On $\boldsymbol{\mu}$-Symmetric Polynomials and D-plus

Jing Yang[1] ⋆ and Chee K. Yap[2] ⋆⋆

[1] SMS-HCIC, Guangxi University for Nationalities, China
Email: yangjing0930@gmail.com
[2] Courant Institute of Mathematical Sciences
New York University, USA.
Email: yap@cs.nyu.edu

Abstract. We study functions of the roots of a univariate polynomial of degree $n \geq 1$ in which the roots have a given multiplicity structure $\boldsymbol{\mu}$, denoted by a partition of n. For this purpose, we introduce a theory of $\boldsymbol{\mu}$-symmetric polynomials which generalizes the classic theory of symmetric polynomials. We designed three algorithms for checking if a given root function is $\boldsymbol{\mu}$-symmetric: one based on Gröbner bases, another based on preprocessing and reduction, and the third based on solving linear equations. Experiments show that the latter two algorithms are significantly faster. We were originally motivated by a conjecture about the $\boldsymbol{\mu}$-symmetry of a certain root function $D^+(\boldsymbol{\mu})$ called D-plus. This conjecture is proved to be true. But prior to the proof, we studied the conjecture experimentally using our algorithms.

1 Introduction

Suppose $P(x) \in \mathbb{Z}[x]$ is a polynomial with m distinct complex roots $r_1, \ldots, r_m$ where r_i has multiplicity μ_i. Write $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$ where we may assume $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_m$. Thus $n = \sum_{i=1}^m \mu_i$ is the degree of $P(x)$. Consider the following function of the roots

$$D^+(P(x)) := \prod_{1 \leq i < j \leq m} (r_i - r_j)^{\mu_i + \mu_j}.$$

Call this the **D-plus** root function. This root function[3] was introduced by Becker et al [2] in their complexity analysis of a root clustering algorithm. The original motivation of this paper was to try to prove that $D^+(P(x))$ is a rational function in the coefficients of $P(x)$.

We may write "$D^+(\boldsymbol{\mu})$" instead of $D^+(P(x))$ since the expression in terms of roots $\mathbf{r}$ depends only on the multiplicity structure $\boldsymbol{\mu}$. For example, if $\boldsymbol{\mu} = (2, 1)$ then $D^+(\boldsymbol{\mu}) = (r_1 - r_2)^3$ and this turns out to be $\left[a_1^3 - (9/2)a_0 a_1 a_2 + (27/2)a_0^2 a_3\right]/a_0^3$ where $P(x) = \sum_{i=0}^3 a_{3-i} x^i$. More generally, for any function $F(\mathbf{r}) = F(r_1, \ldots, r_m)$,

⋆ Jing's work is supported by the Special Fund for Guangxi Bagui Scholars (WBS 2014-01) and the Startup Foundation for Advanced Talents in Guangxi University for Nationalities (2015MDQD018)

⋆⋆ Chee's work is supported by Guangxi University for Nationalities and by NSF Grant # CCF-1564132.

[3] In [2], the D-plus function was called a generalized discriminant.

we ask whether evaluating F at the m distinct roots of a polynomial $P(x)$ with multiplicity structure $\boldsymbol{\mu}$ is rational in the coefficients of $P(x)$. In case $P(x)$ has only simple roots, the Fundamental Theorem of Symmetric Functions tells us the complete answer: $F(\mathbf{r})$ is rational iff $F(\mathbf{r})$ is a symmetric polynomial. We extend this theorem to the case of non-simple roots: if the roots of $P(x)$ have multiplicity structure $\boldsymbol{\mu}$, then we define what it means for $F(\mathbf{r})$ to be $\boldsymbol{\mu}$-symmetric. As expected, this characterizes when $F(\mathbf{r})$ is rational in the coefficients of $P(x)$. It is non-trivial to check if any given root function F (in particular $F = D^+(\boldsymbol{\mu})$) is $\boldsymbol{\mu}$-symmetric. We will design three algorithms for this task. Although we feel that $\boldsymbol{\mu}$-symmetry is a natural concept, to our knowledge, this has not been systematically studied before.

Overview of Paper. In Section 2, we define $\boldsymbol{\mu}$-symmetric polynomials and show some preliminary properties of such polynomials. Then three algorithms for checking $\boldsymbol{\mu}$-symmetry are given in Sections 3-5. Section 6 proves the $\boldsymbol{\mu}$-symmetry of $D^+(\boldsymbol{\mu})$. In Section 7, we show experimental results from our Maple implementation of the three algorithms. We conclude in Section 8.

The full version of this paper includes 3 appendices: A: Maple source code, B: Description of benchmark polynomials, and C: All the proofs. may be downloaded from http://cs.nyu.edu/exact/papers/.

2 $\boldsymbol{\mu}$-Symmetric Polynomials

Throughout, assume K is a field of characteristic 0. For our purposes, $K = \mathbb{Q}$ will do. We fix three sequences of variables $\mathbf{x} = (x_1, \ldots, x_n)$, $\mathbf{z} = (z_1, \ldots, z_n)$ and $\mathbf{r} = (r_1, \ldots, r_m)$ where $n \geq m \geq 1$.

Let $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$ be a partition of n where $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_m \geq 1$. We may denote this relation by $\boldsymbol{\mu} \vdash n$. We call $\boldsymbol{\mu}$ an m**-partition** if it has exactly m parts. A **specialization** σ is any function of the form $\sigma : \{x_1, \ldots, x_n\} \to \{r_1, \ldots, r_m\}$. We say σ is of **type** $\boldsymbol{\mu}$ if $|\sigma^{-1}(r_i)| = \mu_i$ for $i = 1, \ldots, m$. We say σ is **canonical** if $\sigma(x_i) = r_j$ and $\sigma(x_{i+1}) = r_k$ implies $j \leq k$. Clearly the canonical specialization of type $\boldsymbol{\mu}$ is unique, and we may denote it by $\sigma_{\boldsymbol{\mu}}$.

Consider the polynomial rings $K[\mathbf{x}]$ and $K[\mathbf{r}]$. Any specialization $\sigma : \{x_1, \ldots, x_r\} \to \{r_1, \ldots, r_m\}$ can be extended naturally into a K-homomorphism $\sigma : K[\mathbf{x}] \to K[\mathbf{r}]$ where $P = P(\mathbf{x}) \in K[\mathbf{x}]$ is mapped to $\sigma(P) = P(\sigma(x_1), \ldots, \sigma(x_n))$. When σ is understood, we may write "$\overline{P}$" for the homomorphic image $\sigma(P)$.

We denote the i**-th elementary symmetric functions** ($i = 1, \ldots, n$) in $K[\mathbf{x}]$ by $e_i = e_i(\mathbf{x})$. E.g., $e_1 := \sum_{i=1}^n x_i$, $e_2 := \sum_{1 \leq i < j \leq n} x_i x_j$, $\ldots$, $e_n := \prod_{i=1}^n x_i$. Also define $e_0 := 1$. Typically, we write $\overline{e}_i$ for $\sigma_{\boldsymbol{\mu}}(e_i)$ when $\boldsymbol{\mu}$ is understood.

The key definition is the following: a polynomial $F \in K[\mathbf{r}]$ is said to be $\boldsymbol{\mu}$**-symmetric** if there is a symmetric polynomial $\widehat{F} \in K[\mathbf{x}]$ such that $\sigma_{\boldsymbol{\mu}}(\widehat{F}) = F$ where $n = \sum_{i=1}^m \mu_i$. We call $\widehat{F}$ the $\boldsymbol{\mu}$**-lift** (or simply "lift") of F. If $\mathring{F} \in K[\mathbf{z}]$ satisfies $\mathring{F}(e_1, \ldots, e_n) = \widehat{F}(\mathbf{x})$ then we call $\mathring{F}$ the $\boldsymbol{\mu}$**-kernel** of F.

Remarks 1. Note that the $\boldsymbol{\mu}$-lift of F is defined if and only if F is $\boldsymbol{\mu}$-symmetric.

2. We view the z_i's as symbolic representation of $e_i(\mathbf{x})$'s.
3. Although $\widehat{F}$ and $\mathring{F}$ are mathematically equivalent, the kernel concept lends itself to direct evaluation based on coefficients of $P(x)$.

The Fundamental Theorem on Symmetric Functions implies the following:

Lemma 1. *If $f(\mathbf{r}) \in K[\mathbf{r}]$ is $\boldsymbol{\mu}$-symmetric, then for any $P(x) = \sum_{i=1}^{n} c_i x^i \in K[x]$ of degree n, if P has m distinct roots $\rho_1, \ldots, \rho_m$ with multiplicity $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$, $F(\rho_1, \ldots, \rho_m) \in K$.*

We want to study the lift $\widehat{F} \in K[\mathbf{x}]$ of a $\boldsymbol{\mu}$-symmetric polynomial $F \in K[\mathbf{r}]$ of total degree δ. If we write F as the sum of its homogeneous parts, $F = F_1 + \cdots + F_\delta$, then $\widehat{F} = \widehat{F}_1 + \cdots + \widehat{F}_\delta$. Hence, we may restrict F to be homogeneous.

Next consider a polynomial $G(\mathbf{z}) \in K[\mathbf{z}]$. Suppose there is a **weight function**

$$\omega : \{z_1, \ldots, z_n\} \to \mathbb{N} = \{1, 2, \ldots\}$$

then for any term $t = \prod_{i=1}^{n} z_i^{e_i}$, its **$\omega$-degree** is $\sum_{i=1}^{n} e_i \omega(z_i)$. Normally, $\omega(z_i) = 1$ for all i; but in this paper, we are also interested in the weight function where $\omega(z_i) = i$. For short, we simply call this ω-degree of t its **weighted degree**. E.g., the weighted degree of $z_1^2 z_3$ is 5. The weighted degree of a polynomial $G(\mathbf{z})$ is just the maximum weighted degree of terms in its support. A polynomial $G(\mathbf{z})$ is said to be **weight homogeneous** if all of its terms have the same weighted degree. Note that the kernel $\mathring{F}$ of F is not unique: for any kernel $\mathring{F}$, we can decompose it as $\mathring{F} = \mathring{F}_0 + \mathring{F}_1$ where $\mathring{F}_0$ is the weight homogeneous part of $\mathring{F}$ of weighted degree δ, and $\mathring{F}_1 := \mathring{F} - \mathring{F}_0$. Then $\mathring{F}(\bar{e}_1, \ldots, \bar{e}_n) = F$ implies that $\mathring{F}_0(\bar{e}_1, \ldots, \bar{e}_n) = F$ and $\mathring{F}_1(\bar{e}_1, \ldots, \bar{e}_n) = 0$. We can always omit $\mathring{F}_1$ from the kernel of F. We shall call any polynomial $G(\mathbf{z}) \in K[\mathbf{z}]$ a **$\boldsymbol{\mu}$-constraint** if $G(\bar{e}_1, \ldots, \bar{e}_n) = 0$. Thus, $\mathring{F}_1$ is a $\boldsymbol{\mu}$-constraint. We may check that the set of $\boldsymbol{\mu}$-constraints forms an ideal in $K[\mathbf{z}]$ which we call the **$\boldsymbol{\mu}$-ideal**.

3 Computing Kernels via Gröbner Bases

In this section, we consider a Gröbner basis algorithm to compute the $\boldsymbol{\mu}$-kernel of a given polynomial $F \in K[\mathbf{r}]$, or detect that it is not $\boldsymbol{\mu}$-symmetric. For this purpose, define the following ideal:

$$\mathcal{I}_{\boldsymbol{\mu}} := \langle v_1, \ldots, v_n \rangle \tag{1}$$

where $v_i := z_i - \bar{e}_i$ $(i = 1, \ldots, n)$. Note that $\mathcal{I}$ is an ideal in $K[\mathbf{z}, \mathbf{r}]$. Moreover, we define $\mathcal{G}_{\boldsymbol{\mu}}$ to be the Gröbner basis of $\mathcal{I}_{\boldsymbol{\mu}}$ relative to the term ordering where $z_i \prec r_j$ for all i and j. The following is a generalization of Proposition 4 in [3, Chapter 7, Section 1].

Theorem 1. *Let $R \in K[\mathbf{r}, \mathbf{z}]$ be the normal form of $F \in K[\mathbf{r}]$ relative to $\mathcal{G}_{\boldsymbol{\mu}} \subseteq K[\mathbf{r}, \mathbf{z}]$. Then F is $\boldsymbol{\mu}$-symmetric iff $R \in K[\mathbf{z}]$. Moreover, if $R \in K[\mathbf{z}]$ then R is the $\boldsymbol{\mu}$-kernel of F.*

Theorem 1 leads to the following algorithm.

```
G-kern($F, \boldsymbol{\mu}$):
    Input:  $F \in K[\mathbf{r}]$ and $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$.
    Output: the $\boldsymbol{\mu}$-kernel of $F$ or say "$\overset{\circ}{F}$ does not exist"
        $\mathcal{B} \leftarrow \{z_1 - \overline{e}_1(\mathbf{r}), \ldots, z_n - \overline{e}_n(\mathbf{r})\}$
        $ord \leftarrow plex(r_m, \ldots, r_1, z_n, \ldots, z_1)$
        $\mathcal{G} \leftarrow GroebnerBasis(\mathcal{B}, ord)$
        $R \leftarrow NormalForm(F, \mathcal{G}, ord)$
        If $\deg(R, \mathbf{r}) > 0$ then
            Return "$\overset{\circ}{F}$ does not exist"
        Return $R$
```

4 Checking $\boldsymbol{\mu}$-symmetry via Preprocessing and Reduction

In the previous section, we show how to compute $\boldsymbol{\mu}$-kernels using Gröbner bases. This algorithm is quite slow when $\boldsymbol{\mu} \neq (1, 1, \ldots, 1)$. In the next two sections, we will introduce two methods based on an analysis of the following two K-vector spaces:

- $K^{\delta}_{\texttt{sym}}[\mathbf{x}]$: the set of symmetric homogeneous polynomials of degree δ in $K[\mathbf{x}]$
- $K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$: the set of $\boldsymbol{\mu}$-symmetric polynomials of degree δ in $K[\mathbf{r}]$

The first method is based on preprocessing and reduction: we first compute a basis for $K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$, and then use the basis to reduce $F(\mathbf{r})$. The second method directly computes the $\boldsymbol{\mu}$-kernel of $F(\mathbf{r})$ by solving linear equations.

First consider $K^{\delta}_{\texttt{sym}}[\mathbf{x}]$. By a **weak partition** of δ, we mean $\alpha = (\alpha(1), \alpha(2), \ldots, \alpha(\delta))$ where $\alpha(1) \geq \alpha(2) \geq \cdots \geq \alpha(\delta) \geq 0$ and $\alpha = \sum \alpha(i) = \delta$. Note that $\alpha(i)$ can be 0 in weak partitions. If α is a weak partition of δ with no part $\alpha(i)$ larger than n, we will write $\alpha \vdash (\delta, n)$. Let $e_\alpha := \prod_{i=1}^{\delta} e_{\alpha(i)}$. E.g., if $\delta = 4, n = 2, \alpha = (2, 1, 1, 0)$ then $e_\alpha = e_2 e_1^2 e_0 = e_2 e_1^2$.

Let $T(\mathbf{x})$ denote the set of terms of $\mathbf{x}$, and $T^\delta(\mathbf{x})$ denote those terms of degree δ. A typical element of $T^\delta(\mathbf{x})$ is $\prod_{i=1}^{n} x_i^{e_i}$ where $e_1 + \cdots + e_n = \delta$. We totally order the terms in $T^\delta(\mathbf{x})$ using the lexicographic ordering in which $x_1 \prec x_2 \prec \cdots \prec x_n$. Given any $F \in K(\mathbf{x})$, its **support** is $\texttt{Supp}(F) \subseteq T(\mathbf{x})$ such that F can be uniquely written as

$$F = \sum_{p \in \texttt{Supp}(F)} c(p) p \tag{2}$$

where $c : \texttt{Supp}(F) \to K \setminus \{0\}$ denote the coefficients of F. Let the **leading term** $\texttt{Lt}(F)$ be equal to the $p \in \texttt{Supp}(F)$ which is the largest under the lexicographic ordering. For instance, $\texttt{Supp}(e_1) = \{x_1, \ldots, x_n\}$ and $\texttt{Lt}(e_1) = x_n$. Also $\texttt{Supp}(e_1 e_2) = \{x_i x_j x_k : 1 \leq i \neq j \leq n, 1 \leq k \leq n\}$ and $\texttt{Lt}(e_1 e_2) = x_n^2 x_{n-1}$. The coefficient of $\texttt{Lt}(F)$ in F is the **leading coefficient** of F, denoted by $\texttt{Lc}(F)$. Call $\texttt{Lm}(F) := \texttt{Lc}(F)\texttt{Lt}(F)$ the **leading monomial** of F. This is well-known:

Proposition 2 *The set $\mathcal{B}^{\delta}_{n} := \{e_\alpha : \alpha \vdash (\delta, n)\}$ is a K-basis for $K^{\delta}_{\texttt{sym}}[\mathbf{x}]$.*

Next consider the set $K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$ comprising the $\boldsymbol{\mu}$-symmetric functions of degree δ. The map $\sigma_{\boldsymbol{\mu}} : K^{\delta}_{\mathsf{sym}}[\mathbf{x}] \to K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$ is an onto K-homomorphism. Thus $K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$ is a vector space which is generated by the set $\sigma_{\boldsymbol{\mu}}(\mathcal{B}^{\delta}_{n}) := \left\{\sigma_{\boldsymbol{\mu}}(G) : G \in \mathcal{B}^{\delta}_{n}\right\}$. It follows that there is a maximal independent set $\overline{\mathcal{B}}^{\delta}_{n} \subseteq \sigma_{\boldsymbol{\mu}}(\mathcal{B}^{\delta}_{n})$ that is a basis for $K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$. The set $\overline{\mathcal{B}}^{\delta}_{n}$ may be a proper subset of $\sigma_{\boldsymbol{\mu}}(\mathcal{B}^{\delta}_{n})$.

Now we generate the basis of the vector space $K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$ with which one could easily check whether a given polynomial is in this vector space or not. For this purpose, we introduce a reduction procedure and its applications. A set $\mathcal{B} \subseteq K[\mathbf{r}]$ is **linearly independent** if any non-trivial K-linear combination over $\mathcal{B}$ is non-zero; otherwise, $\mathcal{B}$ is **linearly dependent**. We say $\mathcal{B}$ is **canonical** if $\mathcal{B}$ is linearly independent and ordered as $\mathcal{B} = (B_1, \ldots, B_\ell)$ with $\mathtt{Lt}(B_i) \prec \mathtt{Lt}(B_j)$ for all $i < j$.

Given a polynomial $F \in K[\mathbf{r}]$, we say it is **reduced** relative to $\mathcal{B} = (B_1, \ldots, B_\ell)$ if $\mathtt{Lt}(B_i) \notin \mathtt{Supp}(F)$ for each $i = 1, \ldots, \ell$. We can reduce F relative to $\mathcal{B}$ by subtracting from F a linear combination of elements in $\mathcal{B}$ as shown in Figure 1.

```
reduce($F, \mathcal{B}$):
    Input:  $F \in K^{\delta}[\mathbf{r}]$, $\mathcal{B} = (B_1, \ldots, B_\ell)$ is canonical and each $B_i \in K^{\delta}[\mathbf{r}]$
    Output: $R$ such that $F = \sum_{i=1}^{\ell} c_i B_i + R$ with $c_i \in K$ and
            $R$ is reduced relative to $\mathcal{B}$.
        Let $R \leftarrow 0$, $i \leftarrow \ell$
        If $\mathcal{B} = ()$ then
            Return $F$
        While ($F \neq 0$)
            $p \leftarrow \mathtt{Lt}(F)$
            If $p = \mathtt{Lt}(B_i)$ then
                $F \leftarrow F - \frac{\mathtt{Lc}(F)}{\mathtt{Lc}(B_i)} B_i$; $i \leftarrow i - 1$
            elseif $p \succ \mathtt{Lt}(B_i)$ then
                $R \leftarrow R + \mathtt{Lc}(F) \cdot p$; $F \leftarrow F - \mathtt{Lc}(F) \cdot p$
            else
                $i \leftarrow i - 1$
            If $i = 0$ then Return $R + F$
        Return $R$
```

Fig. 1. The `reduce` algorithm.

The termination of the `reduce` algorithm is guaranteed by the following:

Lemma 3. *The algorithm* $\mathtt{reduce}(F, \mathcal{B})$ *takes at most* $\#\mathtt{Supp}(F) + \sum_{i=1}^{\ell} \#\mathtt{Supp}(B_i) - 1$ *loops. Moreover, this bound is tight in the worst case.*

It is easy to see that $\mathtt{reduce}(F, \mathcal{B}) = 0$ iff $\mathcal{B} \cup \{F\}$ is linearly dependent. This gives rise to the `canonize` algorithm in Figure 2 for constructing a canonical set from any set $\mathcal{B} \subseteq K[\mathbf{r}]$. Clearly $\mathtt{canonize}(\mathcal{B})$ terminates in $|\mathcal{B}|$ loops. Finally, we use `reduce` and `canonize` algorithms to construct the `isMuSymmetric` algorithm for checking the $\boldsymbol{\mu}$-symmetry of a polynomial.

Lemma 4. *The algorithm* $\mathtt{isMuSymmetric}(F, \boldsymbol{\mu})$ *halts. Moreover, it outputs* "Yes" *iff F is $\boldsymbol{\mu}$-symmetric.*

```
canonize($\mathcal{B}$):
  Input:  $\mathcal{B} \subseteq K[\mathbf{r}]$.
  Output: a maximal canonical $\mathcal{C} \subseteq \mathcal{B}$
    Let $\mathcal{C} \leftarrow ()$ (empty sequence)
    While $\mathcal{B} \neq \emptyset$
        $B \leftarrow pop(\mathcal{B})$
        $B' \leftarrow reduce(B, \mathcal{C})$
        If $B' \neq 0$ then
            $\mathcal{C} \leftarrow prepend(B', \mathcal{C})$
            $\mathcal{C} \leftarrow sort(\mathcal{C})$
    Return $\mathcal{C}$
```

```
isMuSymmetric($F, \boldsymbol{\mu}$):
  Input:  $F \in K[\mathbf{r}]$, $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$
  Output: Yes if $F$ is $\boldsymbol{\mu}$-symmetric;
          otherwise return No.
    $\delta \leftarrow \deg(F, \mathbf{r})$
    $n \leftarrow \sum_{i=1}^{m} \mu_i$
    $\mathcal{B} \leftarrow \{\bar{e}_\alpha : \alpha \vdash (\delta, n)\}$
    $\mathcal{C} \leftarrow canonize(\mathcal{B})$
    If reduce($F, \mathcal{C}$) $= 0$ then
        Return "Yes"
    Return "No"
```

Fig. 2. The `canonize` and `isMuSymmetric` algorithms.

5 Computing Kernels via Solving Linear Systems

We now outline a method to compute the kernel of $F(\mathbf{r})$ by solving a linear system of equations.

Recall that $F \in K[\mathbf{r}]$ is $\boldsymbol{\mu}$-symmetric iff there is a $\mathring{F} \in K[\mathbf{z}]$ such that $\mathring{F}(\bar{e}_1, \ldots, \bar{e}_n) = F$. We propose to first write $\mathring{F}(\mathbf{z})$ as an indeterminate polynomial $G(\mathbf{k}; \mathbf{z}) \in K[\mathbf{k}][\mathbf{z}]$ which has homogeneous weighted degree δ with indeterminate coefficients $\mathbf{k}$. Each term of weighted degree δ has the form $\mathbf{z}_\alpha := \prod_{i=1}^{\delta} z_{\alpha(i)}$ where $\alpha = (\alpha(1), \ldots, \alpha(\delta))$ is a weak partition of δ with parts at most n, i.e., $\alpha \vdash (\delta, n)$. Let the set of all such partitions be denoted $I_n^\delta := \{\alpha : \alpha \vdash (\delta, n)\}$ Then $G(\mathbf{k}; \mathbf{z})$ can be written as $G(\mathbf{k}; \mathbf{z}) := \sum_{\alpha \in I_n^\delta} k_\alpha \mathbf{z}_\alpha$ where each k_α is an indeterminate. Here, $\mathbf{k} := (k_\alpha : \alpha \in I_n^\delta)$. Next, we plug in $\bar{e}_i$'s for the z_i's to get $H(\mathbf{k}; \mathbf{r}) := G(\mathbf{k}; \bar{e}_1, \ldots, \bar{e}_n)$ which we view as a polynomial in $K[\mathbf{k}][\mathbf{r}]$. We then set up the equation

$$H(\mathbf{k}; \mathbf{r}) = F(\mathbf{r}) \tag{3}$$

to solve for the values of $\mathbf{k}$. Note that total degree of G in $\mathbf{k}$ is 1, i.e., $\deg(G, \mathbf{k}) = 1$. Therefore, $\deg(H, \mathbf{k}) = 1$. Thus (3) amounts to solving a linear system of equations in $\mathbf{k}$. The above procedure can be summarized as the `E-kern` algorithm.

6 The $\boldsymbol{\mu}$-symmetry of $D^+(\boldsymbol{\mu})$

Recall the definition of D^+ given in Section 1: $D^+(\boldsymbol{\mu}) = \prod_{1 \leq i < j \leq m} (r_i - r_j)^{\mu_i + \mu_j}$. We say D^+ is *injective* on an argument $\boldsymbol{\mu}$ if for all $\boldsymbol{\mu}' \neq \boldsymbol{\mu}$, $D^+(\boldsymbol{\mu}) \neq D^+(\boldsymbol{\mu}')$. Clearly, D^+ is not injective on any $\boldsymbol{\mu}$ of the form $\boldsymbol{\mu} = (\mu_1, \mu_2)$: in this case, $D^+ = (r_1 - r_2)^n$ for any $\boldsymbol{\mu} = (\mu_1, \mu_2)$ where $\mu_1 + \mu_2 = n$.

Lemma 5. *D^+ is injective on any $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$ where $m > 2$.*

We next introduce a $\boldsymbol{\mu}$-symmetric function Δ that is useful in the study of the $\boldsymbol{\mu}$-symmetry of D^+. It is closely related to the notion of subdiscriminants [1,

```
E-kern($F, \boldsymbol{\mu}$):
    Input:  $F \in K[\mathbf{r}]$ and $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$
    Output: the kernel of $F$ if $F$ is $\boldsymbol{\mu}$-symmetric;
            otherwise, return "$F$ is not $\boldsymbol{\mu}$-symmetric".
        $\delta \leftarrow \deg(F, \mathbf{r})$; $n \leftarrow \sum_{i=1}^{m} \mu_i$
        Construct the index set $I_n^{\delta}$.
        $G \leftarrow \sum_{\alpha \in I_n^{\delta}} k_\alpha z_\alpha$
        $H \leftarrow G(\mathbf{k}; \overline{e}_1, \ldots, \overline{e}_n)$
        Extract Coeffs($H, \mathbf{r}$) and Coeffs($F, \mathbf{r}$).
        Find a solution $\mathbf{k} = \mathbf{k}_0$ of the linear system
                Coeffs($H, \mathbf{r}$) = Coeffs($F, \mathbf{r}$).
        If $\mathbf{k}_0$ is nondefined
            Return "$F$ is not $\boldsymbol{\mu}$-symmetric"
        Return $G(\mathbf{k}_0; \mathbf{z})$
```

Section 4.1]. First, we need some notations: let $[n] := \{1, \ldots, n\}$, and $\binom{[n]}{k}$ denote the set of k-subsets of $[n]$. For $k = 0, \ldots, n-2$, we may define the function

$$S_k^n = S_k^n(\mathbf{x}) := \sum_{I \in \binom{[n]}{n-k}} \prod_{i \neq j \in I} \Big(x_i - x_j\Big)^2 \tag{4}$$

called the k**th subdiscriminant** in n variables. We may also define $S_{n-1}^n := 1$. When $k = 0$, we have $S_0^n = \prod_{i \neq j \in [n]} \Big(x_i - x_j\Big)^2$. If the x_i's are roots of a polynomial $P(x)$ of degree n, then S_0^n is the standard discriminant of $P(x)$. Clearly S_k^n is a symmetric polynomial in $\mathbf{x}$.

Lemma 6. *Define $\Delta := \prod_{1 \le i < j \le m} (r_i - r_j)^2$.*
(a) Δ is $\boldsymbol{\mu}$-symmetric with lift given by $\widehat{\Delta} = \frac{1}{\prod_{i=1}^m \mu_i} \cdot S_{n-m}^n$ where S_{n-m}^n is the $(n-m)$-th subdiscriminant.
(b) When $m = 2$, we have an explicit formula for the lift of Δ: with $n = \mu_1 + \mu_2$,

$$\widehat{\Delta} = \frac{(n-1)e_1^2 - 2ne_2}{\mu_1 \mu_2}.$$

We now prove the conjecture for special cases for arbitrary n. First, consider the case where $\boldsymbol{\mu} = (a, a, \ldots, a)$.

Theorem 2. *If all μ_i's are equal to a, then $D^+(\boldsymbol{\mu})$ is $\boldsymbol{\mu}$-symmetric with lift given by $\widehat{F}_n(\mathbf{x}) = \left(\frac{1}{a^m} \cdot S_{n-m}^n\right)^a$ where S_{n-m}^n is given by Lemma 6(a).*

Another special case of $D^+(\boldsymbol{\mu})$ is when $m = 2$:

Theorem 3. *For all $\boldsymbol{\mu} = (\mu_1, \mu_2)$, $D^+(\boldsymbol{\mu})$ has a $\boldsymbol{\mu}$-kernel $\mathring{F}_n$ where $\boldsymbol{\mu} \vdash n$:*

- *n is even: $\mathring{F}_n = \left(\frac{(n-1)z_1^2 - 2nz_2}{\mu_1 \mu_2}\right)^{n/2}$*

- n *is odd:* $\mathring{F}_n = \left(\frac{(n-1)z_1^2 - 2nz_2}{\mu_1\mu_2}\right)^{\frac{n-3}{2}} \left(k_1 z_1^3 + k_2 z_1 z_2 + k_3 z_3\right)$
  where $(k_1, k_2, k_3) = \left(\frac{-(n-1)(n-2)}{d}, \frac{3n(n-2)}{d}, \frac{-3n^2}{d}\right)$ *and* $d = \mu_1\mu_2(\mu_1 - \mu_2)$.

Now we prove the $\boldsymbol{\mu}$-symmetry of D^+ for general cases.

Theorem 4. *$D^+(\boldsymbol{\mu})$ is $\boldsymbol{\mu}$-symmetric and $\mathring{D}^+(\boldsymbol{\mu}) = \frac{1}{c}H$ where*

- $c = c(\boldsymbol{\mu}) = (-1)^{mn + \frac{n(n-1)}{2} + \sum_{i=1}^m i\mu_i} \cdot (n-m)! \prod_{i=1}^m \mu_i^{\mu_i}$
- *$D = D(P)$ is the discriminant of $P(x) = \sum_{i=0}^n c_{n-i}x^i$ with multiplicity structure $\boldsymbol{\mu}$*
- $H := \frac{\partial^{n-m} D}{\partial c_n^{n-m}}\Big|_{c_i = (-1)^i z_i c_0} \Big/ c_0^{m+n-2}$.

The kernel formula tells us that if $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$ and $\boldsymbol{\mu}' = (\mu'_1, \ldots, \mu'_{m'})$ with $m = m'$, then there exists a constant a such that $\mathring{D}^+(\boldsymbol{\mu}) = a\mathring{D}^+(\boldsymbol{\mu}')$.

7 Experiments

Table 1 shows timings of our algorithms (`G-kern`, `E-kern` and `isMuSymmetric`) for checking the existence of the $\boldsymbol{\mu}$-kernel of $F \in K[\mathbf{r}]$, or reporting "No" otherwise. They are implemented in Maple (see code in Appendix A). These experiments use Maple 2017 on a Windows laptop with an Intel(R) Core(TM) i7-7660U CPU (2.50GHz, 8GB RAM). We use a test suite of 12 polynomials of degrees ranging from 6–20 (see Appendix B), with corresponding $\boldsymbol{\mu}$ with $n = \sum_{i=1}^m \mu_i$ ranging from 4–6. These polynomials are either D^+ polynomials or subdiscriminants, or their variants to create non-$\boldsymbol{\mu}$-symmetric polynomials.

Table 1. Timing for computing $\boldsymbol{\mu}$-kernel of F of degree δ. Here $n = \sum_{i=1}^m \mu_i$, `canonize` is a preprocessing step in `isMuSymmetric` and total= the sum of `canonize` time and `reduce` time.

F	δ	$\boldsymbol{\mu}$	n	Y/N	`G-kern` Time (sec)	`E-kern` Time (sec)	speedup (`G-kern`/ `E-kern`)	`isMuSymmetric`			speedup (`G-kern`/ `isMuSymmetric`)
								`canonize` (sec)	`reduce` (sec)	total (sec)	
F1	12	[1, 1, 1, 1]	4	Y	0.453	0.235	1.9	0.094	0.000	0.094	4.8
F2	8	[2, 1, 1]	4	Y	0.328	0.015	21.9	0.016	0.015	0.031	10.6
F3	20	[1, 1, 1, 1, 1]	5	Y	34.141	187.703	0.2	3.766	0.031	3.797	9.0
F4	15	[2, 1, 1, 1]	5	Y	>600.000	1.875	>320.0	0.391	0.015	0.406	>1477.8
F4x	6	[2, 1, 1, 1]	5	N	>600.000	0.015	>40000.0	0.000	0.016	0.016	>37500.0
F5	6	[2, 2, 1]	5	Y	68.031	0.032	2126.0	0.000	0.000	0.000	Inf
F5x	6	[2, 2, 1]	5	N	0.078	0.000	Inf	0.000	0.016	0.016	4.9
F6	10	[2, 2, 1]	5	Y	0.438	0.078	5.6	0.031	0.000	0.031	14.1
F6x	10	[2, 2, 1]	5	N	0.406	0.047	8.6	0.031	0.016	0.047	8.6
F7	18	[3, 1, 1, 1]	6	Y	>600.000	9.000	>66.7	3.390	0.063	3.453	>173.8
F8	12	[3, 2, 1]	6	Y	>600.000	0.360	>1666.7	0.187	0.000	0.187	>3208.6
F9	6	[2, 2, 2]	6	Y	8.734	0.000	Inf	0.000	0.000	0.000	Inf

Table 1 shows that `E-kern` is significantly faster than `G-kern` on all but in this case, $\boldsymbol{\mu} = (1, \ldots, 1)$, i.e., the ideal $\mathcal{I}_{\boldsymbol{\mu}} = \langle v_1, \ldots, v_n \rangle$ is symmetric in $\mathbf{r}$. Possibly, the Gröbner basis algorithm in Maple is highly optimized for such ideals.

One may also see that `isMuSymmetric` is also a very efficient method for checking the μ-symmetry of a polynomial. In particular, the preprocessing procedure `canonize` is independent on F, so one can compute the canonical set first and store it in a database. The actual time to reduce a given F using a canonical set is relatively small. The speedup of `G-kern`/`isMuSymmetric` may be partly attributed to the fact that `G-kern` outputs more information than `isMuSymmetric`. In the full paper, we will extend `isMuSymmetric` into an algorithm to actually compute the kernel.

8 Conclusion

We introduced the concept of $\boldsymbol{\mu}$-symmetric polynomials as a generalization of the classical symmetric polynomial, and designed efficient algorithms to compute $\boldsymbol{\mu}$-kernel of such polynomials. In particular, we proved the $\boldsymbol{\mu}$-symmetry of the root function $D^+(\boldsymbol{\mu})$; this function played a key role in the complexity analysis of the root clustering algorithm in [2]. We will continue to explore the application of the new result in designing efficient root clustering algorithms.

References

1. S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in Real Algebraic Geometry (Algorithms and Computation in Mathematics)*. Springer-Verlag, New York, 2nd edition, 2016.
2. R. Becker, M. Sagraloff, V. Sharma, J. Xu, and C. Yap. Complexity analysis of root clustering for a complex polynomial. In *41st Proc. ISSAC*, pages 71–78, 2016. July 19-22, Waterloo, Canada.
3. D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. Springer-Verlag, New York, 3rd edition, 2007.

About the Appendices. Appendix A lists our Maple code sources; Appendix B describes our benchmark polynomials; Appendix C provides all the proofs. These appendices are provided only for the referees' convenience. They will be removed in the proceedings version.

Appendix A. Maple Code for G-kern, E-kern and isMuSymmetric

```
restart:
with(Groebner):
with(combinat):

GenmElSym := proc(mu)
    # Input:  mu, multiplicity structure
    # Output: the n elementary symmetric polynomials in x[1],...,x[n]
    #         where n is the summation of mu[i]

    local n,m,i, Q, f;

    n := add(mu);
    m := nops(mu);
    Q := []:
    f := collect(product((x+r[i])^(mu[i]), i=1..m)-x^n, x):
    for i from 1 to n do
         Q := [op(Q), coeff(f, x, n-i)]:
    od:
    return expand(Q);
end:

Gkern := proc(F, mu)
    # Input:  F, a polynomial in r; mu, multiplicity structure
    # Output: the kernel of F or "No" when F is not mu-symmetric

    local m,n,e,P,B,R;

    m := nops(mu):
    n := add(mu):
    e := GenmElSym(mu):
    P := [seq(z[i]-e[i], i=1..n)]:
    B := Basis(P, plex(seq(r[m-i+1],i=1..m), seq(z[i], i=1..n))):
    R := NormalForm(expand(F), B,
                  plex(seq(r[m-i+1],i=1..m), seq(z[i], i=1..n)));

    if degree(R, {seq(r[i],i=1..m)})>0 then
         return "No";
    else
         return R;
    fi;
end:

Ekern := proc(F, mu)
    # Input:  F, a polynomial in r; mu, multiplicity structure
    # Output: the kernel of F or "No" when F is not mu-symmetric

    local n, m, delta, P, p, e, N, i,j, G,H, C, sol, ind;

    n     := add(mu);
    m     := nops(mu):
    delta := degree(F, {seq(r[i], i=1..m)});
    if delta>=n then
        P := partition(delta, n);
    else
        P := partition(delta);
    fi;

    e := GenmElSym(mu);
    N := 0;G := 0:
    for p in P do
        N := N+1;
        G := G + k[N]*product(z[p[i]], i=1..nops(p));
    od:

    H  := subs({seq(z[i]=e[i], i=1..n)}, G);
    C  := {coeffs(collect(H-F, {seq(r[i], i=1..m)},
                          'distributed'), [seq(r[i], i=1..m)])};
    sol := solve(C);
    if nops({sol})=0 then
        return "No";
    else
        ind := indets({seq(rhs(sol[i]), i=1..N)});
        G := subs({seq(ind[i]=0, i=1..nops(ind))}, subs(sol, G));
        return factor(G);
    fi;
end:

isMuSymmetric := proc(F, mu)
    # Input:  F, a polynomial in r; mu, multiplicity structure
    # Output: "Yes" when F is mu-symmetric, and "No" otherwise.

  local m, delta, n, S, P, B, p, R, i;
  m     := nops(mu):
  delta := degree(F, {seq(r[i], i=1..m)}):
  n     := add(mu);
  S     := GenmElSym(mu);
  if delta>=n then
    P := partition(delta,n);
  else
    P := partition(delta);
  fi;
  B := []:
  for p in P do
    B := [op(B), product(S[p[i]], i=1..nops(p))];
  od:
  B  := expand(B):
  B  := Canonize(B);
  R  := muReduce(F, B);

  if R=0 then
    return "Yes";
  else
    return "No";
  fi;
end:

muReduce := proc(F, B)
# Compute the reduced polynomial of F w.r.t. B

  local m, Bc, l, i, k, q, f, lf, lb;

  m  := nops(indets(B)):
  Bc := [1, op(B)];
  l  := nops(Bc);
  i  := l;
  q  := 0:
  f  := expand(F):

  if l=1 then return f; fi;

  while f<>0 do
    lf := LeadingTerm(f, plex(seq(r[m-k+1], k=1..m)));
    lb := LeadingTerm(Bc[i], plex(seq(r[m-k+1], k=1..m)));
    if lf[2]=lb[2] then
      f := expand(f-lf[1]/lb[1]*Bc[i]);
      i := i-1;
      if i=1 then
        q := q+f;
        f := 0;
      fi;
    elif LeadingTerm(lf[2]-lb[2],
                     plex(seq(r[m-k+1], k=1..m)))=lf[2] then
      q := q + lf[1]*lf[2]:
      f := f - lf[1]*lf[2]:
    else
      i := i-1;
      if i=1 then
        q := q+f;
        f := 0;
      fi;
    fi;
  od:

  return q;
end:

Canonize := proc(Fs)
# Generate the canonical set of Fs

  local m, B, P, b;

  m := nops(indets(Fs));
  B := [];
  P := stack[new](op(Fs)):
  while stack[depth](P)>0 do
    b := stack[pop](P);
    b := muReduce(b, B);
    if b<>0 then
      B := polyInsert(b, B):
    fi;
  od;

  return B;
end:

polyInsert := proc(b, B)

  local m,p,lB,l,i:

  m  := nops(indets([b, op(B)]));
  p  := LeadingMonomial(b, plex(seq(r[m-k+1], k=1..m)));
  lB := LeadingMonomial(B, plex(seq(r[m-k+1], k=1..m)));
  l  := nops(B):
  if l=0 then return [b]: fi;
  for i from 1 to l do
    if LeadingMonomial(p-lB[i],
                   plex(seq(r[m-k+1], k=1..m))) = lB[i] then
        return [op(1..i-1, B), b, op(i..l, B)];
    else
      if i=l then return [op(B), b]; fi;
    fi;
  od;

end:
```

Appendix B. Benchmark Polynomials

We have 9 polynomials Fi (for $i = 1, \ldots, 9$) that are (resp.) $\boldsymbol{\mu}_i$-symmetric. Also, $F4x, F5x, F6x$ are variants of Fi $(i = 4, 5, 6)$ which are not $\boldsymbol{\mu}_i$-symmetric.

```
# F1: Discriminant of polynomial of degree 4 = D+(1,1,1,1)
F1 :=  (r[1]-r[2])^2*(r[1]-r[3])^2*(r[3]-r[2])^2
      *(r[1]-r[4])^2*(r[2]-r[4])^2*(r[3]-r[4])^2:
mu_1:=[1,1,1,1]:

# F2: D+(2,1,1)
F2 := [(r[1]-r[2])^3*(r[1]-r[3])^3*(r[2]-r[3])^2:
mu_2:=[2,1,1]:

# F3: D+(1,1,1,1,1)
F3 :=  (r[1]-r[2])^2*(r[1]-r[3])^2*(r[1]-r[4])^2*(r[1]-r[5])^2
                    *(r[2]-r[3])^2*(r[2]-r[4])^2*(r[2]-r[5])^2
                    *(r[3]-r[4])^2*(r[3]-r[5])^2*(r[4]-r[5])^2:
mu_3 := [1,1,1,1,1]:

# F4: D+(2,1,1,1)
F4 :=   (r[1]-r[2])^3*(r[1]-r[3])^3*(r[1]-r[4])^3
       *(r[2]-r[3])^2*(r[2]-r[4])^2*(r[3]-r[4])^2:
mu_4 := [2,1,1,1]:

# F4x: square free part of F4
F4x :=  (r[1]-r[2])*(r[1]-r[3])*(r[1]-r[4])
       *(r[2]-r[3])*(r[2]-r[4])*(r[3]-r[4]):
mu_4x := [2,1,1,1]:

# F5: Subdiscriminant where n=5, k=2
F5 :=  (r[1]-r[2])^2*(r[1]-r[3])^2*(r[3]-r[2])^2:
mu_5 := [2,2,1]:
```

```
# F5x: perturbation of F5
F5x := (r[1]-r[2])^2*(r[1]-r[3])^2*(r[3]-r[2])^2 + r[1]^6:
mu_5x := [2,2,1]:

# F6: D+(2,2,1)
F6 :=  (r[1]-r[2])^4*(r[1]-r[3])^3*(r[3]-r[2])^3:
mu_6 := [2,2,1]:

# F6x: variant of F6 obtained by permutation of variables
F6x := (r[1]-r[3])^2*(r[1]-r[2])^2*(r[2]-r[3])^2 + r[1]^6:
mu_6x := [2,2,1]:

# F7: D+(3,1,1,1)
F7 :=  (r[1]-r[2])^4*(r[1]-r[3])^4*(r[1]-r[4])^4
      *(r[2]-r[3])^2*(r[2]-r[4])^2*(r[3]-r[4])^2:
mu_7 := [3,1,1,1]:

# F8: D+(3,2,1)
F8 :=  (r[1]-r[2])^5*(r[1]-r[3])^4*(r[2]-r[3])^3:
mu_8 := [3,2,1]:

# F9: Subdiscriminant where n=6, k=3
F9 :=  (r[1]-r[2])^2*(r[1]-r[3])^2*(r[3]-r[2])^2:
mu_9 := [2,2,2]:
```

Appendix C. All Proofs

- Lemma 1.
  Proof. If the polynomial F is $\boldsymbol{\mu}$-symmetric, there is a polynomial $\mathring{F} \in K[\mathbf{z}]$ such that $\mathring{F}(\overline{e}_1(\mathbf{r}), \ldots, \overline{e}_n(\mathbf{r})) = F(\mathbf{r})$. Evaluating of the above equation at $\mathbf{r} = (\rho_1, \ldots, \rho_m)$ leads to

  $$F(\rho_1, \ldots, \rho_m) = \mathring{F}\,(-c_1/c_0, \ldots, (-1)^n c_n/c_0) \in K$$

  by Viete's formula for roots. **Q.E.D.**

- Theorem 1.
  Proof. ($\Leftarrow$) Since R be the normal form of F w.r.t. $\mathcal{G}_{\boldsymbol{\mu}} = \{G_1, \ldots, G_t\}$, F can be written as

  $$F = \sum_{i=1}^{t} A_i G_i + R,$$

  where $A_i \in K[\mathbf{z}, \mathbf{r}]$.

First suppose that $R \in K[\mathbf{z}]$. Then for each i, substitute $\overline{e}_i$ for z_i in the above formula for F. Then the left side is unchanged and the right hand side becomes $R(\overline{e}_1, \ldots, \overline{e}_n)$. Hence, R is the kernel of F.

($\Rightarrow$) Suppose that $F \in K[\mathbf{r}]$ is $\boldsymbol{\mu}$-symmetric. Then there exist a polynomial $\mathring{F} \in K[\mathbf{z}]$ such that $\mathring{F}(\overline{e}_1, \ldots, \overline{e}_n) = F$. Since $\mathcal{G}_{\boldsymbol{\mu}}$ is the Gröbner basis of $\mathcal{I}_{\boldsymbol{\mu}}$, $\mathcal{G}_{\boldsymbol{\mu}} \cap K[\mathbf{z}]$ is the Gröbner basis of the elimination ideal $\mathcal{I}_{\boldsymbol{\mu}} \cap K[\mathbf{z}]$. Assume $\widetilde{R}$ is the normal form of $\mathring{F}$ w.r.t $\mathcal{G}_{\boldsymbol{\mu}} \cap K[\mathbf{z}]$. We want to show that $\widetilde{R}$ is the normal form of F w.r.t. $\mathcal{G}_{\boldsymbol{\mu}}$.

To prove this, first note that in $K[\mathbf{z}, \mathbf{r}]$, a monomial in $\overline{e}_1, \ldots, \overline{e}_n$ can be written as follows:

$$\begin{aligned} \overline{e}_1^{\alpha_1} \cdots \overline{e}_n^{\alpha_n} &= (z_1 - (z_1 - \overline{e}_1))^{\alpha_1} \cdots (z_n - (z_n - \overline{e}_n))^{\alpha_n} \\ &= z_1^{\alpha_1} \cdots z_n^{\alpha_n} + B_1(z_1 - \overline{e}_1) + \cdots + B_n(z_n - \overline{e}_n) \end{aligned}$$

for some $B_1, \ldots, B_n \in K[\mathbf{z}, \mathbf{r}]$. Multiplying by an appropriate constant and adding over the exponents appearing in $\mathring{F}$, it follows that

$$\mathring{F}(\overline{e}_1, \ldots, \overline{e}_n) = \mathring{F}(z_1, \ldots, z_n) + C_1(z_1 - \overline{e}_1) + \cdots + C_n(z_n - \overline{e}_n), \tag{5}$$

where $C_1, \ldots, C_n \in K[\mathbf{z}, \mathbf{r}]$. Meanwhile, the fact that $\widetilde{R}$ is the remainder of $\mathring{F}$ w.r.t. $\mathcal{G}_{\boldsymbol{\mu}} \cap K[\mathbf{z}]$ implies

$$\mathring{F}(z_1, \ldots, z_n) = C_1'(z_1 - \overline{e}_1) + \cdots + C_n'(z_n - \overline{e}_n) + \widetilde{R}(z_1, \ldots, z_n), \tag{6}$$

where $C_1', \ldots, C_n' \in K[\mathbf{z}, \mathbf{r}]$. Combining (5) and (6) and taking $F = \mathring{F}(\overline{e}_1, \ldots, \overline{e}_n)$ into account, we obtain

$$F = \mathring{F}(\overline{e}_1, \ldots, \overline{e}_n) = D_1(z_1 - \overline{e}_1) + \cdots + D_n(z_n - \overline{e}_n) + \widetilde{R}(z_1, \ldots, z_n),$$

where $D_1, \ldots, D_n \in K[\mathbf{z}, \mathbf{r}]$. Observe that $\widetilde{R} \in K[\mathbf{z}]$ and no terms of $\widetilde{R}$ is divisible by an element of $\mathcal{G}_{\boldsymbol{\mu}} \backslash K[\mathbf{z}]$. Moreover, no terms of $\widetilde{R}$ is divisible by an element of $\mathtt{Lt}(\mathcal{G}_{\boldsymbol{\mu}} \cap K[\mathbf{z}])$ because $\mathcal{G}_{\boldsymbol{\mu}} \cap K[\mathbf{z}]$ is the Gröbner basis of $\mathcal{I}_{\boldsymbol{\mu}} \cap K[\mathbf{z}]$. This proves that the normal form lies in $K[\mathbf{z}]$ when F is $\boldsymbol{\mu}$-symmetric.

The second part of the theorem follows immediately from the above arguments.

Q.E.D.

- Lemma 3.
  Proof. In each loop, either i is decreased by 1 (in which case, we used B_i for reduction) or i is unchanged (in which case, a term in $\mathtt{Supp}(F) \backslash \mathtt{Lt}(\mathcal{B})$ is removed from the support of F). Thus the number of loops N_{loop} will be no greater than the sum of ℓ and the number of possible terms in $(\mathtt{Supp}(F) \cup \mathtt{Supp}(\mathcal{B})) \backslash \mathtt{Lt}(\mathcal{B})$. Noting that $\mathtt{Lt}(\mathcal{B}) \subseteq \mathtt{Supp}(F) \cup \mathtt{Supp}(\mathcal{B})$ and $\mathtt{Lt}(B_i) \neq \mathtt{Lt}(B_j)$ when $i \neq j$, we have

$$\#\left((\mathtt{Supp}(F) \cup \mathtt{Supp}(\mathcal{B})) \backslash \mathtt{Lt}(\mathcal{B})\right) = \#(\mathtt{Supp}(F) \cup \mathtt{Supp}(\mathcal{B})) - \ell.$$

Therefore,

$$\begin{aligned}\#\left((\texttt{Supp}(F)\cup\texttt{Supp}(\mathcal{B}))\backslash\texttt{Lt}(\mathcal{B})\right) =&\#(\texttt{Supp}(F)\cup\texttt{Supp}(\mathcal{B}))-\ell\\ \le&\#\texttt{Supp}(F)+\sum_{i=1}^{\ell}\#\texttt{Supp}(B_i)-\ell.\end{aligned}$$

Case 1: "=" holds. In this case, $\texttt{Supp}(F),\texttt{Supp}(B_1),\ldots,\texttt{Supp}(B_\ell)$ are disjoint sets. Then we immediately get $\texttt{Supp}(F)\backslash\texttt{Lt}(\mathcal{B})=\texttt{Supp}(F)$ and thus the number of loops

$$N_{loop}\le\#\texttt{Supp}(F)+\#\mathcal{B}-1\le\#\texttt{Supp}(F)+\sum_{i=1}^{\ell}\#\texttt{Supp}(B_i)-1.$$

Case 2: "=" does not hold. Then

$$\#\left((\texttt{Supp}(F)\cup\texttt{Supp}(\mathcal{B}))\backslash\texttt{Lt}(\mathcal{B})\right)\le\#\texttt{Supp}(F)+\sum_{i=1}^{\ell}\#\texttt{Supp}(B_i)-\ell-1.$$

It follows that

$$\begin{aligned}N_{loop}\le&\ell+\#\left((\texttt{Supp}(F)\cup\texttt{Supp}(\mathcal{B}))\backslash\texttt{Lt}(\mathcal{B})\right)-1\\ \le&\#\texttt{Supp}(F)+\sum_{i=1}^{\ell}\#\texttt{Supp}(B_i)-1.\end{aligned}$$

To prove the bound is tight, consider $F=p_1+q_1+\cdots+q_s$ and $\mathcal{B}=(p_1,\ldots,p_\ell)$ with the term ordering $p_1\prec\cdots\prec p_\ell\prec q_1\prec\cdots q_s$. In the first s loops, since $\texttt{Lt}(F)\succ p_\ell$, i is unchanged and $q_1,\ldots,q_s$ are removed from F. In the next $\ell-1$ loops, since $\texttt{Lt}(F)=p_1\prec p_2\prec\cdots\prec p_\ell$, F is unchanged and i will drop to 1. In the last loop, since $\texttt{Lt}(F)=p_1=\texttt{Lt}(B_1)$, F will be reduced relative to B_1 to 0. So the total number of loops is $s+\ell=\#\texttt{Supp}(F)+\sum_{i=1}^{\ell}\#\texttt{Supp}(B_i)-1$.

Q.E.D.

- Lemma 4
  Proof. First, the termination of `isMuSymmetric` follows from those of `canonize` and `reduce`. We only need to show partial correctness: when the algorithm halts, then it outputs 'Yes' iff F is $\boldsymbol{\mu}$-symmetric. Assume $\deg(F,\mathbf{r})=\delta$. Recall that $F\in K[\mathbf{r}]$ is $\boldsymbol{\mu}$-symmetric iff there exists a homogeneous symmetric polynomial $\widehat{F}\in K[\mathbf{x}]$ of degree δ such that $\sigma_{\boldsymbol{\mu}}(\widehat{F})=F(\mathbf{r})$. By Proposition 2, $\widehat{F}$ is symmetric and with degree δ iff $\widehat{F}\in K^{\delta}_{\texttt{sym}}[\mathbf{x}]$. Thus $F=\sigma_{\boldsymbol{\mu}}(\widehat{F})\in K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$ where $K^{\delta}_{\boldsymbol{\mu}}[\mathbf{r}]$ is a K-vector space with the basis generated by $\mathcal{C}=\{\overline{e}_\alpha:\alpha\vdash(\delta,n)\}$. If $\mathcal{B}=\mathit{canonize}(\mathcal{C})$, then $\mathcal{B}$ is the basis we want to obtain. Therefore, if F is $\boldsymbol{\mu}$-symmetric iff $\texttt{reduce}(F,\mathcal{B})=0$.

Q.E.D.

- Lemma 5
  Proof. To prove D^+ is injective for any $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$ where $m > 2$, we assume that $m > 2$ for any $\boldsymbol{\mu}$. Then we need to show that for any $\boldsymbol{\mu}' = (\mu'_1, \ldots, \mu'_{m'})$, if $D^+(\boldsymbol{\mu}) = D^+(\boldsymbol{\mu}')$ then $\boldsymbol{\mu} = \boldsymbol{\mu}'$. Suppose $D^+(\boldsymbol{\mu}) = D^+(\boldsymbol{\mu}')$. We now show that $\boldsymbol{\mu} = \boldsymbol{\mu}'$. Clearly, m' must be equal to m. Our supposition implies that $\mu_i + \mu_j = \mu'_i + \mu'_j$ $(1 \le i < j \le m)$. Consider the following linear system:

$$\{\mu_2 + \mu_3 = \mu'_2 + \mu'_3\} \cup \{\mu_1 + \mu_i = \mu'_1 + \mu'_i : i = 2, \ldots, m\}.$$

  Solving the last $n - 1$ equations for μ_i, we get $\mu_i = \mu'_1 + \mu'_i - \mu_1$ for all $2 \le i \le m$. Substituting for μ_i into the first equation yields

$$\mu_2 + \mu_3 = (\mu'_1 + \mu'_2 - \mu_1) + (\mu'_1 + \mu'_3 - \mu_1) = \mu'_2 + \mu'_3,$$

  which implies $\mu_1 = \mu'_1$. It follows immediately that $\mu_i = \mu'_i$ for $i = 2, \ldots, m$. **Q.E.D.**

- Lemma 6.
  Proof. Let $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)$. Consider the m-th subdiscriminant S^n_m in n variables. We may verify that

$$\sigma_{\boldsymbol{\mu}}(S^n_{n-m}) = \mathit{\Delta} \cdot \prod_{i=1}^{m} \mu_i.$$

  This is equivalent to

$$\sigma_{\boldsymbol{\mu}}\left(\frac{1}{\prod_{i=1}^{m} \mu_i} \cdot S^n_{n-m}\right) = \mathit{\Delta}.$$

  Therefore $\frac{1}{\prod_{i=1}^{m} \mu_i} S^n_{n-m}$ is the the $\boldsymbol{\mu}$-kernel of $\mathit{\Delta}$.
  To obtain the explicit formula in case $m = 2$, consider the symmetric polynomial $Q := \sum_{i<j}(x_i - x_j)^2$. It is easy to check that $Q = (n-1)e_1^2 - 2ne_2$. A simple calculation shows that

$$\sigma_{\boldsymbol{\mu}}(Q) = \mu_1\mu_2(r_1 - r_2)^2.$$

  Thus, we may choose $\widehat{\mathit{\Delta}} = \frac{(n-1)e_1^2 - 2ne_2}{\mu_1\mu_2}$. **Q.E.D.**

- Theorem 2.
  Proof. Since $\mu_i = a$ $(1 \le i \le m)$,

$$D^+(\boldsymbol{\mu}) = \prod_{i<j}(r_i - r_j)^{2a} = \left(\prod_{i<j}(r_i - r_j)^2\right)^a.$$

  This expression for D^+ is $\boldsymbol{\mu}$-symmetric since $\prod_{i<j}(r_i - r_j)^2$ is $\boldsymbol{\mu}$-symmetric by Lemma 6(a). Moreover, Lemma 6(a) also shows that the lift of $\prod_{i<j}(r_i - r_j)^2$ is $\frac{1}{a^m} \cdot S^n_{n-m}(\mathbf{x})$. Thus we may choose $\widehat{F}_n = \left(\frac{1}{a^m} \cdot S^n_{n-m}\right)^a$. **Q.E.D.**

- Theorem 3.
  Proof. From Lemma 6(b), we know that $(r_1 - r_2)^2$ is $\boldsymbol{\mu}$-symmetric for arbitrary n and $(r_1 - r_2)^2 = \frac{(n-1)\overline{e}_1^2 - 2n\overline{e}_2}{\mu_1\mu_2}$.
  When n is even,

$$\begin{aligned} D^+(\boldsymbol{\mu}) = \left((r_1 - r_2)^2\right)^{\frac{n}{2}} &= \left(\frac{(n-1)\overline{e}_1^2 - 2n\overline{e}_2}{\mu_1\mu_2}\right)^{\frac{n}{2}} \\ &= \left(\frac{(n-1)\overline{e}_1^2 - 2n\overline{e}_2}{\mu_1\mu_2}\right)^{\frac{n}{2}} = \mathring{F}_n(\overline{e}_1, \overline{e}_2). \end{aligned}$$

Thus the case for even n is proved. It remains to prove the case for odd n. First, it may be verified that

$$(r_1 - r_2)^3 = k_1\overline{e}_1^3 + k_2\overline{e}_1\overline{e}_2 + k_3\overline{e}_3,$$

where

$$k_1 = \frac{-(n-1)(n-2)}{d}, \quad k_2 = \frac{3n(n-2)}{d}, \quad k_3 = \frac{-3n^2}{d} \text{ and } d = \mu_1\mu_2(\mu_1 - \mu_2).$$

It follows that

$$\begin{aligned} D^+(\boldsymbol{\mu}) &= \left((r_1 - r_2)^2\right)^{\frac{n-3}{2}} (r_1 - r_2)^3 \\ &= \left(\frac{(n-1)\overline{e}_1^2 - 2n\overline{e}_2}{\mu_1\mu_2}\right)^{\frac{n-3}{2}} \left(k_1\overline{e}_1^3 + k_2\overline{e}_1\overline{e}_2 + k_3\overline{e}_3\right) \\ &= \left(\frac{(n-1)\overline{e}_1^2 - 2n\overline{e}_2}{\mu_1\mu_2}^{\frac{n}{2}}\right) \left(k_1\overline{e}_1^3 + k_2\overline{e}_1\overline{e}_2 + k_3\overline{e}_3\right) \\ &= \mathring{F}_n(\overline{e}_1, \overline{e}_2, \overline{e}_3) \end{aligned}$$

where

$$k_1 = \frac{-(n-1)(n-2)}{d}, \quad k_2 = \frac{3n(n-2)}{d}, \quad k_3 = \frac{-3n^2}{d} \text{ and } d = \mu_1\mu_2(\mu_1 - \mu_2).$$

Q.E.D.

- Theorem 4.
  Proof. Let $P(x) = \sum_{i=0}^{n} c_{n-i}x^i$ and $G := \frac{\partial^{n-m} D}{\partial c_n^{n-m}} \in K[c_0, \ldots, c_n]$. First we show that

$$H = G(c_0, -z_1c_0, \ldots, (-1)^n z_n c_0)/c_0^{m+n-2} \in K[\mathbf{z}]. \tag{7}$$

By discriminant theory, D is homogeneous of degree $2n-2$ in $c_0, \ldots, c_n$. Note that a term in D either becomes zero or has degree $2n-2-(n-m)$ after taking $(n-m)$-th derivative for c_n. Thus G is homogeneous of degree $m+n-2$ in $c_0, \ldots, c_n$. It follows that any term in $\mathtt{Supp}(G)$ will become the product of c_0^{m+n-2} and a monomial in $\mathbf{z}$ after the substitution of $c_i = (-1)^i z_i c_0$ $(1 \leq i \leq n)$. Therefore,

$$H = G(c_0, -z_1c_0, \ldots, (-1)^i z_i c_0, \ldots, (-1)^n z_n c_0)/c_0^{m+n-2} \in K[\mathbf{z}]. \tag{8}$$

In the remaining part, we will show that $H(\overline{e}_1, \ldots, \overline{e}_n) = cD^+$, i.e.,

$$G(c_0, -\overline{e}_1c_0, \ldots, (-1)^i\overline{e}_ic_0, \ldots, (-1)^n\overline{e}_nc_0)/c_0^{m+n-2} = cD^+. \tag{9}$$

Since $\deg(P', x) = n-1$, P' has $n-1$ roots in the closure of K, say $\beta_1, \ldots, \beta_{n-1}$. Then the classical discriminant $D(P)$ of P is

$$D(P) = \frac{(-1)^{\frac{n(n-1)}{2}}}{c_0}\mathtt{res}(P, P', x) \in K[c_0, \ldots, c_n]. \tag{10}$$

Recall

$$\mathtt{res}(P, P', x) = (nc_0)^n \prod_{i=1}^{n-1} P(\beta_i) \tag{11}$$

where β_i's are the roots of $P'(x)$. Then

$$D(P) = (-1)^{\frac{n(n-1)}{2}} n^n c_0^{n-1} \prod_{i=1}^{n-1} P(\beta_i). \tag{12}$$

Note that c_n does not appear in P'. Thus β_i is independent with c_n. On the other hand, $\deg(P, c_n) = 1$ and $\mathtt{Coef}(P, c_n) = 1$, which imply $\frac{dP(\beta_i)}{dc_n} = 1$. Therefore,

$$G = (-1)^{\frac{n(n-1)}{2}} c_0^{n-1} \cdot n^n \cdot (n-m)! \sum_{\alpha \in \binom{[n-1]}{m-1}} \prod_{i \in \alpha} P(\beta_i). \tag{13}$$

In that follows, we will evaluate $P(\beta_i)$ for $i = 1, \ldots, m-1$ with the assumption that P has the multiplicity structure $\boldsymbol{\mu}$.

With such assumption, evaluating P at $c_i = (-1)^i c_0 \overline{e}_i$ leads to $P = c_0 \prod_{i=1}^m (x - r_i)^{\mu_i}$. It is easy to see that P' has $n-m$ known roots, i.e., $\underbrace{r_1, \ldots, r_1}_{\mu_1 - 1}, \ldots, \underbrace{r_m, \ldots, r_m}_{\mu_m - 1}$.

Let $\beta_m, \ldots, \beta_{n-1}$ be these $n-m$ roots. Note that the terms in (13) are products of $m-1$ $P(\beta_i)$'s and $P(\beta_i) = 0$ if $i \geq m$. It follows that only one term does not get vanished in (13), which is $\prod_{i=1}^{m-1} P(\beta_i)$. Thus

$$\begin{aligned}
H(\overline{e}_1,\ldots,\overline{e}_n) =&\frac{1}{c_0^{m+n-2}} G(c_0, -\overline{e}_1 c_0, \ldots, (-1)^n \overline{e}_n c_0)\\
=&\frac{(-1)^{\frac{n(n-1)}{2}} c_0^{n-1} n^n (n-m)! \prod_{i=1}^{m-1} P(\beta_i)\Big|_{\substack{c_i=(-1)^i c_0 \overline{e}_i\\ 1\le i\le n}}}{c_0^{m+n-2}}\\
=&\frac{(-1)^{\frac{n(n-1)}{2}} c_0^{n-1} n^n (n-m)! \cdot c_0^{m-1} \prod_{i=1}^{m-1} \prod_{j=1}^{m} (\beta_i - r_j)^{\mu_j}}{c_0^{m+n-2}}\\
=&(-1)^{n(m-1)+\frac{n(n-1)}{2}} n^n (n-m)! \prod_{j=1}^{m} \prod_{i=1}^{m-1} (r_j-\beta_i)^{\mu_j}\\
=&(-1)^{n(m-1)+\frac{n(n-1)}{2}} (n-m)! \prod_{j=1}^{m} [Q(r_j)]^{\mu_j} \qquad (14)
\end{aligned}$$

where

$$Q(x) = \frac{P'}{nc_0 \prod_{i=1}^{m} (x-r_i)^{\mu_i - 1}} = \frac{1}{n} \sum_{i=1}^{m} \Big[\mu_i \prod_{\substack{1\le k\le m\\ k\ne i}} (x-r_k)\Big]. \qquad (15)$$

Evaluating $Q(x)$ at $x = r_j$ yields $Q(r_j) = \mu_j \prod_{\substack{1\le k\le m\\ k\ne j}} (r_j - r_k)$. After substituting $Q(r_j)$ into (14), we get

$$\begin{aligned}
H(\overline{e}_1,\ldots,\overline{e}_n) =&(-1)^{n(m-1)+\frac{n(n-1)}{2}} (n-m)! \prod_{j=1}^{m} [H(r_j)]^{\mu_j}\\
=&(-1)^{n(m-1)+\frac{n(n-1)}{2}} (n-m)! \prod_{j=1}^{m} \Big[\mu_j \prod_{\substack{1\le k\le m\\ k\ne j}} (r_j-r_k)\Big]^{\mu_j}\\
=&(-1)^{n(m-1)+\frac{n(n-1)}{2}} (n-m)! \prod_{j=1}^{m} \mu_j^{\mu_j} \cdot \prod_{j=1}^{m} \Big[\prod_{\substack{1\le k\le m\\ k\ne j}} (r_j-r_k)\Big]^{\mu_j}\\
=&(-1)^{n(m-1)+\frac{n(n-1)}{2}} (n-m)! \prod_{j=1}^{m} \mu_j^{\mu_j} \cdot\\
&\qquad (-1)^{\sum_{i=1}^{m} (i-1)\mu_i} \prod_{1\le j<k\le m} (r_j - r_k)^{\mu_j+\mu_k}\\
=&(-1)^{n(m-2)+\frac{n(n-1)}{2}+\sum_{i=1}^{m} i\mu_i} (n-m)! \prod_{j=1}^{m} \mu_j^{\mu_j} D^+ = cD^+
\end{aligned}$$

where $c = (-1)^{mn+\frac{n(n-1)}{2}+\sum_{i=1}^{m} i\mu_i} (n-m)! \prod_{j=1}^{m} \mu_j^{\mu_j}$. **Q.E.D.**