The Vanishing Ideal of a Finite Set of Points with Multiplicity Structures

Na Lei, Xiaopeng Zheng and Yuxue Ren

Abstract Given a finite set of arbitrarily distributed points in affine space with multiplicity structures, we present an algorithm to compute the reduced Gröbner basis of the vanishing ideal under the lexicographic order. We split the problem into several smaller ones which can be solved by induction over variables and then use our new algorithm for intersection of ideals to compute the result of the original problem. The new algorithm for intersection of ideals is mainly based on the Extended Euclidean Algorithm. Our method discloses the essential geometric connection between the relative position of the points with multiplicity structures and the leading monomials of the reduced Gröbner basis of the vanishing ideal.

Keywords Vanishing ideal · Points with multiplicity structures · Reduced Gröbner basis · Intersection of ideals

1 Introduction

To describe the problem, first we give the definitions below.

Definition 1 $D \subseteq \mathbb{N}_0^n$ is called a lower set in n dimensional affine space as long as $\forall d \in D$ if $d_i \neq 0$, $d - e_i$ lies in D where $e_i = (0, \ldots, 0, 1, 0, \ldots, 0)$ with the 1 situated at the ith position ($1 \leq i \leq n$). For a lower set D, we define its limiting set $E(D)$ to be the set of all $\beta \in \mathbb{N}_0^n - D$ such that whenever $\beta_i \neq 0$, then $\beta - e_i \in D$.

This work was supported by National Natural Science Foundation of China under Grant No. 11271156.

N. Lei (✉) · X. Zheng · Y. Ren
School of Mathematics, Jilin University, Changchun 130012, China
e-mail: lein@jlu.edu.cn

X. Zheng
e-mail: 490290756@qq.com

Y. Ren
e-mail: 1051221994@qq.com

R. Feng et al. (eds.), *Computer Mathematics*, DOI 10.1007/978-3-662-43799-5_21

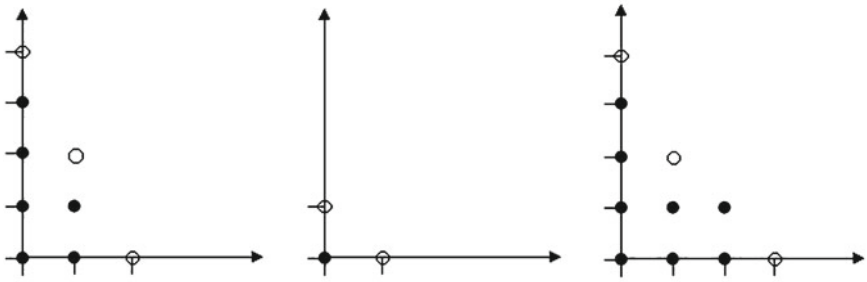

Fig. 1 Illustration of three lower sets and their limiting sets

As showed in Fig. 1, there are three lower sets and their limiting sets. The elements of the lower sets are marked by solid circles and the elements of the limiting sets are marked by blank circles.

Let k be a field and p be a point in the affine space k^n, i.e. $p = (p_1, \ldots, p_n) \in k^n$. Let $k[X]$ be the polynomial ring over k, where we write $X = (X_1, X_2, \ldots, X_n)$ for brevity's sake.

Definition 2 $\langle p, D\rangle$ represents a point p with multiplicity structure D, where p is a point in affine space k^n and $D \subseteq \mathbb{N}_0^n$ is a lower set. $\sharp D$ is called the multiplicity of point p (here we use the definition in [1]). For each $d = (d_1, \ldots, d_n) \in D$, we define a corresponding functional

$$L(f) = \frac{\partial^{d_1+\cdots+d_n}}{\partial x_1^{d_1}, \ldots, \partial x_n^{d_n}} f(p).$$

Hence for any given finite set of points with multiplicity structures $H = \{\langle p_1, D_1\rangle, \ldots, \langle p_t, D_t\rangle\}$, m functionals $\{L_i; i = 1, \ldots, m\}$ can be defined where $m \triangleq \sharp D_1 + \cdots + \sharp D_t$. We call

$$I(H) = \{f \in k[X]; L_i(f) = 0,\ i = 1, \ldots, m\}$$

the vanishing ideal of the set of the points H. The vanishing ideal problem we are focusing on is to compute the reduced Gröbner basis of the vanishing ideal for any given finite set of points H, which arises in several applications, for example, see [2] for statistics, [3] for biology, and [4–6] for coding theory.

A polynomial time algorithm for this problem was first given by Buchberger and Möller [7], then significantly improved by Marinari et al. [8], and Abbott et al. [9]. These algorithms perform Gauss elimination on a generalized Vandermonde matrix and have a polynomial time complexity in the number of points and in the number of variables. Jeffrey and Gao [10] presented a new algorithm that is essentially a generalization of Newton interpolation for univariate polynomial and has a good

computational performance when the number of variables is small relative to the number of points.

In this paper the problem we consider is under the Lexicographical order with $X_1 \succ X_2 \succ \cdots X_n$ and a more transparent algorithm will be given. The ideas are summed-up as follows:

- Construct the reduced Gröbner basis of $I(H)$ and get the quotient basis by induction over variables (define $\{M; M$ is a monomial and it is not divisible by the leading monomial for any polynomial in $I(H)\}$ as the **quotient basis** for the vanishing ideal $I(H)$).
- Get the quotient basis of the vanishing ideal purely according to the geometric distribution of the points with multiplicity structures.
- Split the original n-variable problem into smaller ones which can be solved by converting them into $(n-1)$-variable problems.
- Compute the intersection of the ideals of the smaller problems by using Extended Euclidean Algorithm.

Our algorithm can get a lower set by induction over variables for any given set of points with multiplicity structures, and by constructing the reduced Gröbner basis at the same time we can prove that the lower set is the quotient basis. There are several publications which have a strong connection to the our work although they are all only focusing on the quotient basis, ignoring the reduced Gröbner basis of the vanishing ideal. Paper [11] gives a computationally efficient algorithm to get the quotient basis of the vanishing ideal over a set of points with no multiplicity structures and the authors introduce the interesting lex game to describe the problem and the algorithm. Paper [12] offers a purely combinatorial algorithm to obtain the quotient basis and the algorithm can handle the set of points with multiplicity structures as well.

The advantage of our method is insight rather than efficient computation. The computation cost depends closely on the structure of the given set of points and a full complexity analysis would be infeasible. Our method may not be particularly efficient, but is geometrically intuitive and appealing. The clear geometric meaning of our method reveals the essential connection between the relative position of the points with multiplicity structures and the quotient basis of the vanishing ideal, providing us a new perspective of view to look into the vanishing ideal problem and helping study the structure of the reduced Gröbner basis of zero dimensional ideal under lexicographic order. What's more, our method leads to the discovery of a new algorithm to compute the intersection of two zero dimensional ideals.

Since one important feature of our method is the clear geometric meaning, to demonstrate it we present an example in Sect. 2 together with some auxiliary pictures which can make the algorithms and conclusions in this paper easier to understand. In Sects. 3 and 4 some definitions and notions are given. Sections 5 and 6 are devoted to our main algorithms of computing the reduced Gröbner basis and the quotient basis together with the proofs. In Sect. 7 we demonstrate the algorithm to compute the intersection of two ideals and some applications.

2 Example

We will use two different forms to represent the set of points with multiplicity structures H in this paper.

For easier description, we introduce the matrix form which consists of two matrices $\langle \mathscr{P} = (p_{i,j})_{m\times n}, \mathscr{D} = (d_{i,j})_{m\times n}\rangle$ with $\mathscr{P}_i, \mathscr{D}_i$ denoting the ith row vectors of $\mathscr{P}$ and $\mathscr{D}$ respectively. Each pair $\{\mathscr{P}_i, \mathscr{D}_i\}$ $(1 \le i \le m)$ defines a functional in the following way.

$$L_i(f) = \frac{\partial^{d_{i,1}+\cdots+d_{i,n}}}{\partial x_1^{d_{i,1}} \ldots \partial x_n^{d_{i,n}}} f|_{x_1=p_{i,1},\ldots,x_n=p_{i,n}}.$$

And the functional set defined here is the same with that defined by the way in Sect. 1 with respect to H.

For example, given a set of three points with their multiplicity structures $\{\langle p_1, D_1\rangle, \langle p_2, D_2\rangle, \langle p_3, D_3\rangle\}$, where $p_1 = (1, 1)$, $p_2 = (2, 1)$, $p_3 = (0, 2)$, $D_1 = \{(0, 0), (0, 1), (1, 0)\}$, $D_2 = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$, $D_3 = \{(0, 0), (1, 0)\}$, the matrix form is like the follows.

$$\mathscr{P} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 2 & 1 \\ 2 & 1 \\ 2 & 1 \\ 2 & 1 \\ 0 & 2 \\ 0 & 2 \end{pmatrix}, \mathscr{D} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 0 & 0 \\ 1 & 0 \end{pmatrix}.$$

For intuition's sake, we also represent the points with multiplicity structures in a more intuitive way as showed in the left picture of Fig. 2 where each lower set that represents the multiplicity structure of the corresponding point p is also put in

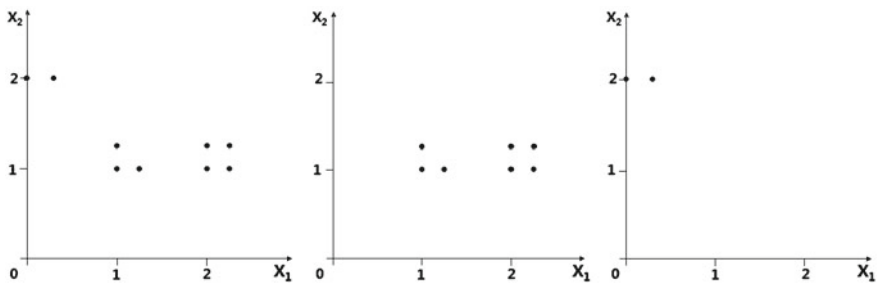

Fig. 2 The left picture represents H, the middle one is for H_1 and the right one for H_2

the affine space with the zero element (0,0) situated at p. This intuitive representing form is the basis of the geometric interpretation of our algorithm.

We take the example above to show how our method works and what the geometric interpretation of our algorithm is like:

Step 1: Define mapping $\pi : H \mapsto k$ such that $\langle p = (p_1, \ldots, p_n), D\rangle \in H$ is mapped to $p_n \in k$. So $H = \{\langle p_1, D_1\rangle, \langle p_2, D_2\rangle, \langle p_3, D_3\rangle\}$ consists of two π-fibres: $H_1 = \{\langle p_1, D_1\rangle, \langle p_2, D_2\rangle\}$ and $H_2 = \{\langle p_3, D_3\rangle\}$ as showed in the middle and the right pictures in Fig. 2. Each fibre defines a new problem, so we split the original problem defined by H into two small ones defined by H_1 and H_2 respectively.

Step 2: Solve the small problems. Take the problem defined by H_1 for example.

First, it's easy to write down one element of $I(H_1)$:

$$f_1 = (X_2 - 1)(X_2 - 1) = (X_2 - 1)^2 \in I(H_1).$$

The geometry interpretation is: we draw two lines sharing the same equation of $X_2 - 1 = 0$ to cover all the points as illustrated in the left picture in Fig. 3 and the corresponding polynomial is f_1.

According to the middle and the right pictures in Fig. 3, we can write down another two polynomials in $I(H_1)$:

$$f_2 = (X_2 - 1)(X_1 - 1)(X_1 - 2)^2 \text{ and } f_3 = (X_1 - 1)^2(X_1 - 2)^2.$$

It can be checked that $G_1 = \{f_1, f_2, f_3\}$ is the reduced Gröbner basis of $I(H_1)$, and the quotient basis is $\{1, X_1, X_2, X_1X_2, X_1^2, X_2X_1^2, X_1^3\}$. In the following, we don't distinguish explicitly an n-variable monomial $X_1^{d_1}X_2^{d_2}\ldots X_n^{d_n}$ with the element $(d_1, d_2, \ldots, d_n)$ in $\mathbb{N}_0^n$. Hence this quotient basis can be written as a subset of $\mathbb{N}_0^n$: $\{(0, 0), (1, 0), (0, 1), (1, 1), (2, 0), (2, 1), (3, 0)\}$, i.e. a lower set, denoted by D'.

In fact we can get the lower set in a more direct way by pushing the points with multiplicity structures leftward which is illustrated in the picture below (lower set D' is positioned in the right part of the picture with the (0,0) element situated at point (0,1)). The elements of the lower set D' in the right picture in Fig. 4 are marked by solid circles. The blank circles constitute the limiting set $E(D')$ and they are the leading terms of the reduced Gröbner basis $\{f_1, f_2, f_3\}$.

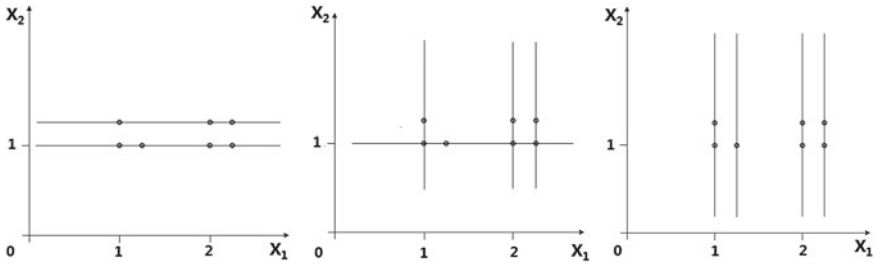

Fig. 3 Three ways to draw lines to cover the points

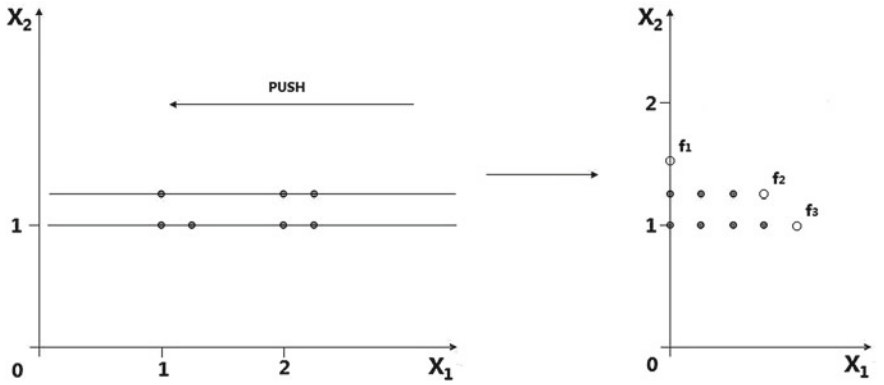

Fig. 4 Push the points leftward to get a lower set

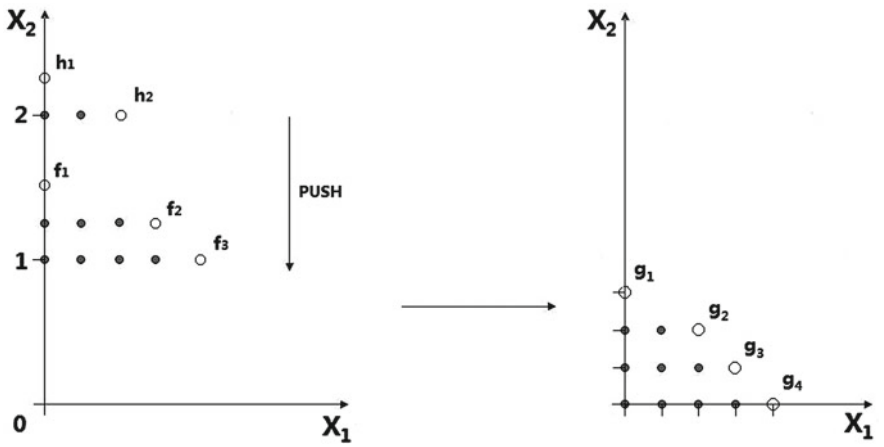

Fig. 5 Get the lower set D based on D' and D''

In the same way, we can get the Gröbner basis $G_2 = \{h_1, h_2\}$ and a lower set D'' for the problem defined by H_2, where $h_1 = (X_2 - 2)$, $h_2 = X_1^2$, $D'' = \{(0, 0), (1, 0)\}$.

Step 3: Compute the intersection of the ideals $I(H_1)$ and $I(H_2)$ to get the result for the problem defined by H.

First, we construct a new lower set D based on D', D'' in an intuitive way: let the solid circles fall down and the elements of D'' rest on the elements of D' to form a new lower set D which is showed in the right part of Fig. 5 and the blank circles represent the elements of the limiting set $E(D)$.

Then we need to find $\sharp E(D)$ polynomials vanishing on H with leading terms being the elements of $E(D)$. Take $X_1^3 X_2 \in E(D)$ for example to show the general way we do it.

We need two polynomials which vanish on H_1 and H_2 respectively, and their leading terms both have the **same degrees** of X_1 with that of the desired monomial $X_1^3X_2$ and both have the **minimal degrees** of X_2. Notice that f_2 and $X_1 \cdot h_2$ satisfy the requirement. And then we multiply f_2 and $X_1 \cdot h_2$ with h_1, f_1 respectively which are all univariate polynomials in X_2 to get two polynomials q_1, q_2 such that q_1 and q_2 both vanish on H. Obviously q_1 and q_2 still have the **same degrees** of X_1 with that of the desired monomial $X_1^3X_2$.

$$q_1 = f_2 \cdot h_1 = (X_2 - 1)(X_1 - 1)(X_1 - 2)^2(X_2 - 2),$$

$$q_2 = X_1 \cdot h_2 \cdot f_1 = X_1^3(X_2 - 1)^2.$$

Next try to find two univariate polynomials in X_2: r_1, r_2 such that $q_1 \cdot r_1 + q_2 \cdot r_2$ vanishes on H (which is obviously true already) and has the desired leading term $X_1^3X_2$.

$$\begin{aligned} q_1 &= (X_2 - 2)(X_2 - 1)X_1^3 - (5X_2^2 - 15X_2 + 10)X_1^2 \\ &\quad + (8X_2^2 - 24X_2 + 16)X_1 - 4X_2^2 + 12X_2 - 8, \\ q_2 &= (X_2 - 1)^2X_1^3. \end{aligned}$$

To settle the leading term issue, write q_1, q_2 as univariate polynomials in X_1 as above. Because $X_2 \prec X_1$ and the highest degrees of X_1 of the leading terms of q_1, q_2 are both 3, we know that as long as the leading term of $(X_2 - 2)(X_2 - 1)X_1^3 \cdot r_1 + (X_2 - 1)^2X_1^3 \cdot r_2$ is $X_1^3X_2$, the leading term of $q_1 \cdot r_1 + q_2 \cdot r_2$ is also $X_1^3X_2$.

$$\begin{aligned} &(X_2 - 2)(X_2 - 1)X_1^3 \cdot r_1 + (X_2 - 1)^2X_1^3 \cdot r_2 \\ &\quad = X_1^3(X_2 - 1)\left((X_2 - 2) \cdot r_1 + (X_2 - 1) \cdot r_2\right) \end{aligned}$$

Obviously if and only if $(X_2 - 2) \cdot r_1 + (X_2 - 1) \cdot r_2 = 1$ we can keep the leading term of $q_1 \cdot r_1 + q_2 \cdot r_2$ to be $X_1^3X_2$. In this case $r_1 = -1$ and $r_2 = 1$ will be just perfect. In our algorithm we use Extended Euclidean Algorithm to compute r_1, r_2.

Finally, we obtain

$$g_3 = q_1 \cdot r_1 + q_2 \cdot r_2$$

$$= (X_2 - 1)X_1^3 + (5X_2^2 - 15X_2 + 10)X_1^2 - (8X_2^2 - 24X_2 + 16)X_1 + 4X_2^2 - 12X_2 + 8$$

which vanishes on H and has $X_1^3X_2$ as its leading term.

In the same way, we can get $g_1 = (X_2 - 1)^2(X_2 - 2)$ for X_2^3, $g_2 = (X_2 - 1)^2X_1^2$ for $X_1^2X_2^2$ and $g_4 = X_1^4 + 6(X_2^2 - 2X_2)X_1^3 - 13(X_2^2 - 2X_2)X_1^2 + 12(X_2^2 - 2X_2)X_1 - 4(X_2^2 - 2X_2)$ for X_1^4. In fact we need to compute g_1, g_2, g_3 and g_4 in turn according to the lexicographic order because we need reduce g_2 by g_1, reduce g_3 by g_2 and g_1, and reduce g_4 by g_1, g_2 and g_3.

The reduced polynomial set can be proved in Sect. 6 to be the reduced Gröbner basis of the intersection of two ideals which is exactly the vanishing ideal over H, and D is the quotient basis.

This example shows what the geometric interpretation of our method is like: for any given point with multiplicity structure $\langle p_i, D_i \rangle$, we put the lower set D_i into the affine space with the (0,0) element situated at p_i to intuitively represent it, and it can be imagined as $\sharp D_i$ small balls in the affine space; for bivariate problem, we first push the balls along the X_1-axis, then along the X_2-axis to get a lower set as we did in the example above; the lower set is exactly the quotient basis and the limiting set of the lower set is the set of the leading monomials of the reduced Göbner basis. This intuitive understanding can be applied to the n-variable problem and can help us understand the algorithm better in the following.

3 Notions

First, we define the following mappings:

$$\begin{aligned}
&\mathrm{proj} : \mathbb{N}_0^n \longrightarrow \mathbb{N}_0 \\
&\qquad (d_1, \ldots, d_n) \longrightarrow d_n. \\
&\widehat{\mathrm{proj}} : \mathbb{N}_0^n \longrightarrow \mathbb{N}_0^{n-1} \\
&\qquad (d_1, \ldots, d_n) \longrightarrow (d_1, \ldots, d_{n-1}). \\
&\mathrm{embed}_c : \mathbb{N}_0^{n-1} \longrightarrow \mathbb{N}_0^n \\
&\qquad (d_1, \ldots, d_{n-1}) \longrightarrow (d_1, \ldots, d_{n-1}, c).
\end{aligned}$$

Let $D \subset \mathbb{N}_0^n$, and naturally we define $\widehat{\mathrm{proj}}(D) = \{\widehat{\mathrm{proj}}(d) | d \in D\}$, and $\mathrm{embed}_c(D') = \{\mathrm{embed}_c(d) | d \in D'\}$ where $D' \subset \mathbb{N}_0^{n-1}$. In fact we can apply these mappings to any set $O \subset k^n$ or any matrix of n columns, because there is no danger of confusion. For example, let M be a matrix of n columns, and $\widehat{\mathrm{proj}}(M)$ is a matrix of $n-1$ columns with the first $n-1$ columns of M reserved and the last one eliminated.

The embed_c mapping embeds a lower set of the $n-1$ dimensional space into the n dimensional space. When the embed_c operation parameter c is zero, we can get a lower set of $\mathbb{N}_0^n$ by mapping each element $d = (d_1, \ldots, d_{n-1})$ to $d = (d_1, \ldots, d_{n-1}, 0)$ as showed below.

Blank circles represent the elements of the limiting sets. Note that after the embed_c mapping, there is one more blank circle. In this case, the limiting set is always increased by one element $(0, \ldots, 0, 1)$.

In the case the embed_c operation parameter c is not zero, it is obvious that what we got is not a lower set any more. But there is another intuitive fact we should realize (Fig. 6).

Theorem 1 *Assume $D_0, D_1, \ldots, D_\ell$ are lower sets in $n-1$ dimensional space, and $D_0 \supseteq D_1 \supseteq \cdots \supseteq D_\ell$. Let $\hat{D}_i = embed_i(D_i), i = 0, \ldots, \ell$. Then $D = \bigcup_{i=0}^{\ell} \hat{D}_i$ is a lower set in n dimensional space, and $E(D) \subseteq C$ where $C = \bigcup_{i=0}^{\ell} embed_i(E(D_i)) \bigcup \{(0, \ldots, 0, \ell+1)\}$.*

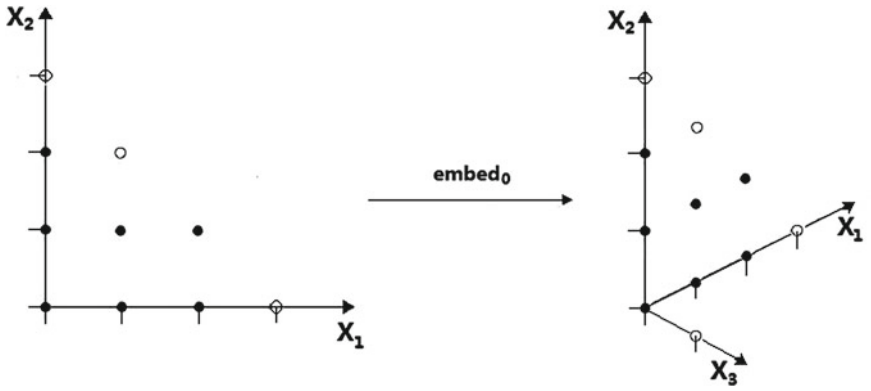

Fig. 6 Embed the lower set in 2-D space into 3-D space with parameter $c = 0$

Proof First to prove D is a lower set. $\forall d \in D$, let $i = \text{proj}(d)$, then $d \in \hat{D}_i$ i.e. $\widehat{\text{proj}}(d) \in \widehat{\text{proj}}(\hat{D}_i) = D_i$. Because D_i is a lower set, hence for $j = 1, \ldots, n-1$, if $d_j \neq 0$, then $\widehat{\text{proj}}(d) - \widehat{\text{proj}}(e_j) \in D_i$ where $e_j = (0, \ldots, 0, 1, 0, \ldots, 0)$ with the 1 situated at the jth position. So $d - e_j \in \hat{D}_i \subseteq D$. For $j = n$, if $i = 0$, we are finished. If $i \neq 0$, there must be $d - e_n \in \hat{D}_{i-1} \subseteq D$. Because if $d - e_n \notin \hat{D}_{i-1}$, we have $\widehat{\text{proj}}(d) \notin D_{i-1}$. Since we already have $\widehat{\text{proj}}(d) \in D_i$, this is contradictory to $D_i \subseteq D_{i-1}$.

Second, assume $\forall d \in E(D)$, $\widehat{\text{proj}}(d) \notin D_i, i = 0, \ldots, \ell$. If $\widehat{\text{proj}}(d)$ is a zero tuple, then d_n must be $\ell + 1$, that is $d \in C$. If $\widehat{\text{proj}}(d)$ is not a zero tuple, then we know $d_n < \ell + 1$. If $d_j \neq 0$, $j = 1, \ldots, n-1$, then $d - e_j \in \text{embed}_{d_n}(D_{d_n})$. Then $\widehat{\text{proj}}(d) - \widehat{\text{proj}}(e_j) \in D_{d_n}$, that is $\widehat{\text{proj}}(d) \in E(D_{d_n})$. Finally with the embed_{d_n} operation we have $d \in \text{embed}_{d_n}(E(D_{d_n}))$ where $d_n < \ell + 1$. So $d \in C$. □

4 Addition of Lower Sets

In this section, we define the addition of lower sets which is the same with that in [13], the following paragraph and Fig. 7 are basically excerpted from that paper with a little modification of expression.

To get a visual impression of what the addition of lower sets are, look at the example in Fig. 7. What is depicted there can be generalized to arbitrary lower sets D_1, D_2 and arbitrary dimension n. The process can be described as follows. Draw a coordinate system of $\mathbb{N}_0^n$ and insert D_1. Place a translate of D_2 somewhere on the X_2-axis. The translate has to be sufficiently far out, so that D_1 and the translate D_2 do not intersect. Then push the elements of the translate of D_2 down along the X_2-axis until on room remains between them and the elements of D_1. The resulting lower set is denoted by $D_1 + D_2$.

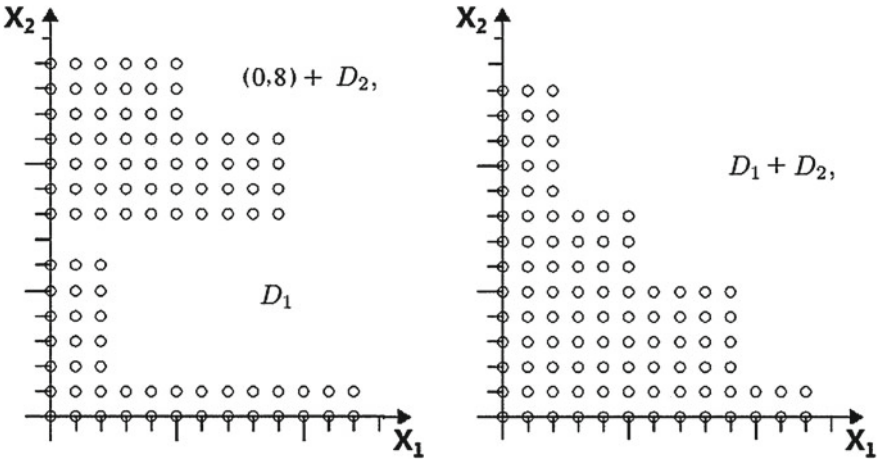

Fig. 7 Addition of D_1 and D_2

Intuitively, we define algorithm **ALS** (short for *Addition of Lower Sets*) to realize the addition of lower sets.

Algorithm **ALS:** Given two lower sets in n dimensional space D_1, D_2, determine another lower set as the addition of D_1, D_2, denoted by $D := D_1 + D_2$.

Step 1 $D := D_1$;
Step 2 If $\sharp D_2 = 0$ return D. Else pick $a \in D_2$, $D_2 := D_2 \setminus \{a\}$.

Step 2.1 If $a \in D$, add the last coordinate of a with 1. Go to Step 2.1.
Else
$D := D \bigcup \{a\}$, go to Step 2.

Given three lower sets D_1, D_2, D_3, the addition we defined satisfies:

1. $D_1 + D_2 = D_2 + D_1$,
2. $(D_1 + D_2) + D_3 = D_1 + (D_2 + D_3)$,
3. $D_1 + D_2$ is a lower set,
4. $\sharp(D_1 + D_2) = \sharp D_1 + \sharp D_2$.

These are all the same with that in [13]. And the proof can be referred to it.

As implied in the example of Sect. 2, when we want to get a polynomial with leading term d_3 showed in the right part of Fig. 8, we need two polynomials with the leading terms d_1, d_2 which are not the elements of the lower sets and have the same degrees of X_1 as d_3 and the minimal degrees of X_2 as showed in the left part of Fig. 8. In other words, $d_1 \notin D_1$, $d_2 \notin D_2$, $\widehat{\text{proj}}(d_1) = \widehat{\text{proj}}(d_2) = \widehat{\text{proj}}(d_3)$, $\text{proj}(d_1) + \text{proj}(d_2) = \text{proj}(d_3)$. It's easy to understand that these equations hold for the addition of three or even more lower sets.

We use algorithm **CLT** (short of *Computing the Leading Term*) to get the leading terms d_1 and d_2 from d_3 respectively.

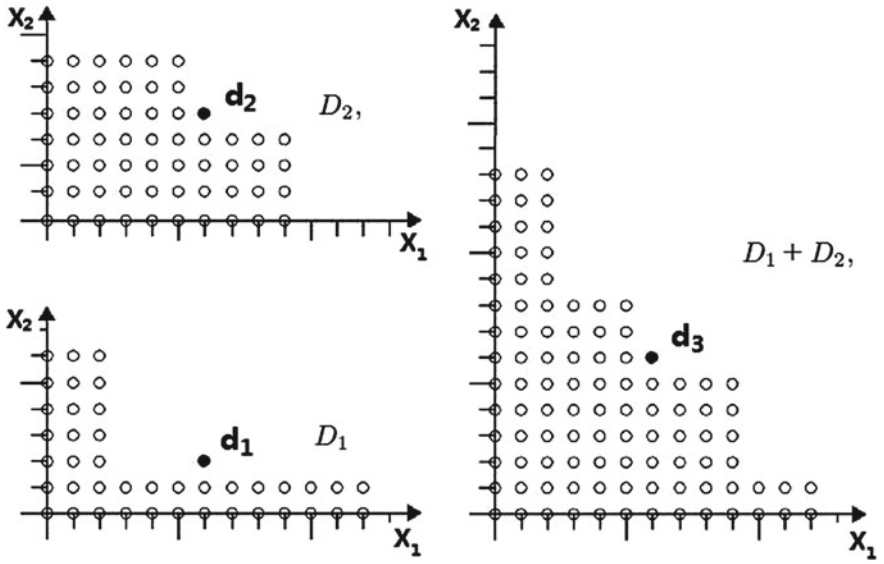

Fig. 8 $\widehat{\text{proj}}(d_1) = \widehat{\text{proj}}(d_2) = \widehat{\text{proj}}(d_3)$, $\text{proj}(d_1) + \text{proj}(d_2) = \text{proj}(d_3)$

Algorithm **CLT:** Given $a \in \mathbb{N}_0^n$, and a lower set D in n dimensional space satisfying $a \notin D$. Determine another $r = (r_1, \ldots, r_n) \in \mathbb{N}_0^n$ which satisfies that $r \notin D$, $\widehat{\text{proj}}(r) = \widehat{\text{proj}}(a)$ and $(r_1, \ldots, r_{n-1}, r_n - 1) \in D$, denoted by $r := \text{CLT}(a, D)$.

Step 1 Initialize r such as $\widehat{\text{proj}}(r) = \widehat{\text{proj}}(a)$ and $\text{proj}(r) = 0$.
Step 2 if $r \notin D$, return r, else $r_n := r_n + 1$, go to Step 2.

Then $d_1 = \text{CLT}(d_3, D_1)$, $d_2 = \text{CLT}(d_3, D_2)$.

Definition 3 For any $f \in k[X]$, view it as an element in $k(X_n)[X_1, \ldots, X_{n-1}]$ and define $\text{LC}_n(f)$ to be the leading coefficient of f which is a univariate polynomial in X_n.

Here is the algorithm **CP** (short for *Computing the Polynomial*) which can compute the polynomial with the leading term returned by algorithm **CLT**.

Algorithm **CP:** D is a lower set in n dimensional space, $a \in \mathbb{N}_0^n$ and $a \notin D$, $G := \{f_{ed} \in k[X]; \text{the leading term of } f_{ed} \text{ is } ed,\ ed \in E(D)\}$, algorithm **CP** returns a polynomial p whose leading term is $\text{CLT}(a, D)$. Denoted by $p := \text{CP}(a, D, G)$.

Step 1 $c := \text{CLT}(a, D)$.
Step 2 Select $c' \in E(D), s.t.\ c'$ is a factor of c. $d := \frac{c}{c'}$.
(d is well defined because $c \notin D$).
Step 3 $p := f_{c'} \cdot d$ where $f_{c'}$ is an element of G whose leading term is $c' \in E(D)$.
(p is well defined because d is well defined).

Remark 1 $\mathrm{LC}_n(f_{c'}) = \mathrm{LC}_n(p)$ in Step 3. Since c has the minimal degree of X_n according to algorithm **CLT**, there exists no element $c'' \in E(D)$ which is a factor of c satisfying $\mathrm{proj}(c'') < \mathrm{proj}(c)$. Hence monomial d in the algorithm does not involve the variable X_n.

5 Associate a Lower set $D(H)$ to a set of Points H with Multiplicity Structures

For any given set of points H with multiplicity structures in n dimensional space, we can construct a lower set $D(H)$ in n dimensional space by induction.

Univariate case: Assume $H = \{\langle p_1, D_1\rangle, \ldots, \langle p_t, D_t\rangle\}$ is a set of points with multiplicity structures in one dimensional space, then the lower set is $D(H) = \{0, 1, \ldots, \sum_{i=1}^{t} \sharp D_i\}$.

Assume the $n-1$ $(n > 1)$ dimensional problem has been solved, now for the n dimensional situation, we first focus on the **Special case**.

Special case: Assume $H = \{\langle p_1, D_1\rangle, \ldots, \langle p_t, D_t\rangle\}$ is a set of points with multiplicity structures in the n $(n > 1)$ dimensional space where all the points share the same X_n coordinate. Write H in matrix form as $\langle \mathscr{P}, \mathscr{D}\rangle$ and all the entries in the last column of matrix $\mathscr{P}$ have the same value. Classify the row vectors of $\langle \mathscr{P}, \mathscr{D}\rangle$ to get $\{\langle \mathscr{P}_0, \mathscr{D}_0\rangle, \ldots, \langle \mathscr{P}_w, \mathscr{D}_w\rangle\}$ according to the values of the entries in the last column of matrix $\mathscr{D}$ and we guarantee the corresponding relationship between the row vectors of matrix $\mathscr{P}$ and matrix $\mathscr{D}$ holds in $\langle \mathscr{P}_i, \mathscr{D}_i\rangle$ $(0 \le i \le w)$. All the entries in the last column of $\mathscr{D}_i$ are the same i and the entries of the last column of $\mathscr{P}_i$ stay the same too. Then eliminate the last columns of $\mathscr{P}_i$ and $\mathscr{D}_i$ to get $\langle \widehat{\mathrm{proj}}(\mathscr{P}_i), \widehat{\mathrm{proj}}(\mathscr{D}_i)\rangle$ which represents a set of points with multiplicity structures in $n-1$ dimensional space, by induction we get a lower set $\hat{D}_i$ in $n-1$ dimensional space. Then we set

$$D(H) = \bigcup_{i=0}^{w} \mathrm{embed}_i(\hat{D}_i).$$

Next we deal with the **General case**.

General case: Assume $H = \{\langle p_1, D_1\rangle, \ldots, \langle p_t, D_t\rangle\}$ is a set of points with multiplicity structures in the n $(n > 1)$ dimensional space. Split the set of points: $H = H_1 \bigcup H_2 \bigcup \cdots \bigcup H_s$ such that the points of each H_i are in the same π-fibre, i.e. they have the same X_n coordinate $c_i, i = 1, \ldots, s$,and $c_i \neq c_j, \forall i, j = 1, \ldots, s, i \neq j$. According to the **Special case**, for each $i = 1, \ldots, s$, we can get a lower set $D(H_i)$, then we set

$$D(H) = \sum_{i=1}^{s} D(H_i).$$

We now prove $D(H)$ is a lower set although it is easy to understand as long as the geometric interpretation involves. Since it is obviously true for **Univariate case**, induction over dimension would be helpful for the proof.

Proof Assume $D(H)$ is a lower set for the $n-1$ dimensional situation and now we prove the conclusion for n dimensional situation ($n > 1$).

First to prove $D(H)$ of the **Special case** is a lower set.

We claim that $\langle \widehat{\text{proj}}(\mathscr{P}_i), \widehat{\text{proj}}(\mathscr{D}_i)\rangle$ represents a set of points with multiplicity structures in $n-1$ dimensional space ($i = 0, \ldots, w$). For any $D \subset \mathbb{N}_0^n$, define $F_a(D) = \{d \in D |\ \text{proj}(d) = a\}$. Let $U = \{u | u \in \{1, \ldots, t\}, F_i(D_u) \neq \varnothing\}$. So $\langle \widehat{\text{proj}}(\mathscr{P}_i), \widehat{\text{proj}}(\mathscr{D}_i)\rangle$ can be written in the form of $\{\langle \widehat{\text{proj}}(p_u), \widehat{\text{proj}}(F_i(D_u))\rangle | u \in U\}$. It is apparent that $\widehat{\text{proj}}(F_i(D_u))$ is a lower set in $n-1$ dimensional space and can be viewed as the multiplicity structure of the point $\widehat{\text{proj}}(p_u)$. Hence $\langle \widehat{\text{proj}}(\mathscr{P}_i), \widehat{\text{proj}}(\mathscr{D}_i)\rangle$ is a set of points with multiplicity structures in $n-1$ dimensional space.

What's more, we assert $\widehat{\text{proj}}(\mathscr{P}_j)$ is a sub-matrix of $\widehat{\text{proj}}(\mathscr{P}_i)$, and $\widehat{\text{proj}}(\mathscr{D}_j)$ is a sub-matrix of $\widehat{\text{proj}}(\mathscr{D}_i)$, $0 \leq i < j \leq w$. Because of the corresponding relationship between the row vectors in $\mathscr{P}$ and $\mathscr{D}$, we need only to prove $\widehat{\text{proj}}(\mathscr{D}_j)$ is a sub-matrix of $\widehat{\text{proj}}(\mathscr{D}_i)$. If it is not true, there exists a row vector g of $\widehat{\text{proj}}(\mathscr{D}_j)$ which is not a row vector of $\widehat{\text{proj}}(\mathscr{D}_i)$. That is, there exists b ($1 \leq b \leq t$) such that $\text{embed}_j(g)$ is an element of the lower set D_b, and $\text{embed}_i(g)$ is not included in any lower set D_a ($1 \leq a \leq t$). However since $i < j$ and $\text{embed}_j(g) \in D_b$, $\text{embed}_i(g)$ must be included in D_b. Hence our assertion is true.

Since $\widehat{\text{proj}}(\mathscr{P}_j)$ is a sub-matrix of $\widehat{\text{proj}}(\mathscr{P}_i)$, and $\widehat{\text{proj}}(\mathscr{D}_j)$ is a sub-matrix of $\widehat{\text{proj}}(\mathscr{D}_i)$, $0 \leq i < j \leq w$. According to the assumption of induction and the way we construct $D(H)$, we have $\hat{D}_i \supseteq \hat{D}_j, 0 \leq i < j \leq w$, where $\hat{D}_i$, $\hat{D}_j$ are both lower sets. Based on the Theorem 1 in Sect. 3, $D(H) = \bigcup_{i=0}^{w} \text{embed}_i(\hat{D}_i)$ is a lower set, and $E(D(H)) \subseteq \bigcup_{i=0}^{w} \text{embed}_i(E(\hat{D}_i)) \bigcup \{(0, \ldots, 0, w+1)\}$.

Then to prove $D(H)$ of **General case** is a lower set. Since $D(H_i), i = 1, \ldots, s$ are lower sets, and the addition of lower sets is also a lower set according to Sect. 4, $D(H)$ is obviously a lower set. □

6 Associate a set of Polynomials poly(H) to $D(H)$

For every lower set constructed during the induction procedure showed in the last section, we associate a set of polynomials to it.

We begin with the univariate problem as we did in the last section.

6.1 Univariate Problem

P-Univariate case:
Assume $H = \{\langle p_1, D_1\rangle, \ldots, \langle p_t, D_t\rangle\}$ is a set of points with multiplicity structures in one dimensional space, and $D(H) = \{0, 1, \ldots, \sum_{i=1}^{t} \sharp D_i\}$. Then the set of univariate polynomials associated to $D(H)$ is $poly(H) = \{\prod_{i=1}^{t}(X_1 - p_i)^{\sharp D_i}\}$.

Obviously $poly(H)$ of **P-Univariate case** satisfies the following **Assumption**.

Assumption For any given set of points with multiplicity structures H in the $n-1$ $(n>1)$ dimensional space, there are the following properties. For any $\lambda \in E(D(H))$, there exists a polynomial $f_\lambda \in k[X]$ where $X=(X_1,\ldots,X_{n-1})$ such that

- The leading term of f_λ under lexicographic order is X^λ.
- The exponents of all lower terms of f_λ lies in $D(H)$.
- f_λ vanishes on H.
- $\text{poly}(H)=\{f_\lambda|\lambda \in E(D(H))\}$.

Now assume the $(n-1)$-variable $(n>1)$ problem has been solved i.e. for any given set of points with multiplicity structures H in $n-1$ dimensional space, we can construct a set of polynomial $\text{poly}(H)$ which satisfies the **Assumption**. And then to tackle the n-variable problem, we still begin with the special case.

6.2 Special Case of the n-variable ($n>1$) Problem

P-Special case:
Given a set of points with multiplicity structures in n $(n>1)$ dimensional space $H=\{\langle p_1,D_1\rangle,\ldots,\langle p_t,D_t\rangle\}$ or in matrix form $\langle \mathscr{P}=(p_{ij})_{m\times n}, \mathscr{D}=(d_{ij})_{m\times n}\rangle$. All the given points have the same X_n coordinate, i.e. the entries in the last column of $\mathscr{P}$ are the same. We compute $\text{poly}(H)$ with the following steps.

Step 1 $c:=p_{1n}$; $w=\max\{d_{in};i=1,\ldots,m\}$.
Step 2 $\forall i=0,\ldots,w$, define $\mathscr{SD}_i$ as a sub-matrix of $\mathscr{D}$ containing all the row vectors whose last coordinates equal i. Extract the corresponding row vectors of $\mathscr{P}$ to form matrix $\mathscr{SP}_i$, and the corresponding relationship between the row vectors in $\mathscr{P}$ and $\mathscr{D}$ holds for $\mathscr{SP}_i$ and $\mathscr{SD}_i$.
Step 3 $\forall i=0,\ldots,w$, eliminate the last columns of $\mathscr{SP}_i$ and $\mathscr{SD}_i$ to get $\langle \tilde{\mathscr{SP}}_i, \tilde{\mathscr{SD}}_i\rangle$ which represents a set of points in $n-1$ dimensional space with multiplicity structures.According to the induction assumption, we have the polynomial set $\tilde{G}_i=\text{poly}(\langle \tilde{\mathscr{SP}}_i,\tilde{\mathscr{SD}}_i\rangle)$ associated to the lower set $\tilde{D}_i=D(\langle \tilde{\mathscr{SP}}_i,\tilde{\mathscr{SD}}_i\rangle)$.
Step 4 $D:=\bigcup_{i=0}^{w}\text{embed}_i(\tilde{D}_i)$. Multiply every element of $\tilde{G}_i$ with $(X_n-c)^i$ to get G_i. $\tilde{G}:=\bigcup_{i=0}^{w}G_i\bigcup\{(X_n-c)^{w+1}\}$.
Step 5 Eliminate the polynomials in $\tilde{G}$ whose leading term is not included in $E(D)$ to get $\text{poly}(H)$.

Theorem 2 *The* $\text{poly}(H)$ *obtained in* ***P-Special case*** *satisfies the* ***Assumption***.

Proof According to the Sect. 5, $\langle \tilde{\mathscr{SP}}_i,\tilde{\mathscr{SD}}_i\rangle$ represents a set of points with multiplicity structures in $n-1$ dimensional space for $i=0,\ldots,w$. And $\tilde{D}_j\supseteq\tilde{D}_i$, $0\le j\le i\le w$. D is a lower set and $E(D)\subseteq\bigcup_{i=0}^{w}\text{embed}_i(E(\tilde{D}_i))\bigcup\{(0,\ldots,0,w+1)\}$.

For $\lambda = (0, \ldots, 0, w+1) \in E(D)$, we have $f_\lambda = (X_n - c)^{w+1}$. It is easy to check that it satisfies the first three terms of the **Assumption**.

For any other element ed of $E(D), \exists \ell\, s.t.\, ed \in \text{embed}_\ell E(\tilde{D}_\ell)$. So let $\tilde{ed}$ be the element in $E(\tilde{D}_\ell)$ such that $ed = \text{embed}_\ell(\tilde{ed})$. We have $f_{\tilde{ed}}$ vanishes on $\langle \tilde{\mathscr{S}\mathscr{P}}_\ell, \tilde{\mathscr{S}\mathscr{D}}_\ell \rangle$ whose leading term is $\tilde{ed} \in E(\tilde{D}_\ell)$ and the lower terms belong to $\tilde{D}_\ell$. According to the algorithm $f_{ed} = (X_n - c)^\ell \cdot f_{\tilde{ed}} \in \text{poly}(H)$.

First it is easy to check that the leading term of f_{ed} is ed since $ed = \text{embed}_\ell(\tilde{ed})$.

Second, the lower terms of f_{ed} are all in the set $S = \bigcup_{j=0}^{\ell} \text{embed}_j(\tilde{D}_\ell)$ because all the lower terms of $f_{\tilde{ed}}$ are in the set $\tilde{D}_\ell$. $\tilde{D}_0 \supseteq \tilde{D}_1 \supseteq \ldots \tilde{D}_\ell$, so $\text{embed}_j(\tilde{D}_\ell) \subset \text{embed}_j(\tilde{D}_j)$ $(0 \le j \le \ell)$, hence $S \subseteq D = \bigcup_{j=0}^{w} \text{embed}_j(\tilde{D}_j)$ and the second term of the **Assumption** is satisfied.

Third, we are going to prove that f_{ed} vanishes on all the functionals defined by $\langle \mathscr{P}, \mathscr{D} \rangle$, i.e. all the functionals defined by $\langle \mathscr{S}\mathscr{P}_i, \mathscr{S}\mathscr{D}_i \rangle$ $(i = 0, \ldots, w)$. Write all the functionals defined by $\langle \mathscr{S}\mathscr{P}_i, \mathscr{S}\mathscr{D}_i \rangle$ in this form: $L' \cdot \frac{\partial^i}{\partial X_n^i}|_{X_n = c}$ where L' is an $n-1$ variable functional. Substitute the zeroes and use the fact that $f_{\tilde{ed}}$ vanishes on $\langle \tilde{\mathscr{S}\mathscr{P}}_\ell, \tilde{\mathscr{S}\mathscr{D}}_\ell \rangle$, it's apparent that $f_{ed} = (X_n - c)^\ell \cdot f_{\tilde{ed}}$ vanishes on these functionals.

So f_{ed} vanishes on H, and satisfies the first three terms of the **Assumption**.

In summary $\text{poly}(H)$ satisfies the **Assumption**. □

Remark 2 For $f_\lambda \in \text{poly}(H), \lambda \in E(D)$ where $\text{poly}(H)$ is the result gotten in the algorithm above, we have the conclusion that $\text{LC}_n(f_\lambda) = (X_n - c)^{\text{proj}(\lambda)}$.

6.3 General Case of the n-variable ($n > 1$) Problem

P-General case:

Given a set of points with multiplicity structures in n $(n > 1)$ dimensional space $H = \{\langle p_1, D_1 \rangle, \ldots, \langle p_t, D_t \rangle\}$ or in matrix form $\langle \mathscr{P} = (p_{ij})_{m \times n}, \mathscr{D} = (d_{ij})_{m \times n} \rangle$, we are going to get $\text{poly}(H)$.

Step 1 Write H as $H = H_1 \bigcup H_2 \bigcup \cdots \bigcup H_s$ where H_i $(1 \le i \le s)$ is a π-fibre ($\pi : H \mapsto k$ such that $\langle p = (p_1, \ldots, p_n), D \rangle \in H$ is mapped to $p_n \in k$) i.e. the points of H_i have the same X_n coordinate c_i, $i = 1, \ldots, s$, and $c_i \ne c_j, \forall i, j = 1, \ldots, s, i \ne j$.

Step 2 According to the **P-Special case**, we have $D_i' = D(H_i)$, $G_i = \text{poly}(H_i)$. Write H_i as $\langle \mathscr{P}_i, \mathscr{D}_i \rangle$, and define w_i as the maximum value of the elements in the last column of $\mathscr{D}_i$.

Step 3 $D := D_1'$, $G := G_1$, $i := 2$.

Step 4 If $i > s$, go to Step 5. Else

Step 4.1 $D := D + D_i'$; $\hat{G} := \varnothing$. View $E(D)$ as a monomial set $MS := E(D)$.

Step 4.2 If $\sharp MS = 0$, go to Step 4.7, else select the minimal element of MS under lexicographic order, denoted by LT. $MS := MS \setminus \{LT\}$.

Step 4.3

$$f_1 := \mathrm{CP}(LT, D, G),\ f_2 := \mathrm{CP}(LT, D_i', G_i).$$

$$v_\wp := \mathrm{proj}(g_\wp), \text{ where } g_\wp := \mathrm{CLT}(LT, D_\wp'),\ \wp = 1, \ldots, i.$$

Step 4.4

$$q_1 := f_1 \cdot (X_n - c_i)^{w_i+1};\ \ q_2 := f_2 \cdot \prod_{\wp=1}^{i-1} (X_n - c_\wp)^{w_\wp+1}.$$

$$pp1 := (X_n - c_i)^{w_i+1-v_i};\ \ pp2 := \prod_{\wp=1}^{i-1} (X_n - c_\wp)^{w_\wp+1-v_\wp}.$$

Step 4.5 Use Extended Euclidean Algorithm to compute r_1 and r_2 s.t. $r_1 \cdot pp_1 + r_2 \cdot pp_2 = 1$.

Step 4.6 $f := r_1 \cdot q_1 + r_2 \cdot q_2$. Reduce f with the elements in $\hat{G}$ to get f'; $\hat{G} := \hat{G} \bigcup \{f'\}$. Go to Step 4.2.

Step 4.7 $G := \hat{G}$. $i := i + 1$. Go to Step 4.

Step 5 $\mathrm{poly}(H) := G$.

Theorem 3 *The poly(H) obtained in* ***P-General case*** *satisfies the* ***Assumption****.*

Proof It is easy to know $v_\wp \le w_\wp + 1$ according to their definitions, so the polynomials $pp1$ and $pp2$ in Step 4.4 do make sense. And to prove Theorem 3, we need only to prove the situation that $s \ge 2$ in Step 1.

For $i = 2$, $D = D_1' + D_2'$, $\forall ed \in E(D)$, $v := \mathrm{proj}(ed)$ and $X_0 := \frac{X^{ed}}{X_n^v}$. According to Sect. 4, we have $v = v_1 + v_2$. Based on the Remarks 1 and 2, f_1 and f_2 can be written as polynomials of $k(X_n)[X_1, \ldots, X_{n-1}]$: $f_1 = X_0 \cdot (X_n - c_1)^{v_1} + \textit{the rest}$ and $f_2 = X_0 \cdot (X_n - c_2)^{v_2} + \textit{the rest}$ and none of the monomials in *the rest* is greater than or equal to X_0. Because f_1 and $(X_n - c_1)^{w_1+1}$ vanish on H_1, f_2 and $(X_n - c_2)^{w_2+1}$ vanish on H_2, we know that $q_1 = f_1 \cdot (X_n - c_2)^{w_2+1}$ and $q_2 = f_2 \cdot (X_n - c_1)^{w_1+1}$ both vanish on $H_1 \bigcup H_2$. Then f vanishes on $H_1 \bigcup H_2$ where

$$\begin{aligned} f &= r_1 \cdot q_1 + r_2 \cdot q_2 \\ &= X_0 \cdot (X_n - c_1)^{v_1} \cdot (X_n - c_2)^{v_2} (r_1 \cdot (X_n - c_2)^{w_2+1-v_2} + r_2 \cdot (X_n - c_1)^{w_1+1-v_1}) \\ &\quad + \textit{the rest} \\ &= X_0 \cdot (X_n - c_1)^{v_1} \cdot (X_n - c_2)^{v_2} (r_1 \cdot pp1 + r_2 \cdot pp2) + \textit{the rest} \\ &= X_0 \cdot (X_n - c_1)^{v_1} \cdot (X_n - c_2)^{v_2} + \textit{the rest}. \end{aligned}$$

None monomial in *the rest* is greater than or equal to X_0 , so the leading term of f is obviously $X_0 \cdot X_n^v$ which is equal to ed. Naturally $\mathrm{LC}_n(f) = \prod_{j=1}^{i} (X_n - c_j)^{v_j}$ for $i = 2$.

We **assert** that for any i, the polynomial f in [step 4.6] satisfies that $\mathrm{LC}_n(f) = \prod_{j=1}^{i}(X_n - c_j)^{v_j}$.

When $i > 2$, assume the **assertion** above holds for $i-1$. $\forall\, ed \in E(D)$, $v := \mathrm{proj}(ed)$ and $X_0 := \frac{X^{ed}}{X_n^v}$. According to Sect. 4, we have $v = v_1 + \cdots + v_i$. Based on the **assertion** for $i-1$, Remarks 1 and 2, f_1 and f_2 can be written as polynomials of $k(X_n)[X_1, \ldots, X_{n-1}]$:

$$f_1 = X_0 \cdot \prod_{j=1}^{i-1}(X_n - c_j)^{v_j} + \textit{the rest}$$
$$f_2 = X_0 \cdot (X_n - c_i)^{v_i} + \textit{the rest}$$

and none of the monomials in *the rest* is greater than or equal to X_0. Because f_1 and $\prod_{j=1}^{i-1}(X_n - c_j)^{w_j+1}$ vanish on $\bigcup_{j=1}^{i-1} H_j$, f_2 and $(X_n - c_i)^{w_i+1}$ vanish on H_i, we know that $q_1 = f_1 \cdot (X_n - c_i)^{w_i+1}$ and $q_2 = f_2 \cdot \prod_{j=1}^{i-1}(X_n - c_j)^{w_j+1}$ both vanish on $\bigcup_{j=1}^{i} H_j$. Then f vanishes on $\bigcup_{j=1}^{i} H_j$ where

$$\begin{aligned} f &= r_1 \cdot q_1 + r_2 \cdot q_2 \\ &= X_0 \cdot \prod_{j=1}^{i}(X_n - c_j)^{v_j}\left(r_1 \cdot (X_n - c_i)^{w_i+1-v_i} + r_2 \cdot \prod_{j=1}^{i-1}(X_n - c_j)^{w_j+1-v_j}\right) \\ &\quad + \textit{the rest} \\ &= X_0 \cdot \prod_{j=1}^{i}(X_n - c_j)^{v_j}(r_1 \cdot pp1 + r_2 \cdot pp2) + \textit{the rest} \\ &= X_0 \cdot \prod_{j=1}^{i}(X_n - c_j)^{v_j} + \textit{the rest}. \end{aligned}$$

None monomial in *the rest* is greater than or equal to X_0 and the leading term of f is obviously $X_0 \cdot X_n^v$ which is equal to ed. Hence the **assertion** holds for arbitrary i.

Therefore we have proved that for any element $ed \in E(D)$, $f_{ed} := f$ vanishes on H and the leading term is ed. In the algorithm, we compute f_{ed} in turn according to the lexicographic order of the elements of $E(D)$. Once we get a polynomial, we use the polynomials obtained previously to reduce it (refer to Step 4.6). Now to prove the lower terms of the polynomial are all in D after such a reduction operation.

Let D be a lower set, a be a monomial, define $L(a, D) = \{b \in \mathbb{N}_0^n;\, b \prec a, b \in D\}$. Given any $d \notin D$, there exist only two situations: $d \in E(D)$ or $d \notin E(D)$ but $\exists d' \in E(D),\ s.t.\ d'$ is a factor of d. Of course $d' \prec d$.

Consider the sequence $\Re = \{T_1, T_2, T_3, \ldots\}$ of all the monomials with the elements of D discarded and all the elements are arranged according to the lexicographic order, use induction on it to prove that for every element T_t $(t > 0)$ we can construct

a vanishing polynomial with the leading term T_t and all the lower terms are in D, i.e. T_t can be represented as the linear combination of the elements of $L(T_t, D)$.

The very first vanishing polynomial we got in the algorithm is a univariate polynomial in X_n whose leading term is exactly T_1. It is obvious that the lower terms are in D, i.e. T_1 can be represented as the combination of the elements of $L(T_1, D)$.

Assume that T_{m-1} ($m \geq 2$) can be written as the combination of the elements of $L(T_{m-1}, D)$, now to prove it is true for T_m.

If $T_m \in E(D)$, the algorithm provides us a vanishing polynomial whose leading term is T_m, i.e. T_m can be represented as the combination of the terms which are all smaller than T_m. According to the induction assumption, for any lower term $T \notin D$ of the polynomial, T can be represented as the linear combination of the elements of $L(T, D)$, then T_m can be represented as the linear combination of the elements of $L(T_m, D)$.

If $T_m \notin E(D)$, there exists $d' \in E(D)$ s.t. $T_m = T'_m \cdot d'$. Since $d' \prec T_m$, according to the assumption, we can substitute d' with the linear combination of the elements of $L(d', D)$. Since all the elements in $L(d', D)$ are smaller than d', then T_m can be represented as the combination of elements which are all smaller than T_m. Then for the same reason described in the last paragraph, T_m can be represented as the linear combination of the elements of $L(T_m, D)$.

Therefore for every element T_t ($t > 0$) we can construct a vanishing polynomial with the leading term T_t and all the lower terms are in D. Particularly for any $ed \in E(D)$, all the lower terms of the polynomial f_{ed} we got in the algorithm after the reduction operation are in D. □

Remark 3 According to the proof of Theorem 3, f and f' in Step 4.6 for arbitrary i satisfy that $\mathrm{LC}_n(f) = \mathrm{LC}_n(f') = \prod_{j=1}^{i}(X_n - c_j)^{v_j}$.

Theorem 4 *Given a set of points H with multiplicity structures, $poly(H)$ is the reduced Gröbner basis of the vanishing ideal $I(H)$ and $D(H)$ is the quotient basis under lexicographic order.*

Proof Let m be the number of functionals defined by H and then $m = \dim(k[X]/I(H))$. Denote by J the ideal generated by $\mathrm{poly}(H)$. According to the **Assumption**, $\mathrm{poly}(H) \subseteq I(H)$. So $\dim(k[X]/I(H)) \leq \dim(k[X]/J)$. Let C be the set of the leading terms of polynomials in J under lexicographic order, then $C \supseteq \bigcup_{\beta \in E(D(H))}(\beta + \mathbb{N}_0^n)$ where the latter union is equal to $\mathbb{N}_0^n \backslash D(H)$. Then we can get $C' = \mathbb{N}_0^n \backslash C \subseteq D(H)$. Because $k[X]/J$ is isomorphic as a k-vector space to the k-span of C', here C' is viewed as a monomial set. So $\dim(k[X]/J) \leq \sharp D(H) = m$. Hence we have

$$m = \dim(k[X]/I(H)) \leq \dim(k[X]/J) \leq m.$$

Therefore $J = I(H)$, where $J = \langle \mathrm{poly}(H) \rangle$. Hence it is easy to know that $\mathrm{poly}(H)$ is exactly the reduced Gröbner basis of the vanishing ideal under lexicographic order, and $D(H)$ is the quotient basis. □

Based on Remark 3 and Theorem 4, we can naturally get the following lemma.

Lemma 1 *Assume G is the reduced Gröbner basis of some zero dimensional n-variable polynomial ideal under lexicographic order with $X_1 \succ X_2 \succ \cdots \succ X_n$. Define $p_0(G)$ as the univariate polynomial in X_n of G. View $g \in G$ as polynomial of $k(X_n)[X_1, \ldots, X_{n-1}]$ and define $LC_n(g)$ to be the leading coefficient of g which is a univariate polynomial in X_n and we have the conclusion that $LC_n(g)$ is always a factor of $p_0(G)$.*

Proof In fact for any given zero dimensional n-variable polynomial ideal, its reduced Gröbner basis G can be constructed from its zeros in the way our algorithm provides. Because the reduced Gröbner basis under lexicographic order is unique, Remark 3 holds for all the elements of G i.e. $\forall g \in G$, $\mathrm{LC}_n(g) = \prod_{j=1}^{s}(X_n - c_j)^{v_j}$ and particularly $p_0(G) = \prod_{j=1}^{s}(X_n - c_j)^{w_j+1}$ (refer the algorithm of **P-General case** for the symbols c_j, v_j, w_j). Because $v_j \leq w_j + 1$, $\mathrm{LC}_n(g)$ is a factor of $p_0(G)$. □

7 Intersection of Ideals

Based on Lemma 1 and the algorithm of **P-General case** in Sect. 6, we present a new algorithm named **Intersection** to compute the intersection of two ideals I_1 and I_2 satisfying that the greatest common divisor of $p_0(G_1)$ and $p_0(G_2)$ equals 1 where G_1 and G_2 are respectively the reduced Gröbner bases of I_1 and I_2 under the lexicographic order i.e. satisfying that the zeros of I_1 and that of I_2 does not share even one same X_n coordinate.

Denote by $Q(G)$ the quotient basis where G is the reduced Gröbner basis. The following algorithm **CPI** (short for *Computing the Polynomial for Intersection*) is a sub-algorithm called in algorithm **Intersection**.

Algorithm **CPI:** G is a reduced Gröbner basis, for any given monomial LT which is not in $Q(G)$, we get a polynomial p in $\langle G \rangle$ whose leading term is a factor of LT: the $X_1, \ldots, X_{n-1}$ components of the leading term are the same with that of LT and the X_n component has the lowest degree. Denoted by $p := \mathrm{CPI}(LT, G)$.

Step 1 $G' := \{g \in G |$ the leading monomial of g is a factor of $LT\}$.
Step 2 $G'' := \{g \in G' |$ the leading monomial of g has the smallest degree of X_n for that of all the elements in $G'\}$.
Step 3 Select one element of G'' and multiply it by a monomial of $X_1, \ldots, X_{n-1}$ to get p whose leading monomial is LT.

Algorithm **Intersection:** G_1 and G_2 are the reduced Gröbner bases of two different ideals satisfying that $\mathrm{GCD}(p_0(G_1), p_0(G_2)) = 1$. Return the reduced Gröbner basis of the intersection of these two ideals, denoted by $G := \mathrm{Intersection}(G_1, G_2)$.

Step 1 $D := Q(G_1) + Q(G_2)$. View $E(D)$ as a monomial set. $G := \varnothing$.

Step 2 If $E(D) = \varnothing$, the algorithm is done. Else select the minimal element of $E(D)$, denoted by T. $E(D) := E(D)\backslash\{T\}$.

Step 3

$$f_1 := \mathrm{CPI}(T, G_1),\ f_2 := \mathrm{CPI}(T, G_2).$$

$$q_1 := f_1 \cdot p_2,\ q_2 := f_2 \cdot p_1.$$

Step 4

$$t_1 := \frac{p_0(G_2)}{\mathrm{LC}_n(f_2)},\ t_2 := \frac{p_0(G_1)}{\mathrm{LC}_n(f_1)}.$$

Step 5 Use Extended Euclidean Algorithm to find r_1, r_2 s.t.

$$r_1 \cdot t_1 + r_2 \cdot t_2 = 1.$$

Step 6 $f := q_1 \cdot r_1 + q_2 \cdot r_2$. Reduce f with G to get f', and $G := G \bigcup \{f'\}$. Go to Step 2.

Remark 4 This algorithm is essentially the same with Step 4.1–Step 4.7 of **P-General case** in Sect. 6, so it is obvious that r_1, r_2 in Step 5 do exist and the polynomials in the algorithm are all well defined. Besides, D in Step 1 is not empty, so it is easy to know the result of this algorithm can never be empty.

Because this algorithm is essentially the same with Step 4.1–Step 4.7 of **P-General case** in Sect. 6, here we omit the proof. And in return, the algorithm of **P-General case** in Sect. 6 can be simplified according to this **Intersection** algorithm: we can delete the last sentence in Step 2 and replace Step 4.3 and Step 4.4 respectively by:

Step 4.3′

$$f_1 := \mathrm{CP}(LT, D, G),\ f_2 := \mathrm{CP}(LT, D_i', G_i).$$

Step 4.4′

$$q_1 := f_1 \cdot p_0(G_i);\ \ q_2 := f_2 \cdot p_0(G).$$

$$pp1 := \frac{p_0(G_i)}{\mathrm{LC}_n(f_2)};\ \ pp2 := \frac{p_0(G)}{\mathrm{LC}_n(f_1)}.$$

8 Conclusion

During the induction of the algorithm in Sect. 6, we can record the leading coefficients for later use to save the computation cost and the computation cost is mainly on the Extended Euclidean Algorithm. But it's hard to compute how many times we need to use the Extended Euclidean Algorithm for a given problem, and the computation cost depends closely on the structures of the given set of points.

The benefit is the explicit geometric interpretation. For any given point with multiplicity structure $\langle p, D\rangle$, we put the lower set D into the affine space with the $(0, 0)$ element situated at p to intuitively represent it, and it can be imagined as $\sharp D_i$ small balls in the affine space. Given a set of points with multiplicity structures in n dimensional space, in the way showed in Sect. 2, we push the balls first along X_1-axis, then along X_2-axis, and so on, at last along X_n-axis to finally get a lower set which turns out to be exactly the quotient basis under the lexicographic order with $X_1 \succ X_2 \succ \cdots \succ X_n$. In the future, we will try to apply this geometric interpretation to our previous work on Birkhoff problem [14].

The geometric interpretation in this paper reveals the essential connection between the relative position of the points with multiplicity structures and the quotient basis of the vanishing ideal. It provides us a new perspective of view to look into the vanishing ideal problem and helps study the structure of the reduced Gröbner basis of zero dimensional ideal under lexicographic order. The new algorithm **Intersection** which computes the intersection of two ideals and Lemma 1 are the direct byproducts of our algorithm. Lemma 1 reveals important property of the reduced Gröbner basis under lexicographic order, which is necessary for a set of polynomials to be a reduced Gröbner basis. Lemma 1 can also help us to solve the polynomial system. It is well-known that the Gröbner basis of an ideal under lexicographic order holds good algebraic structures and hence is convenient to use for polynomial system solving [15]. Once we get the reduced Gröbner basis G of a zero dimensional ideal, to solve the polynomial system, we need first compute the roots of $p_0(G)$. Since $\mathrm{LC}_n(g)$ $(g \neq p_0(G),\ g \in G)$ is a factor of $p_0(G)$, computing the roots of $\mathrm{LC}_n(g)$ which has a smaller degree would be helpful for saving the computation cost.

Lederer [13] presented an algorithm to compute the reduced Gröbner basis of the vanishing ideal over a set of points with no multiplicity structures. The author splits the problem into several small ones and combines the results of the small problems by using Lagrange interpolation method to get the final result and the idea really inspired us a lot. Because the problem considered here concerning the points with multiplicity structures, we have to consider **P-Special case** and **P-General case**, and the Lagrange interpolation method is not available any more, we use the Extended Euclidean Algorithm instead.

References

1. Stetter, H.J.: Numerical Polynomial Algebra. SIAM, Philadelphia (2004)
2. Pistone, G., Riccomagno, E., Wynn, H.P.: Algebraic Statistics: Computational Commutative Algebra in Statistics. Monographs on statistics & applied probability. Chapman & Hall/CRC, Boca Raton (2001)
3. Laubenbacher, R., Stigler, B.: A computational algebra approach to the reverse engineering of gene regulatory networks. J. Theor. Biol. **229**, 523–537 (2004)
4. Guruswami, V., Sudan, M.: Improved decoding of Reed-Solomon and algebraic geometric codes. IEEE Trans. Inf. Theory **46**, 1757–1767 (1999)
5. Koetter, R., Vardy, A.: Algebraic soft-decision decoding of Reed-Solomon codes. IEEE Trans. Inf. Theory **49**, 2809–2825 (2003)

6. Sudan, M.: Decoding of Reed-Solomon codes beyond the error-correction bound. J. Complex. **13**, 180–193 (1997)
7. Buchberger, B., Möller, H.M.: The construction of multivariate polynomials with preassigned zeros. In: Computer Algebra, EUROCAM'82. Lecture Notes in Computer Science, vol. 144, Springer, Berlin, pp. 24–31 (1982)
8. Marinari, M.G., Möller, H.M., Mora, T.: Gröbner basis of ideals defined by functionals with an application to ideals of projective points. Appl. Algebra Eng. Commun. Comput. **4**, 103–145 (1993)
9. Abbott, J., Bigatti, A., Kreuzer, M., Robbiano, L.: Computing ideals of points. J. Symb. Comput. **30**, 341–356 (2000)
10. Farr, J., Gao, S.: Computing Gröbner bases for vanishing ideals of finite sets of points. In: Proceedings of the Conference, ACA (2003)
11. Felszeghy, B., Ráth, B., Rónyai, L.: The lex game and some applications. J. Symb. Comput. **41**, 663–681 (2006)
12. Cerlienco, L., Mureddu, M.: From algebraic sets to monomial linear bases by means of combinatorial algorithms. Discret. Math. **139**, 73–87 (1995)
13. Lederer, M.: The vanishing ideal of a finite set of closed points in affine space. J. Pure Appl. Algebra **212**, 1116–1133 (2008)
14. Lei, N., Chai, J., Xia, P., Li, Y.: A fast algorithm for the multivariate Birkhoff interpolation problem. J. Comp. Appl. Math. **236**, 1656–1666 (2011)
15. Cox, D., Little, J., OShea, D.: Ideals, Varieties, and Algorithms. Springer, New York (1997)