y + 3z + 6$. Then $W := L \cap Z = \{w_1, w_2, w_3, w_4, w_5\}$, where

\[
\begin{align*}
w_1 &= (1, -1.1428571429, -1), \\
w_2 &= (0, -0.8571428571, 0), \\
w_3 &= (-1.1428571429 + i \ast 0.9147320339, -0.8163265306 - i \ast 0.2613520097, \ 0.1428571429 - i \ast 0.9147320339), \\
w_4 &= (-1, -0.5714285714, 1), \\
w_5 &= (-1.1428571429 - i \ast 0.9147320339, -0.8163265306 + i \ast 0.2613520097, \ 0.1428571429 + i \ast 0.9147320339).
\end{align*}
\]

We now illustrate Algorithm \ref{Alg:WitnessSet} (IrrWitnessPointSet):

- Use the linear space $L_1$ to define the linear projection $\pi : \mathbb{C}^3 \rightarrow \mathbb{C}^3$ as follows

\[
\pi(x, y, z) := \left( \begin{array}{ccc} 4 & 7 & 2 \\ 5 & 7 & 3 \\ 1 & 2 & 3 \end{array} \right) \left( \begin{array}{c} x \\ y \\ z \end{array} \right) = (4x + 7y + 2z, 5x + 7y + 3z, x + 2y + 3z).
\]

- Define the linear space $L_{1,c}$ of dimension one in $\mathbb{C}^3$ by the linear equations $l_1 = 4x + 7y + 2z + 6$, $l_c = 5x + 7y + 3z + c$, where $c$ is generically chosen in $\mathbb{C}$. Then

\[
\pi_{Z \cap L_{1,c}}(x, y) = (-6, -c, x + 2y + 3z).
\]

- Define the linear function $\lambda : \mathbb{C}^2 \rightarrow \mathbb{C}$ by $\lambda(x, y, z) := x + 2y + 3z$.

- For $a=6$, let $V_a = V_a := \{w_{11} = (1, -1.1428571429, -1) \} \subset W$, $L_{1,a} := L$ the linear space defined by $l_1 = 4x + 7y + 2z + 6$, $l_a = 5x + 7y + 3z + 6$. Then

\[
\{\lambda(a) = \{\sum_{v \in V_a} \lambda(v) = w_{11}[1] + 2(w_{11}[2]) + 3(w_{11}[3])\} = \{-4.2857142858\}.
\]

- Let $b=9$, $L_{1,b}$ the linear space defined by $l_1 = 4x + 7y + 2z + 6$, $l_b = 5x + 7y + 3z + 9$. Compute $V_b := (tL_{1,a} + (1 - t)L_{1,b}) \cap Z = \{w_{12} = (1.671699881657157, -0.4776285376163331, -0.671699881657164)\}$ as $t$ goes from $1$ to $0$, using $V_a$ as a start solution. $Z(b) = \{w_{12}[1] + 2(w_{12}[2]) + 3(w_{12}[3])\} = \{-13.2986568385470012\}$.

- Let $c=63$, $L_{1,c}$ the linear space defined by $l_1 = 4x + 7y + 2z + 6$, $l_c = 5x + 7y + 3z + 63$. Compute $V_c := (tL_{1,a} + (1 - t)L_{1,c}) \cap Z = \{w_{13} = (3.935100643260828, 14.30425695906836, -60.93510064326094)\}$ as $t$ goes
Remark 4.1. The points of a Witness point set are computed approximately by using the homotopy continuation method. Therefore the result of the Zero Sum Relation is only almost zero.
5. Examples and timings with Singular and Bertini

In this section we provide examples with timings of the algorithms WitnessPointSuperSet, WitnessPointSet, and IrrWitnessPointSet implemented in SINGULAR to compute the numerical decomposition of a given algebraic variety defined by a polynomial system and compare them with the results of Bertini. We did not use the parallel features of Bertini.

We tested versions of the implementations in Bertini using the cascade algorithm and using the regenerative cascade algorithm. Timings are conducted by using the 32-bit version of SINGULAR 3-1-1 (cf. [8]) and Bertini 1.2 (cf. [3]) on an Intel® Core(TM)2 Duo CPU P8400 @ 2.26 GHz 2.27 GHz, 4 GB RAM under the Kubuntu Linux operating system.

Let Z be the algebraic variety defined by the following polynomial system:

Example 5.1. (cf. [17]).

\[ f(x, y, z) = \left( \begin{array}{c} (y - x^2)(x^2 + y^2 + z^2 - 1)(x - \frac{1}{2}) \\ (z - x^3)(x^2 + y^2 + z^2 - 1)(y - \frac{1}{2}) \\ (y - x^2)(z - x^3)(x^2 + y^2 + z^2 - 1)(z - \frac{1}{2}) \end{array} \right) \]

Example 5.2. (cf. [22], Example 13.6.4).

\[ f(x, y, z) = \left( \begin{array}{c} x(y^2 - x^3)(x - 1) \\ x(y^2 - x^3)(y - 2)(3x + y) \end{array} \right) \]

Example 5.3.

\[ f(x, y, z) = \left( \begin{array}{c} (x^3 + z)(x^2 - y) \\ (x^3 + y)(x^2 - z) \\ (x^3 + z)(x^3 + y)(z^2 - y) \end{array} \right) \]

Example 5.4.

\[ f(x, y, z) = \left( \begin{array}{c} x(y^2 - x^3)(x - 1) \\ x(3x + y)(y^2 - x^3)(y - 2) \\ x(y^2 - x^3)(x^2 - y) \end{array} \right) \]

Example 5.5.

\[ f(x, y, z) = \left( \begin{array}{c} (x - 1)((x^3 + z) + (x^2 - y)) \\ (x^3 + z)(x^2 - y) \\ (x^3 + z)(x^2 - 1) \end{array} \right) \]

Example 5.6.

\[ f(x, y, z) = \left( \begin{array}{c} (y - x^2)(x^2 + y^2 + z^2 - 1)(x - \frac{1}{2}) + x^5 \\ (z - x^3)(x^2 + y^2 + z^2 - 1)(y - \frac{1}{2}) + y^4 \\ (y - x^2)(z - x^3)(x^2 + y^2 + z^2 - 1)(z - \frac{1}{2}) + z^6 \end{array} \right) \]

Example 5.7.

\[ f(x, y, z) = \left( \begin{array}{c} x(y^2 - x^3)(x - 1) + y^2 \\ x(y^2 - x^3)(y - 2)(3x + y) + x^3 \end{array} \right) \]
Example 5.8.
\[
f(x, y, z) = \left( \begin{array}{c}
(x^3 + z)(x^2 - y) + x^4 \\
(x^3 + y)(x^2 - z) + y^3 \\
(x^3 + z)(x^3 + y)(z^2 - y) + z^5
\end{array} \right)
\]

Example 5.9.
\[
f(x, y, z) = \left( \begin{array}{c}
f_1 = -35688914118603000072x^5 + 1948764938x^4 + 35688914118603000072x^2y^2 - 1948764938xy^2 \\
f_2 = -51052002429375403020x^5y - 170173341312513440 xy^2 + 11692589628x^2y^2 + 3897529876xy^2 + 51052002429375403020x^2y^3 + 170173341312513440xy^4 - 11692589628x^2y^2 - 3897529876xy^3
\end{array} \right)
\]

Example 5.10.
\[
f(x, y, z) = \left( \begin{array}{c}
f_1 = -356737285367005125x^5 - 92300457164036000x^3y + 1121648050080163317x^2z + 290209720279281056yz \\
f_2 = -356737285367005125x^5 + 887060318883271500x^3z + 1121648050080163317x^2z - 2789081819567309964yz \\
f_3 = -356737285367005125x^5 + 887060318883271500x^3z - 887060318883271500x^3yz + 1121648050080163317x^2z - 2789081819567309964yz
\end{array} \right)
\]

Example 5.11.
\[
f(x, y, z) = \left( \begin{array}{c}
f_1 = x^5y^2 + 2x^3y^4 + xy^6 + 2x^3y^2z^2 + 2xy^4z^2 + xy^2z^4 - x^4y^2 - 2x^2y^4 - y^6 - x^5z - 2x^3y^2z - xy^4z - 2x^2y^2z^2 - 2y^4z^2 - 2x^3z^3 - 2xy^2z^3 - y^2z^4 - x^5z - 3x^3y^2 - 3xy^4 + x^3z + 2x^2y^2z + y^4z - 3xy^2z^2 + 2x^2z^3 + y^3z^2 + 3z^3 + 3x^2y^2 - 3x^2z - 3y^2z - 3z^3 - 2y^2 - 2xz + 2z \\
f_2 = x^6y + 2x^4y^3 + x^4y^6 + 2x^4yz^2 + 2x^4y^3z^2 + 2x^2yz^2 + 5x^6 - 10x^4y^2 - 5x^2y^4 - 4x^3y^2 - 2x^2y^3z - x^5y^2 - 10x^2y^2z^2 - 2x^2yz^3 - 2y^4z^3 - 5x^2z^4 - 3x^4y - 3x^2y^3 + 5x^4z + 10x^2y^2z + 5y^4z - 3x^2yz^2 + 10x^2z^3 + 10y^2z^3 + 5z^5 + 15x^4 + 15x^2y^2 + 3x^2yz + 3y^4 + 15x^2z^2 + 3y^2z + 2x^2y - 15x^2z - 15y^2z - 10x^2 - 2yz + 10z \\
f_3 = x^6y^2 + 2x^4y^4 + x^2y^6 + 2x^4y^2z^3 + 2x^2y^2z^3 + 2x^2y^2z - 2x^2y^4z^2 - 17x^2y^2z^2 - y^6z^2 - 2x^4z^2 - 11x^2y^2z^2 - 2y^4z^2 - x^2y^6 + 7x^4z^4 + 18x^2y^2z^2 + 18z^6 + 15x^4z^2 + 27x^2y^2z^3 + 15y^4z^3 + 9x^2z^3 + 9y^2z^3 + 21x^2y^2 + 21x^2y^4 - 4x^4z^2 + 13x^4y^2z^2 - 4y^4z^2 - 11x^2z^4 - 11y^2z^2 - 7z^6 - 21x^4z - 40x^2y^2z - 21y^4z - 24x^2z^3 - 24y^2z^3 - 3z^5 - 14x^2y^2 + 19x^2z^2 + 19y^2z^2 + 21z^3 + 14x^2z + 14y^2z + 2z^3 - 14z^2
\end{array} \right)
\]
Example 5.12.

\[ f(x_1, x_2, x_3, x_4, x_5) = \begin{pmatrix} x_5^2 + x_1 + x_2 + x_3 + x_4 - x_5 - 4 \\ x_5^2 + x_1 + x_2 + x_3 - x_4 + x_5 - 4 \\ x_5^2 + x_1 + x_2 - x_3 + x_4 + x_5 - 4 \\ x_5^2 + x_1 - x_2 + x_3 + x_4 + x_5 - 4 \\ x_1^2 - x_1 + x_2 + x_3 + x_4 + x_5 - 4 \end{pmatrix} \]

Example 5.13.

\[ f(a, b, c, d, e, f, g) = \begin{pmatrix} a^2 + 2de + 2cf + 2bg + a \\ 2ab + e^2 + 2df + 2cg + b \\ b^2 + 2ac + 2ef + 2dg + c \\ 2bc + 2ad + f^2 + 2eg + d \\ c^2 + 2bd + 2ae + 2fg + e \\ 2cd + 2be + 2af + g^2 + f \\ d^2 + 2ce + 2bf + 2ag + g \end{pmatrix} \]

Example 5.14. cyclic 4-roots problem. (cf. [4, 6]).

Example 5.15. cyclic 5-roots problem. (cf. [4, 6]).

Example 5.16. cyclic 6-roots problem. (cf. [4, 6]).

Example 5.17. cyclic 7-roots problem. (cf. [4, 6]).

Example 5.18. cyclic 8-roots problem. (cf. [4, 6]).

Example 5.19.

\[ f(x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{21}, x_{22}, x_{23}, x_{24}, x_{25}, x_{31}, x_{32}, x_{33}, x_{34}, x_{35}) = \begin{pmatrix} -x_{12}x_{21} + x_{11}x_{22} \\ -x_{13}x_{22} + x_{12}x_{23} \\ -x_{14}x_{23} + x_{13}x_{24} \\ -x_{15}x_{24} + x_{14}x_{25} \\ -x_{22}x_{31} + x_{21}x_{32} \\ -x_{23}x_{32} + x_{22}x_{33} \\ -x_{24}x_{33} + x_{23}x_{34} \\ -x_{25}x_{34} + x_{24}x_{35} \end{pmatrix} \]

Table summarizes the results of the timings to compute the numerical decomposition.

Remark 5.1. The timings show that for an increasing number of variables the original method of (cf. [14, 15, 17, 21, 22]) becomes more efficient. One reason is that the computation of triangular sets which is used in SINGULAR for solving polynomial systems is expensive in this case. Therefore the Algorithm 1, Algorithm 2 becomes slow in this situation. This is not true for Algorithm 3. Replacing the solving of polynomial system using triangular sets by homotopy function methods but keeping the computation of the dimension and starting in this dimension is more efficient in a case of a large number of variables.

\(^7\)(re) means using the regenerative cascade algorithm instead of the cascade algorithm
Table 1. Total running times for the computing a numerical decomposition of the examples above

| Example | BERTINI | BERTINI (re) | SINGULAR |
|---------|---------|--------------|----------|
| 5.1     | 134.45s | 39s          | 36.07s   |
| 5.2     | 3.08s   | 2.5s         | 1.49s    |
| 5.3     | 1min 21.28s | 27.4s | 4.02s    |
| 5.4     | 18.56s  | 2.7s         | 1.77s    |
| 5.5     | 15.36s  | 8.6s         | 1.29s    |
| 5.6     | 4min 13s | 15min 2s    | 2min 27s |
| 5.7     | 1.83s   | 1.6s         | 0.39s    |
| 5.8     | 3min 29s | 10min 43s   | 1.69s    |
| 5.9     | 16s     | 7s           | 2s       |
| 5.10    | 2min 57s | 28s          | 2min 35s |
| 5.11    | 44min 56s | 2min 37s   | 4min 3s  |
| 5.12    | 4.73s   | 6s           | 0.37s    |
| 5.13    | 5.84s   | 8s           | 1s       |
| 5.14    | 1.43s   | 4.3s         | 0.79s    |
| 5.15    | 3.54s   | 10s          | 0.57s    |
| 5.16    | 3min 23.26s | 2min 29s | 1.43s    |
| 5.17    | 2h 11min 57s | 32min 17s | stopped after 5h |
| 5.18    | 19h 45min 17s | 6h 45min2s | stopped after 50h |
| 5.19    | 1min 57s | 51s          | stopped after 3h |

REFERENCES

[1] E. L. Allgower and K. George. Numerical Continuation Methods, an Introduction. Springer Series in Comput. Math, Vol. 13, Springer-Verlag, Berlin, Heidelberg, New York, 1990.
[2] E. Arbarello, M. Cornalba, P.A. Griffiths and J. Harris. Geometry of Algebraic Curves, volume I. Volume 267 of Grundlehren Math. Wiss., Springer Verlag, New York, 1985.
[3] D. J. Bates, J. D. Hauenstein, A. J. Sommese and C. W. Wampler. Bertini : Software for Numerical Algebraic Geometry. http://www.nd.edu/~sommese/bertini/.
[4] S. C. Billups, A. P. Morgan and L. T. Watson. Algorithm 652. Hompack: A Suite codes for Globally Convergent Homotopy Algorithms. ACM Transactions on Mathematical Software, Vol. 13. No. 3, September 1987, Pages 281-310.
[5] G. Björck and R. Fröberg. A faster way to count the solutions of inhomogeneous systems of algebraic equations, with applications to cyclic n-roots. J. Symbolic Computation, 12:329–336, 1991.
[6] G. Björck. Functions of modulus one on Zn whose Fourier transforms have constant modulus, and cyclic n-roots. In: J.S. Byrnes and J.F. Byrnes, Editors, Recent Advances in Fourier Analysis and its Applications 315, NATO Adv. Sci. Inst. Ser. C. Math. Phys. Sci., Kluwer (1989), pp. 131-140.
[7] R. M. Corless, A. Galligo, I. S. Kotsireas and S. M. Watt. A Geometric- Numeric Algorithm for Absolute Factorization of Multivariate Polynomials. ISSAC 2002, July 7-10, 2002, Lille, France.
[8] W. Decker, G.-M Greuel, G. Pfister and H. Schönemann. SINGULAR 3-1-1 — A computer algebra system for polynomial computations. http://www.singular.uni-kl.de (2010).
[9] G. Fischer. Complex Analytic Geometry. Lecture Notes in Mathematics 538 (1976).
[10] W. Fulton. Intersection Theory, volume (3) 2 of Ergeb. Math. Grenzgeb. Springer Verlag, Berlin, 1984.
[11] G.-M. Greuel and G.Pfister. A SINGULAR Introduction to Commutative Algebra. Second edition, Springer (2007).
[12] A. P. Morgan and L. T. Watson. *A globally convergent parallel algorithm for zeros of polynomial systems*. Nonlinear Anal. (1989), 13(11), 1339–1350.

[13] A. P. Morgan and A.J. Sommese. *Computing all solutions to polynomial systems using homotopy continuation*. Appl. Math. Comput., 115-138. Errata: Appl. Math. Comput. 51 (1992), p. 209. Nonlinear Anal.

[14] A.J. Sommese, J Verschelde and C.W. Wampler. *Symmetric functions applied to decomposing solution sets of polynomial systems*. Vol. 40, No. 6, pp. 2026-2046. 2002 Society for Industrial and Applied Mathematics.

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

[16] A.J. Sommese, J Verschelde and C.W. Wampler. (2002a). *A method for tracking singular paths with application to the numerical decomposition*. In algebraic geometry (pp. 329-345). Berlin: de Gruyter.

[17] A.J. Sommese, J Verschelde and C.W. Wampler. (2001a). *Numerical decomposition of solution sets of polynomial systems into irreducible components*. SIAM J. Number. Anal., 38(6), 2022-2046. NID

[18] A.J. Sommese, J Verschelde and C.W. Wampler. (2003). *Numerical irreducible decomposition using PHCpack*. In algebra, geometry, and software systems (pp. 109-129). Berlin: Springer.

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

[20] A.J. Sommese, J Verschelde and C.W. Wampler. *Numerical irreducible decomposition using projections from points on components*. In Symbolic Computation: Solving Equations in Algebra, Geometry, and Engineering, volume 286 of contemporary Mathematics, edited by E.L. Green, S. Hosten, B.C. Lauenbacher, and V. Powers, pages 27-51. AMS 2001.

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

[22] A.J. Sommese and C.W. Wampler. *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science*. ISBN 981-256-184-6. Word Scientific Publishing Co. Pte. Ltd. 2005.

[23] A.J. Sommese and C.W. Wampler. *Numerical algebraic geometry*. In the mathematics of numerical analysis. (Park City, UT, 1995), Vol. 32 of lectures in Appl. Math. (pp. 749-763). Providence, RI: Amer. Math. Soc.

[24] J. Verschelde(1996). *Homotopy continuation methods for solving polynomial systems*. PhD thesis, Katholieke Universiteit Leuven.

[25] J. Verschelde(1999). Algorithm 795: PHCpack: A general-purpose solver for polynomial systems by homotopy continuation. ACM Trans. on Math. Software, 25(2), 251-276.