Primary decomposition of zero-dimensional ideals over arbitrary fields

Sebastian Jambor

We present two algorithms to compute the primary decomposition of zero-dimensional ideals. The algorithms are deterministic and mainly use linear algebra techniques, based on the FGLM approach. This allows us for the first time to give a complete complexity analysis of a primary decomposition algorithm. We show that the primary decomposition can be computed in $O(nd^4)$ field operations and d factorizations of univariate polynomials over the ground field, where n is the number of generators of the polynomial ring and d is the residue class dimension of the ideal. This result and the corresponding algorithm are valid for all fields. A publicly available computer implementation shows that the algorithms also perform very well in practice.

1 Introduction

An algorithm to compute a primary decomposition of a polynomial ideal usually has two parts. In the first, the problem is reduced to a zero-dimensional problem, possibly by enlarging the ground field; this zero-dimensional problem is solved in the second part. Algorithms for the first part are given in [GTZ88] and [EHV92]. The second part seems to have received much more attention. Several algorithms exist, but they usually have some restriction on the ground field K.

Gianni, Trager and Zacharias [GTZ88] propose two algorithms. The first does not have restrictions on K per se, but requires the factorization of polynomials over finite extensions of K; as outlined by Steel [Ste05], this may be problematic if K is infinite of positive characteristic. The second algorithm is restricted to fields of characteristic zero. A similar algorithm is given in [Kre89] and refined in [BW93]; it also requires characteristic zero, but whereas the algorithm in [GTZ88] is probabilistic, this algorithm is deterministic. Another approach when K has characteristic zero was suggested in [Mon02]. The case where K is infinite of positive characteristic is solved in [Ste05]. The authors of [EHV92] address the zero-dimensional problem with a probabilistic approach which requires the field K to be perfect. An algorithm based on Berlekamp factorization addresses the case of finite fields [GWW09].

All of these approaches have in common that they use Gröbner basis computations at some point. Unfortunately, this computation can be very slow and often becomes the bottle neck of the algorithm. This also is true for other commutative algebra algorithms. The approach of [FGLM93] is to replace the Gröbner basis algorithm by linear algebra algorithms; in that case, it was used for an algorithm which changes the monomial order of a Gröbner basis. We will follow their approach and consider primary decomposition from the viewpoint of finite dimensional commutative algebras. We will unify the approaches of [GTZ88, Kre89, BW93, Ste05] to get an algorithm for primary decomposition, provided that the ground field is infinite. The methods are deterministic and only use linear algebra and polynomial factorization. This allows us to provide a complete complexity analysis of the algorithm. Based on these ideas and the approaches of [EHV92, Mon02] we then present a second algorithm which is independent of the ground field, provided that there is an algorithm which can factor univariate polynomials over that field.

The idea to get from ideals to commutative algebras is as follows. Let $K[\underline{x}] := K[x_1, \ldots, x_n]$ be the polynomial ring in n indeterminates over a field K. Every zero-dimensional ideal $I \trianglelefteq K[\underline{x}]$ defines a finite dimensional commutative K-algebra A by setting $A := K[\underline{x}]/I$, and A is generated by the residue classes of the x_i. Conversely, every finite dimensional commutative K-algebra A generated by $a_1, \ldots, a_n \in A$ defines a zero-dimensional ideal $I \trianglelefteq K[\underline{x}]$, namely the kernel of the epimorphism of algebras $K[\underline{x}] \to A$ which sends x_i to a_i. In fact, this gives a bijection between zero-dimensional ideals of $K[\underline{x}]$ and isomorphism classes of finite-dimensional commutative K-algebras with distinguished ordered generating sets of size n. We can use this bijection to translate problems from commutative algebra to problems of linear algebra or basic abstract algebra, thereby replacing Buchberger's or Faugère's algorithm by the Gaussian algorithm or other efficient algorithms.

The idea to use FGLM techniques for tasks other than order change is not new. For example, it has been extended to arbitrary finite-dimensional commutative algebras [MMM93]; this has been extended to the non-commutative case [BTBQM00]. In [AKR05] it is used to compute the intersection of two zero-dimensional ideals, and in [Lak90] to develop a Gröbner basis algorithm for zero-dimensional ideals which has a single-exponential time complexity.

The outline of the paper is as follows. In Section 2 we fix some notations and recall the FGLM idea. Section 3 lists several algorithms which are used in later sections. Primary decomposition is often tightly coupled with an algorithm to compute the radical. In Section 4 we translate the algorithms of [KL91, Kem02] for radical computations to the language of algebras. Sections 5 and 6 form the main part of the paper; the former presents an algorithm for infinite fields, and the latter one for arbitrary fields. Several benchmarks are given in Section 7.

2 Preliminaries

We assume throughout the paper that $2 < \omega \leq 3$ is a feasible matrix multiplication exponent, that is, two $n \times n$ matrices can be multiplied in $O(n^{\omega})$ field operations. Using classical matrix multiplication, $\omega = 3$; Strassen's algorithm, which is implemented in Magma, has $\omega = \log_2(7) \approx 2.8074$ [Str69]; the best known value is $\omega = 2.3727$ for William's variant of the Coppersmith-Winograd algorithm [Wil12]. Note that the complexity to compute the product of two rectangular matrices can also be given in terms of ω. Let $n_1, n_2, n_3 \in \mathbb{N}$, and let $\{i, j, k\} = \{1, 2, 3\}$ such that $n_i \leq n_j \leq n_k$. A simple argument, see for example [Kni95], shows that the product of an $n_1 \times n_2$ matrix by an $n_2 \times n_3$ matrix can be computed in $O(n_i^{\omega-2} n_j n_k)$ field operations.

All theoretical results in this paper are valid for arbitrary fields, unless stated otherwise. This is also true for most algorithms, with the following exceptions. As shown in [Kem02, Remark 8(a)] and [Ste05, Section 5.2], we may always assume that an infinite field of positive characteristic has the form $\mathbb{F}_q(t_1, \ldots, t_m)$ for a prime power q and indeterminates $t_1, \ldots, t_m$. This assumption is used for the algorithms in Sections 4 and 5. Furthermore, the algorithms in Section 5 are only valid for infinite fields.

We further assume that a monomial order on $K[\underline{x}]$ is fixed. Denote by $\operatorname{Mon}(K[\underline{x}])$ the set of monomials of $K[\underline{x}]$. For $f \in K[\underline{x}]$ let $\operatorname{lm}(f)$ be the leading monomial of f, and for $M \subseteq K[\underline{x}]$ let $\operatorname{LM}(M) := \{\operatorname{lm}(f) \mid f \in M\}$ and $\operatorname{NM}(M) := \{m \in M \mid n \nmid m \text{ for all } m \neq n \in M\}$.

Let $f_1, \ldots, f_k$ be elements of a K-algebra A; we denote by $\langle f_1, \ldots, f_k \rangle \trianglelefteq A$ the ideal of A generated by $f_1, \ldots, f_k$, and by $\langle f_1, \ldots, f_k \rangle_K \leq A$ the K-span of $f_1, \ldots, f_k$.

The following result underpins the translation between Gröbner bases and finite-dimensional commutative K-algebras; it is essentially contained in the proof of [FGLM93, Proposition 3.1].

Proposition 2.1. *Let $I \trianglelefteq K[\underline{x}]$ be an ideal and $A := K[\underline{x}]/I$ the residue class algebra; denote by $\nu\colon K[\underline{x}] \to A$ the canonical epimorphism. Let $B := \operatorname{Mon}(K[\underline{x}]) \setminus \operatorname{LM}(I)$ and $L := \operatorname{NM}(\operatorname{LM}(I))$. Since $\nu(B) = \{\nu(b) \,|\, b \in B\}$ is a K-basis of A, for every $f \in K[\underline{x}]$ there exist unique $\lambda_{b,f} \in K$ with $\nu(f) = \sum_{b \in B} \lambda_{b,f} \nu(b)$.*

The set $G := \{M - \sum_{b \in B} \lambda_{b,M} b \,|\, M \in L\}$ is a reduced Gröbner basis of I. Furthermore, $B = \{b \in \operatorname{Mon}(K[\underline{x}]) \mid \nu(b) \notin \langle \nu(n) \mid n \in \operatorname{Mon}(K[\underline{x}]) \text{ with } n < b\rangle_K\}$.

Proof. Clearly, $G \subseteq \ker(\nu) = I$, and $\operatorname{LM}(\langle G \rangle) = \operatorname{LM}(I)$, which proves the first statement. For the second, note that $m \in \operatorname{LM}(I)$ if and only if $m - \sum_{n<m} \mu_n n \in I$ for some $\mu_n \in K$, which is equivalent to $\nu(m) \in \langle \nu(n) \,|\, n < m \rangle_K$. □

Note that the proposition is valid for arbitrary ideals. However, in the following we will only apply it to zero-dimensional ideals.

Definition 2.2. Let $0 \neq e \in K^{1 \times d}$, and let $M := (M_1, \ldots, M_n) \in (K^{d \times d})^n$ be commuting matrices. Then $A := \langle e \cdot m(M_1, \ldots, M_n) \mid m \in \operatorname{Mon}(K[\underline{x}]) \rangle_K$ is a finite-dimensional commutative K-algebra, generated by $a_i := eM_i$. We denote this algebra by $\mathrm{Alg}(M, e)$. If $e = e_1$ is the first standard basis vector, we also write $\mathrm{Alg}(M)$ instead of $\mathrm{Alg}(M, e_1)$, and call M the *FGLM data* of A.

Note that the elements of A are just vectors in $K^{1\times d}$. Every finite-dimensional commutative K-algebra can be described in this way. A special case of this representation is used in [FGLM93].

Remark 2.3. Let $I \trianglelefteq K[\underline{x}]$ be a zero-dimensional ideal; let $A := K[\underline{x}]/I$ and $\nu\colon K[\underline{x}] \to A$ the canonical epimorphism. Set $\mathrm{B}(I) := \mathrm{B}(A) := \mathrm{Mon}(K[\underline{x}]) \setminus \mathrm{LM}(I)$, the *monomial basis* of A. We assume that $\mathrm{B}(I)$ is sorted increasingly. For $1 \leq i \leq n$ let $M_i \in K^{d\times d}$ be the matrix of the linear map $A \to A\colon a \mapsto a \cdot \nu(x_i)$ with respect to the basis $\nu(B)$. Then $A \cong \mathrm{Alg}((M_1, \dots, M_d))$. We call $M = (M_1, \dots, M_n)$ the *FGLM data* of I.

Note that we can use [FGLM93, Procedure 3.1] to calculate efficiently the FGLM data in $O(nd^3)$ field operations, where $d = \dim_K(K[\underline{x}]/I)$.

3 Auxiliary algorithms

Let $A = \mathrm{Alg}(M, e)$. A natural generalization of the FGLM algorithm yields an algorithm to compute the Gröbner basis of $\ker \nu$ in $O(nd^3)$ field operations, where $\nu\colon K[\underline{x}] \to A\colon x_i \mapsto a_i$ is the natural epimorphism, see for example [MMM93]. In our applications, the algebras in questions are quotients of $K[\underline{x}]/I$, where a Gröbner basis for I is known. For this case, we can give an algorithm with a better runtime complexity.

Algorithm 3.1 (IdealBasisToGroebner).
Input: A Gröbner basis G of a zero-dimensional ideal $I \trianglelefteq K[\underline{x}]$ and a K-basis C of an ideal $J \trianglelefteq K[\underline{x}]/I$.
Output: A Gröbner basis for $I' := \ker(\nu')$, where $\nu'\colon K[\underline{x}] \to (K[\underline{x}]/I)/J\colon x_i \mapsto (x_i + I) + J$.

1. Let $B := \mathrm{B}(I) = \{B_1, \dots, B_d\}$ be sorted decreasingly, and $C = (C_1, \dots, C_k)$. Denote by $\nu\colon K[\underline{x}] \to K[\underline{x}]/I$ the canonical epimorphism. Let $\Gamma = (\Gamma_{ij}) \in K^{k\times d}$ be the matrix of the coefficients of C in terms of $\nu(B)$, that is, $C_i = \sum_{j=1}^{d} \Gamma_{ij}\nu(B_j)$. Compute the echelon form $E = (E_{ij}) \in K^{k\times d}$ of Γ.

2. Set $L' := \emptyset$ and $G' := \emptyset$.

   For $i = k, \dots, 1$: Let j be the index of the first non-zero entry of the ith row of E. If B_j is not a multiple of a monomial in L', then add B_j to L' and $\sum_{\ell=1}^{d} E_{i\ell}B_\ell$ to G'.

3. Let $H := \{h \in G \mid m \nmid \mathrm{lm}(h) \text{ for all } m \in L'\}$; let $\rho(h - \mathrm{lm}(h)) \in K^{1\times d}$ be the representation of $h - \mathrm{lm}(h)$ with respect to the basis B, and let $R \in K^{|H|\times d}$ be the matrix with rows $\rho(h - \mathrm{lm}(h))$. Let $P \subseteq \{1, \dots, n\}$ be the set of pivot column indices of E, and let $R' \in K^{|H|\times k}$ be the matrix of the P-columns of R. Compute $S = R - R'E \in K^{|H|\times k}$ and set $H' := \{\mathrm{lm}(h) + \sum_{j=1}^{d} S_{h,j}B_j \mid h \in H\}$.

4. Return $G' \cup H'$.

Remark 3.2. Note that in this algorithm we assume that the basis $\mathrm{B}(I)$ is sorted *decreasingly.* In practice, this can be achieved by simply reversing the order of the columns of all matrices and vectors.

Proposition 3.3. *Algorithm* 3.1 *is correct. It requires $O(nk^{\omega-2}d^2)$ field operations, where $k = \dim_K(J)$ and $d = \dim_K(K[\underline{x}]/I)$.*

Proof. Let $B' = \{B_j \,|\, (i, j) \text{ is not a pivot entry of } E \text{ for every } 1 \leq i \leq k\}$. We first show $\mathrm{LM}(I') = \mathrm{LM}(I) \cup (B \setminus B')$ and $\mathrm{B}(I') = B'$. Since $I \subseteq I'$ we see $\mathrm{LM}(I) \subseteq \mathrm{LM}(I')$. Now let $B_j \in B \setminus B'$, so (i, j) is a pivot entry of E for some $1 \leq i \leq k$. Then $\sum_{\ell=1}^{d} E_{i\ell}\nu(B_\ell) \in \langle C_1, \dots, C_k\rangle_K = J$, so $\sum_{\ell=1}^{d} E_{i\ell}\nu'(B_\ell) = 0$. By definition of I' this implies $\sum_{\ell=1}^{d} E_{i\ell}B_\ell = B_j + \sum_{\ell=j+1}^{d} E_{i\ell}B_\ell \in I'$. Since the elements of B are sorted decreasingly, B_j is the leading monomial, so $B_j \in \mathrm{LM}(I')$. This proves $\mathrm{LM}(I) \cup (B \setminus B') \subseteq \mathrm{LM}(I')$ and hence $\mathrm{B}(I') = \mathrm{Mon}(K[\underline{x}]) \setminus \mathrm{LM}(I') \subseteq \mathrm{Mon}(K[\underline{x}]) \setminus (\mathrm{LM}(I) \cup (B \setminus B')) = B'$. By basic linear algebra, $\nu'(B')$ is a basis of $(K[\underline{x}]/I)/J \cong K[\underline{x}]/I'$, so equality holds in both cases by Proposition 2.1.

It is easy to verify that $L' \cup \mathrm{LM}(H) = \mathrm{NM}(\mathrm{LM}(I) \cup (B \setminus B')) = \mathrm{NM}(\mathrm{LM}(I'))$. Since E is in echelon form, the elements of G' are of the form $M - \sum_{b\in B'} \lambda_{b,M} b$ with $M \in L'$. The computation in Step 3 is equivalent to reducing the rows of R modulo the row span of E, so the elements in H' are the elements of H reduced modulo J. In particular, every element of H' is of the form $M - \sum_{b \in B'} \lambda_{b,M} b$. The correctness now follows by Proposition 2.1.

The computation of the echelon form in Step 1 requires $O(k^{\omega-1}d)$ field operations by [Sto94, Theorem 2.10]; Step 2 requires $O(kd)$ field operations. Step 3 is the multiplication of a $|H| \times k$ and a $k \times d$ matrix, and the addition of two $|H| \times k$ matrices. Note that $\mathrm{LM}(G)$ is a subset of $x_1 B(I) \cup \cdots \cup x_n B(I)$, so $|H| \le |G| \le nd$. Thus the product costs $O(k^{\omega-2}nd^2)$ field operations and the sum costs $O(ndk)$ field operations. In total, the algorithm requires $O(k^{\omega-1}d + k^{\omega-2}nd^2) = O(nk^{\omega-2}d^2)$ field operations. □

The previous algorithm expects a K-basis of the ideal J. In some cases, only a generating set for J as an ideal is known. The next algorithm computes a K-basis using the generating set $\{f_1, \ldots, f_m\}$ and the FGLM data M, by closing the vector space under multiplication with the M_i.

Algorithm 3.4 (IdealBasis).
Input: A finite-dimensional commutative algebra $A = \mathrm{Alg}(M, e)$ and $f_1, \ldots, f_m \in A$.
Output: A K-basis of $\langle f_1, \ldots, f_m \rangle \trianglelefteq A$.

1. Let $E \in K^{\ell_0 \times d}$ be the echelon form of the matrix with rows $f_1, \ldots, f_m$, without zero rows. Set $i := 1$ and $E' := E$.
2. Compute $F := E' \cdot M_i$. Reduce the rows of F modulo E, and let E' be the echelon form of the resulting matrix, without zero rows. Reduce the rows of E modulo E', and add the rows of E' to E.

   Repeat this step until E' is the empty matrix.
3. If $i < n$, set $i := i + 1$, $E' := E$, and go to Step 2. Otherwise return the rows of E.

Proposition 3.5. *Algorithm* 3.4 *is correct and requires $O(nkd^2)$ field operations using classical matrix algorithms, where $k = \max(m, \dim_K(\langle f_1, \ldots, f_m \rangle))$.*

Proof. The correctness is easy to verify; we prove the complexity result. The computation of E in Step 1 can be performed using $O(m^2 d)$ field operations. Step 2 is a loop; assume that it has r iterations. Let ℓ_j be the number of rows in E' at the start of the jth iteration. Then $F = E' M_i$ can be computed using $O(\ell_j d^2)$ field operations. Since E has at most k rows, the reduction of F modulo E requires $O(\ell_j kd)$ field operations, and computation of the echelon form costs $O(\ell_j^2 d)$ field operations. The reduction of E modulo E' can be performed using $O(\ell_{j+1} kd)$ field operations, where $\ell_{r+1} := 0$. Thus every iteration in Step 2 requires $O(\max(\ell_j, \ell_{j+1}) d^2)$ field operations. Since $\ell_1 + \cdots + \ell_r \le \dim_K(\langle f_1, \ldots, f_m \rangle)$, the whole of Step 2 requires $O(kd^2)$ field operations. But Step 2 is executed n times, which proves the overall complexity of $O(nkd^2)$. □

Remark 3.6. The jth iteration in Step 2 can be performed in $O(\max(\ell_j, \ell_{j+1})^{\omega-2} d^2)$ field operations using fast matrix multiplication. But in the worst case, for example if $\ell_j = 1$ for all j, this still yields a complexity of $O(d^3)$ for Step 2. To utilize fast matrix multiplication efficiently, Step 2 can be replaced by the following.

2. For $j = 0, \ldots, \lfloor \log_2(d) \rfloor$: Set $F := E \cdot M^{2^j}$. Reduce F mod E, and let E' be the echelon form of the result. Reduce E mod E', and add the rows of E' to E.

The computation of M^{2^j} can be done by a single squaring in each iteration, costing $O(d^\omega)$ field operations. All other operations can be done in $O(k^{\omega-2}d^2)$ field operations. Thus Step 2 can be performed in $O(d^\omega \log(d))$ field operations, yielding a $O(nd^\omega \log(d))$ alternative for Algorithm 3.4, which is asymptotically faster if k is close to d.

The following algorithm is the inverse of [FGLM93, Procedure 3.1], and a special case of [FGLM93, Procedure 4.1].

Algorithm 3.7 (FGLMDataToGroebner).
Input: $A = \mathrm{Alg}(M, e)$ and $B = \mathrm{B}(A)$.
Output: The Gröbner basis of $\ker \nu$, where $\nu \colon K[\underline{x}] \to A$ is the canonical epimorphism.

1. Set $G := L := \emptyset$ and $N := \{1\}$.

2. Let m be the smallest element in N; set $N := N \setminus \{m\}$.

3. If $m \in B$, then add $mx_1, \dots, mx_n$ to N. Otherwise, if m is not a multiple of an element in L, write $m = B_i x_j$ for some i; let c be the ith row of M_j and add $m - \sum_{\ell=1}^{d} c_\ell B_\ell$ to G and m to L.

4. If $N \neq \emptyset$, then go to Step 2. Otherwise, return G.

Proposition 3.8. *Algorithm* 3.7 *is correct and requires at most $O(nd^2)$ field operations.*

Proof. The correctness follows by Proposition 2.1. The only field operations occur in Step 3. Every Gröbner basis element requires at most d field operations; there are at most nd elements in the Gröbner basis, thus proving the result. □

Definition 3.9. Let A be a finite dimensional K-algebra with basis B, and let $f \in A$. The matrix of the linear map $A \to A \colon a \mapsto a \cdot f$ with respect to B is the *representation matrix* of f.

For example, the M_i in the FGLM-data is the representation matrix of the ith generator. The representation matrix of an arbitrary element can be computed as follows.

Algorithm 3.10 (RepresentationMatrix).
Input: $A = \mathrm{Alg}(M, e)$, $B = \mathrm{B}(A)$, and an element $a \in A$.
Output: The representation matrix of a with respect to $\mathrm{B}(A)$.

1. Let R be the $d \times d$ zero matrix; set $R_1 := a$, where R_1 is the first row of R.

2. For $i = 2, \dots, d$: Write $B_i = B_k x_j$ with $k < i$ and set $R_i := R_k M_j$.

3. Return R.

Proposition 3.11. *Algorithm* 3.10 *is correct and requires at most $O(d^3)$ field operations.*

Proof. The correctness is obvious. Every step of the loop in Step 2 is a vector-matrix multiplication, which requires $O(d^2)$ field operations. Since the loop has $d-1$ steps, this proves the complexity result. □

A central problem in the algorithms is the computation of minimal polynomials of representation matrices; we will use the classical algorithm, which is very efficient in this case.

Algorithm 3.12 (MinimalPolynomial).
Input: The representation matrix F of an element in a finite-dimensional commutative algebra $A = \mathrm{Alg}(M, e)$.
Output: The minimal polynomial of F.

1. Set $v_0 := e$ and $r := 1$; let $E \in K^{1 \times d}$ be the matrix with row e.

2. Set $v_r := v_{r-1} F$, and let w be the reduction of v_r modulo the row span of E. If $w = 0$, go to Step 3. Otherwise, add w to E, increment r, and repeat this step.

3. Compute $\lambda_0, \dots, \lambda_{r-1} \in K$ with $v_r = \lambda_0 v_0 + \dots + \lambda_{r-1} v_{r-1}$.

4. Return $T^r - \lambda_{r-1} T^{r-1} - \dots - \lambda_0$.

Proposition 3.13. *Algorithm* 3.12 *is correct and requires $O(rd^2)$ field operations using classical matrix algorithms, where r is the degree of the minimal polynomial of F.*

Proof. The correctness is easy to verify. The computation and reduction of v_r in Step 2 requires $O(d^2)$ field operations; since this step is executed at most r times, its total cost is $O(rd^2)$. Step 3 is the solution of the linear equation $\lambda \cdot V = v_r$, where $V \in K^{r \times d}$ is the matrix with rows $v_0, \dots, v_{r-1}$, which requires $O(rd^2)$ field operations. □

Remark 3.14. Using techniques similar to Remark 3.6, a runtime complexity of $O(d^\omega \log d)$ can be achieved.

To compute radicals over non-perfect fields, we must use field extensions of the form $K[\sqrt[r]{a}]$ with $a \in K$. After using linear algebra over the extension field, we need to pull the results back to the ground field. The following results make this precise and provide the necessary algorithm.

Remark 3.15. Let $L = K[\alpha]$ be an algebraic extension of K such that α has minimal polynomial $T^r - a \in K[T]$. Then $(1, \alpha, \alpha^2, \ldots, \alpha^{r-1})$ is a K-basis of L, and L embeds into $K^{r \times r}$ via

$$\eta \colon L \to K^{r\times r} \colon c_0 + c_1\alpha + \ldots + c_{r-1}\alpha^{r-1} \mapsto \begin{pmatrix} c_0 & c_1 & \cdots & c_{r-1} \\ ac_{r-1} & c_0 & \cdots & c_{r-2} \\ \vdots & & \ddots & \vdots \\ ac_1 & ac_2 & \cdots & c_0 \end{pmatrix}.$$

We denote the natural extension of η to $L^{k\times \ell} \to L^{rk \times r\ell}$ again by η.

Notation 3.16. Let V be a K-vector space; let $B = (B_1, \ldots, B_d) \in V^d$, and let $M = (M_{ij}) \in M^{k \times d}$. Then $M \cdot B \in V^k$ denotes the k-tuple whose i-th entry is $M_{i1}B_1 + \cdots + M_{id}B_d$.

Algorithm 3.17 (FieldContraction).
Input: An algebraic extension $L = K[\alpha]$ of K such that α has minimal polynomial $T^r - a \in K[T]$, a basis B of a K-vector space V, and $M \in L^{k\times d}$ in row echelon form such that $M \cdot \iota(B)$ is a basis of a subspace $W \leq V \otimes_K L$, where $\iota \colon V \to V \otimes_K L \colon v \mapsto v \otimes 1$.
Output: A matrix $N \in K^{s \times d}$ in row echelon form such that $N \cdot B$ is a K-basis of $\iota^{-1}(W) \leq V$.

1. Note that $M = M^{(0)} \cdot 1 + M^{(1)} \cdot \alpha + \cdots + M^{(r-1)} \cdot \alpha^{r-1}$ with unique $M^{(\ell)} \in K^{k\times d}$. Let $c_1, \ldots, c_{d-k}$ be the indices of columns of M which are not pivot columns. Set $S \in K^{k\times(r-1)(d-k)}$ with $S_{i,j(r-1)+\ell} = M^{(\ell)}_{i,c_{j+1}}$ for all $1 \leq i \leq k$, $0 \leq j < d-k$, and $1 \leq \ell \leq r-1$.
2. Compute a matrix U in row echelon form whose row span is the kernel of S.
3. Return $U \cdot M^{(0)}$.

Proposition 3.18. *Algorithm* 3.17 *is correct and requires $O(rd^\omega)$ field operations in K.*

Proof. Note that $\iota(B) = (B_1 \otimes 1, \ldots, B_d \otimes 1)$ is an L-basis of $V \otimes_K L$, and $(B_1 \otimes 1, B_1 \otimes \alpha, \ldots, B_1 \otimes \alpha^{r-1}, B_2 \otimes 1, \ldots, B_d \otimes \alpha^r)$ is a K-basis of $V \otimes_K L$. Since the row span of M is isomorphic to W as L-vector spaces, the row span of $\eta(M)$ is isomorphic to W as K-vector spaces. An element of W has the form $v \otimes 1$ for some $v \in V$ if and only if it corresponds in this isomorphism to a row ρ such that the entry of ρ corresponding to $B_i \otimes \alpha^j$ is zero for all $1 \leq i \leq d$ and $1 \leq j \leq r-1$. In other words, ρ_i must be zero whenever $i \not\equiv 1 \mod r$. Thus, elements of the form $v \otimes 1$ correspond to those linear combinations of the rows of $\eta(M)$ for which the ith entry is zero for all $i \not\equiv 1 \mod r$. This is exactly the kernel of the matrix consisting of the columns indexed by $\{i \mid 1 \leq i \leq rd \text{ with } i \not\equiv 1 \mod r\}$. However, if p is the index of a pivot column of M, then the columns $(r-1)p+2, \ldots, rp$ of $\eta(M)$ are unit vectors, and the corresponding entries of the kernel elements must be zero. Hence it suffices to compute the kernel of the submatrix of $\eta(M)$ consisting of rows $1, r+1, \ldots, (k-1)r+1$ and columns $r(c_1-1)+2, \ldots, r(c_1-1)+r, \ldots, r(c_m-1)+r$; this is precisely S.

We now prove the complexity result. The matrix S can be computed in $O(kr(d-k))$ field operations. Since $k \leq d$ and $d-k \leq d$, Step 1 can be performed in $O(rd^2)$ field operations. The kernel of S can be read off a column echelon form of S, which can be computed in $O(k(r-1)(d-k)\operatorname{rank}(S)^{\omega-2})$ field operations by [Sto94, Theorem 2.10]. But $\operatorname{rank}(S) \leq \min(k, (r-1)(d-k)) \leq k$ and $k, (d-k) \leq d$, so Step 2 requires $O(rd^\omega)$ field operations. Finally, the product $U \cdot M^{(0)}$ requires $O(d^\omega)$ field operations. □

We illustrate the algorithm and the proof with an example.

Example 3.19. Let $K = \mathbb{F}_2(t)$ and $L = \mathbb{F}_2(\sqrt{t}) = K[\sqrt{t}]$. Let $V = K^{1\times 4}$ with standard basis B, and let

$$M = \begin{pmatrix} 1 & 0 & 0 & t \\ 0 & 1 & 0 & \sqrt{t} \\ 0 & 0 & 1 & \sqrt{t} \end{pmatrix} \in L^{3\times 4}; \qquad \text{then} \qquad \eta(M) = \begin{pmatrix} 1&0&0&0&0&0&t&0 \\ 0&1&0&0&0&0&0&t \\ 0&0&1&0&0&0&0&1 \\ 0&0&0&1&0&0&t&0 \\ 0&0&0&0&1&0&0&1 \\ 0&0&0&0&0&1&t&0 \end{pmatrix} \in K^{6\times 8}.$$

Let $\rho \in K^{1\times 6}$. Then $\rho \cdot \eta(M)$ in the row space of $\eta(M)$ corresponds to an element $v \otimes 1$ of W if and only if the entries in the columns with even indices are zero. However, the columns 2, 4, and 6 are unit vectors, which forces $\rho_2 = \rho_4 = \rho_6 = 0$. Hence it suffices to compute the kernel of the matrix $S = \begin{pmatrix} 0 & 1 & 1 \end{pmatrix}^{\mathrm{tr}}$, which is the row span of $U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}$. The result is

$$UM^{(0)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & t \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & t \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

Thus

$$\langle (1,0,0,t), (0,1,0,\sqrt{t}), (0,0,1,\sqrt{t}) \rangle_L \cap K^{1\times 4} = \langle (1,0,0,t), (0,1,1,0) \rangle_K.$$

(This computation is an intermediate step in Algorithm 4.8 to compute the radical of $\langle x_1^2 + t, x_2^2 + t \rangle \trianglelefteq \mathbb{F}_2(t)[x_1, x_2]$.)

Finally, we provide an algorithm which, given a decomposition $A = A_1 \oplus \cdots \oplus A_k$ as algebras and the FGLM data of A, computes the FGLM data of all A_i. This is an adaptation of [FGLM93, Proposition 3.1]. We need the following two technical results. The first concerns the complexity of an algorithm which is split into sub-algorithms on smaller input; the second states that the monomial basis of a direct summand of an algebra is contained in the monomial basis of the bigger algebra.

Lemma 3.20. *Let $f\colon \mathbb{N} \to \mathbb{R}_{>0}$ with $f(d) \in O(d^s)$ for some $s \geq 1$; let $g(d_1, \dots, d_r) = f(d_1) + \cdots + f(d_r)$ for some $r \in \mathbb{N}$. There exists $M > 0$ such that $|g(d_1, \dots, d_r)| \leq M(d_1 + \ldots + d_r)^s$ for all $d_1, \dots, d_r \in \mathbb{N}$.*

Proof. As $f(d) \in O(d^s)$, there exists $M' > 0$ and $d_0 > 0$ such that $|f(d)| \leq M'd^s$ for all $d > d_0$. Set $M := \max\{M, f(1), \dots, f(d_0)\}$. Then

$$|g(d_1, \dots, d_r)| \leq |f(d_1)| + \cdots + |f(d_r)| \leq Md_1^s + \cdots + Md_r^s \leq M(d_1 + \cdots + d_r)^s. \qquad \square$$

Lemma 3.21. *Let $A, A_1, \dots, A_k$ be finite-dimensional commutative K-algebras such that $A \cong A_1 \oplus \cdots \oplus A_k$; let $\nu\colon K[\underline{x}] \to A$ and $\nu_i\colon K[\underline{x}] \to A_i$ be the canonical epimorphisms. Then*

$$\mathrm{B}(A_i) = \{ b \in \mathrm{B}(A) \mid \nu_i(b) \notin \langle \nu_i(n) \mid n \in \mathrm{B}(A) \text{ with } n < b \rangle_K \}.$$

Proof. For $M \subseteq \mathrm{Mon}(K[\underline{x}])$ and $m \in \mathrm{Mon}(K[\underline{x}])$ set $M_{<m} := \{ n \in M \mid n < m \}$. Assume without loss of generality $A = A_1 \oplus \cdots \oplus A_k$. Then $\nu = \nu_1 + \cdots + \nu_k$, hence

$$\langle \nu(n) \mid n \in \mathrm{Mon}(K[\underline{x}])_{<m} \rangle_K = \langle \nu_1(n) \mid n \in \mathrm{Mon}(K[\underline{x}])_{<m} \rangle_K \oplus \cdots \oplus \langle \nu_k(n) \mid n \in \mathrm{Mon}(K[\underline{x}])_{<m} \rangle_K$$

for all $m \in \mathrm{Mon}(K[\underline{x}])$. Note that $\langle \nu(n) \mid n \in \mathrm{Mon}(K[\underline{x}])_{<m} \rangle_K = \langle \nu(n) \mid n \in \mathrm{B}(A)_{<m} \rangle_K$ by Proposition 2.1, so in particular $\langle \nu_i(n) \mid n \in \mathrm{Mon}(K[\underline{x}])_{<m} \rangle_K = \langle \nu_i(n) \mid n \in \mathrm{B}(A)_{<m} \rangle_K$. Thus using Proposition 2.1 again,

$$\begin{aligned} \mathrm{B}(A_i) &= \{ b \in \mathrm{Mon}(K[\underline{x}]) \mid \nu_i(b) \notin \langle \nu_i(n) \mid n \in \mathrm{Mon}(K[\underline{x}])_{<b} \rangle_K \} \\ &= \{ b \in \mathrm{Mon}(K[\underline{x}]) \mid \nu_i(b) \notin \langle \nu_i(n) \mid n \in \mathrm{B}(A)_{<b} \rangle_K \}. \end{aligned}$$

But if $\nu_i(b) \notin \langle \nu_i(n) \mid n \in \mathrm{B}(A)_{<b} \rangle_K$, then $\nu(b) \notin \langle \nu(n) \mid n \in \mathrm{B}(A)_{<b} \rangle_K$, thus proving the lemma. $\square$

Algorithm 3.22 (MatphiDirectSum).
Input: $A = \mathrm{Alg}(M, e)$ and $A_1, \dots, A_k \trianglelefteq A$ such that $A = A_1 \oplus \cdots \oplus A_k$.
Output: A list $(M_1, B_1), \dots, (M_k, B_k)$ such that $A_i \cong \mathrm{Alg}(M_i)$ and $\mathrm{B}(A_i) = B_i$.

1. Let S_i be a basis of A_i; set $B_i := ()$, $C_i := ()$, and let $\mathcal{M}^{(i)} := (0, \dots, 0)$ be a list of n zero matrices in $K^{\dim(A_i)\times\dim(A_i)}$, for $i = 1, \dots, k$.

   Set $N := [1]$ and $NF := [e]$.

2. Remove the first elements of N and NF; let m and v be the removed elements, respectively.

3. Note that $v = w_1 + \cdots w_k$ with $w_i \in A_i$. Let c_i be the coordinates of w_i with respect to S_i.

   For all $m' \in B_i$ and x_j such that $m = m'x_j$, set the ℓ-th row of $\mathcal{M}_j^{(i)}$ to c_i, where ℓ is the index of m' in B_i. If $c_i \notin \langle C_i \rangle$, then add m to B_i and c_i to C_i.

4. If $c_i \notin \langle C_i \rangle$ for some i in Step 3, then compute mx_j for all $j = 1, \ldots, n$. If $mx_j \notin N$, then insert it such that N stays sorted, and insert $v \cdot M_j$ at the corresponding position in NF.

5. If $N \neq \emptyset$, then go to Step 2. Otherwise, let T_i be the matrix with rows C_i; compute $\mathcal{F}^{(i)} := (\mathcal{M}_1^{(i)} T_i^{-1}, \ldots, \mathcal{M}_n^{(i)} T_i^{-1})$. Return the $\mathcal{F}^{(i)}$ and B_i.

Proposition 3.23. *Algorithm* 3.22 *is correct and requires* $O(nd^3)$ *field operations.*

Proof. The B_j are monomial bases of A_j by Lemma 3.21. Let $\varphi \colon V \to V$ be a linear map of a vector space V over K of dimension d and let X and V be bases of V. Then ${}^X\varphi^Y \in K^{d\times d}$ denotes the matrix whose ith row are the coordinates of $\varphi(X_i)$ with respect to Y. In this notation, $M_j = {}^{\nu(\mathrm{B}(A))}\rho_i{}^{\nu(\mathrm{B}(A))}$, where $\rho_i \colon A \to A \colon a \mapsto a \cdot a_i$ and $\mathrm{B}(A)$ is the monomial basis of A. Moreover, $\mathcal{M}_i^{(j)} = {}^{\nu_j(\mathrm{B}(A_j))}\sigma_i{}^{C_j}$ and $T_j = {}^{\nu_j(\mathrm{B}(A_j))}(\mathrm{id}_{A_j})^{C_j}$, where $\sigma_i = (\rho_i)_{|A_j}$, hence $\mathcal{F}_i^{(j)} = {}^{\nu_j(\mathrm{B}(A_j))}\sigma_i{}^{\nu_j(\mathrm{B}(A_j))}$. That is, $\mathcal{F}^{(j)}$ are the FGLM data of A_j. This proves the correctness.

Steps 1 and 2 do not involve field operations. Let $\mathcal{S}$ be the matrix whose rows are the elements of the S_i. The c_i in Step 3 can be computed as $v \cdot \mathcal{S}^{-1}$. The computation of $\mathcal{S}^{-1}$ costs $O(d^\omega)$ field operations, and has to be performed only once during the course of the algorithm. The computation of $(c_1, \ldots, c_k)$ can be performed in $O(d^2)$ field operations, and must be computed in every iteration of the loop. The total number of entries of all $\mathcal{M}_j^{(i)}$ is at most nd^3, so the setting of the matrix entries costs $O(nd^3)$ field operations. By keeping an echelon form for a basis of $\langle C_i \rangle$, we can check $c_i \in \langle C_i \rangle$ in $O(d_i^2)$ field operations, where $d_i = \dim(A_i)$. By Lemma 3.20, the decision $c_i \in \langle C_i \rangle$ for all i can be performed in $O(d^2)$ field operations. Since the loop has at most nd iterations, Step 3 has a total cost of $O(nd^3)$ field operations. The cost for $v \cdot M_j$ in Step 4 is $O(nd^2)$; this step is only executed if $c_i \notin \langle C_i \rangle$ for some i, which occurs precisely d times. Thus Step 4 has a total cost of $O(nd^3)$. The computation of F_i^{-1} and $\mathcal{F}^{(i)}$ can be performed in $O(nd_i^\omega)$ field operations, for every i, so Step 5 can be performed in $O(nd^\omega)$ field operations, by Lemma 3.20. This concludes the complexity analysis. □

4 An algorithm to compute the radical

Our algorithm to compute the radical of a zero-dimensional ideal over perfect fields is based on the method of Krick and Logar [KL91], which in turn is based on a result by Seidenberg. This method was extended by Kemper [Kem02] to zero-dimensional ideals over arbitrary fields. In this section, we translate the theoretical results and the algorithms to the language of linear algebra.

The idea of the algorithm to compute the radical is used in the primary decomposition algorithm in Section 5. Furthermore, it can be used to compute the prime ideal associated to a primary ideal. But an algorithm to compute the radical is also of independent interest. For instance, it can be used to compute all zeroes of an ideal. A different method to do this efficiently is described in [Rou99].

4.1 Theoretical results

We are dealing with two notions of radicals. On the one hand, the radical of $I \trianglelefteq K[\underline{x}]$ is defined as $\sqrt{I} := \{f \in K[\underline{x}] \mid f^m \in I \text{ for some } m \in \mathbb{N}\}$. On the other hand, the (Jacobson) radical of a finite dimensional commutative K-algebra A, is defined as $\mathrm{rad}(A) := \{a \in A \mid a^m = 0 \text{ for some } m \in \mathbb{N}\}$. Note that $\mathrm{rad}(A) \trianglelefteq A$. The two notions are connected by the following remark, which is apparent from the definitions.

Remark 4.1. Let $I \trianglelefteq K[\underline{x}]$ be a zero-dimensional ideal and $A := K[\underline{x}]/I$. Then $K[\underline{x}]/\sqrt{I} \cong A/\mathrm{rad}(A)$.

We recall the definition of square-free and separable polynomials.

Definition 4.2. Let $f \in K[T]$ with factorization $f = f_1^{e_1} \cdots f_r^{e_r}$ into irreducibles. Then f is *square-free* if $e_i = 1$ for all i, and f is *separable* if it is square-free over every extension field of K.

Note that the notions of square-free and separable are the same if K is perfect.

We begin by translating the results of Seidenberg and Kemper to the language of algebras. (Seidenberg's formulation of the next result uses square-free polynomials instead of separable polynomials.)

Lemma 4.3 ([Sei74, Lemma 92]). *Let A be a finite-dimensional commutative K-algebra generated by $a_1, \ldots, a_n$, and let $\mu_i \in K[T]$ be the minimal polynomial of a_i for $i = 1, \ldots, n$. If μ_i is separable for all i, then* $\operatorname{rad}(A) = \{0\}$.

Proof. Note that $A = K[a_1, \ldots, a_n]$; proceed by induction on n. If $n = 1$ then $A = K[a_1] \cong K[T]/\langle \mu_1 \rangle$ and the result is trivial, so assume now $n > 1$. Let $\mu_1 = h_1 \cdots h_\ell$ be a factorization into irreducibles, so $A = A_1 \oplus \cdots \oplus A_\ell$ with $A_i = \ker(h_i(a_1))$. Since $\operatorname{rad}(A) = \operatorname{rad}(A_1) \oplus \cdots \oplus \operatorname{rad}(A_\ell)$, we can assume that $A = A_1$ and μ_1 is irreducible, so $K[a_1]$ is a field. But $A = K[a_1][a_2, \ldots, a_n]$, so A is a $K[a_1]$-algebra generated by $a_2, \ldots, a_n$. The minimal polynomial of a_j over $K[a_1]$ is a divisor of μ_j, hence in particular separable, so the result follows by induction. □

Thus to compute the radical of an ideal it is enough to ensure that the minimal polynomials of the generators are separable.

If K is a perfect field, this can be accomplished by computing the square-free part of the polynomials, since square-free polynomials over perfect fields are separable. This yields the algorithms of Krick and Logar [KL91]. If K is a non-perfect field, there are square-free polynomials which are non-separable. This problem is overcome by Kemper by using the separable part of a polynomial, which requires the use of extension fields.

Definition 4.4. Let $f \in K[T]$. If $f = \prod_{i=1}^{k} (T - \alpha_i)^{e_i}$, where α_i are the distinct roots of f in an algebraic closure of K, then $\operatorname{sep}(f) := \prod_{i=1}^{k} (T - \alpha_i)$ is the *separable part* of f.

Proposition 4.5 ([Kem02, Theorem 7]). *Let A be a finite-dimensional commutative K-algebra generated by $a_1, \ldots, a_n$, and let $\mu_i \in K[T]$ be the minimal polynomial of a_i for $i = 1, \ldots, n$. Let L/K be an extension field such that $\sigma_i := \operatorname{sep}(\mu_i) \in L[T]$ for all i. Then $\operatorname{rad}(A \otimes_K L) = \langle \sigma_1(a_1 \otimes 1), \ldots, \sigma_n(a_n \otimes 1) \rangle$. Furthermore, $\iota^{-1}(\operatorname{rad}(A \otimes_K L)) = \operatorname{rad}(A)$, where $\iota \colon A \to A \otimes_K L \colon a \mapsto a \otimes 1$.*

Proof. Let $J := \langle \sigma_1(a_1 \otimes 1), \ldots, \sigma_n(a_n \otimes 1) \rangle$. Clearly $\sigma_i(a_i \otimes 1) \in \operatorname{rad}(A \otimes_K L)$, so $J \subseteq \operatorname{rad}(A \otimes_K L)$. Since $\operatorname{rad}(A \otimes_K L/J) = \{0\}$ by Lemma 4.3, equality holds, proving the first part of the proposition. The second part is easily verified. □

4.2 Algorithms

Kemper gives an algorithm to compute the separable part of a polynomial [Kem02, Algorithm 1], which is a variant of [KR00, Proposition 3.7.12]. We follow [vzGG99, Exercise 14.27] instead, which results in an algorithm with better runtime complexity.

Algorithm 4.6 (SeparablePart).
Input: A polynomial $f \in K[T]$ of degree d.
Output: A field extension E/K and the separable part $\operatorname{sep}(f) \in E[T]$.

1. Compute $g := \gcd(f, f')$ and $h := f/g$. If $\operatorname{char}(K) = 0$ or $g = 1$, then return K and h. Otherwise, replace g by $g/\gcd(g, h^n)$.

2. Let $p := \operatorname{char}(K)$. If K is finite, then set $E := K$; if $K = \mathbb{F}_q(t_1, \ldots, t_m)$ for indeterminates $t_1, \ldots, t_m$, then set $E := \mathbb{F}_q(\sqrt[p]{t_1}, \ldots, \sqrt[p]{t_m})$. Compute a pth root s of $E[T]$ and compute E and $\operatorname{sep}(s)$ recursively.

3. Return E and $h \cdot \operatorname{sep}(s)$.

If $|K| = q < \infty$ has characteristic p, then the pth root of $\alpha \in K$ is $\alpha^{q/p}$, which can be computed in $\log(q/p)$ field operations. But if $K = \mathbb{F}_q(t_1, \ldots, t_m)$, then the pth root of $\alpha = \sum_{i \in \mathbb{N}^m} \alpha_i t_1^{i_1} \cdots t_m^{i_m} \in K$ is $\sum \alpha_i^{q/p} \sqrt[p]{t_1}^{i_1} \cdots \sqrt[p]{t_m}^{i_m}$; thus the complexity to compute a pth root cannot be measured in field operations, since it depends on the number of terms of α, which is unbounded. We therefore have to treat the computation of pth roots separately in the following complexity analysis.

Note that $\mathbb{F}_q(\sqrt[p]{t_1},\dots,\sqrt[p]{t_m}) \cong \mathbb{F}_q(t_1,\dots,t_m)$. Thus field operations in $\mathbb{F}_q(\sqrt[p]{t_1},\dots,\sqrt[p]{t_m})$ are just as expensive as field operations in $\mathbb{F}_q(t_1,\dots,t_m)$.

Let $\mathsf{M}\colon \mathbb{N} \to \mathbb{R}$ be a multiplication time for $K[T]$; that is, two polynomials $f, g \in K[T]$ of degree at most d can be computed in $O(\mathsf{M}(d))$ field operations.

Proposition 4.7. *Algorithm* 4.6 *is correct. If* $\operatorname{char}(K) = 0$, *the algorithm requires* $O(\mathsf{M}(d)\log(d))$ *field operations; if* $|K| = q < \infty$ *has characteristic* p, *it requires* $O(\mathsf{M}(d)\log(d) + d\log(q/p))$ *field operations; and if* $K = \mathbb{F}_q(t_1,\dots,t_m)$, *it requires* $O(\mathsf{M}(d)\log(d))$ *field operations and at most* $d+1$ *computations of* p*th roots of elements in* K. *Moreover, if* K *is perfect, then* $E = K$, *and if* $K = \mathbb{F}_q(t_1,\dots,t_m)$, *then* $E = \mathbb{F}_q(\sqrt[r]{t_1},\dots,\sqrt[r]{t_m})$ *with* $r \le \upsilon_p(d)$, *where* υ_p *is the* p*-adic valuation on* $\mathbb{Z}$.

Proof. This is an easy exercise. (Note that $\gcd(g, h^n) = \gcd(g, h^n \bmod g)$, and $h^n \bmod g$ can be computed in $O(\log(d))$ polynomial multiplications using repeated squaring.) □

Algorithm 4.8 (Radical).
Input: The Gröbner basis of a zero-dimensional ideal $I \trianglelefteq K[\underline{x}]$.
Output: The Gröbner basis of $\sqrt{I}$.

1. Let M be the FGLM data of I. For $1 \le i \le n$ compute the minimal polynomial $\mu_i \in K[T]$ of M_i.
2. Compute the separable parts $\sigma_i := \operatorname{sep}(\mu_i) \in L[T]$, where $L = K$ if K is perfect and $L = \mathbb{F}_q(\sqrt[r]{t_1},\dots,\sqrt[r]{t_m})$ for some p-power r if $K = \mathbb{F}_q(t_1,\dots,t_m)$.
3. Compute an L-basis C' for the ideal $J = \langle \sigma_1(x_1 \otimes 1),\dots,\sigma_n(x_n \otimes 1)\rangle / I \otimes_K L \trianglelefteq A \otimes_K L$ using IdealBasis.
4. If K is perfect, set $C := C'$ and go to Step 5. Otherwise, apply FieldContraction m times to compute a basis C for $\iota^{-1}(J)$, where $\iota\colon A \to A \otimes_K L\colon a \mapsto a \otimes 1$.
5. Return IdealBasisToGroebner(I, C).

Remark 4.9. The tensor product $A \otimes_K L$ is just an extension of the base field. In particular, a Gröbner basis for I is also a Gröbner basis for $I \otimes_K L$, and the FGLM data for I is also the FGLM data for $I \otimes_K L$.

Proposition 4.10. *Algorithm* 4.8 *is correct. If* $\operatorname{char}(K) = 0$, *then it requires* $O(nd^3)$ *field operations; if* $\operatorname{char}(K) = p$ *and* $|K| = q$, *then it requires* $O(nd^3 + nd\log(q/p))$ *field operations; and if* $K = \mathbb{F}_q(t_1,\dots,t_m)$, *then it requires at most* $O(nd^3 + md^{\omega+1})$ *field operations and at most* nd *computations of* p*th roots of elements in* K.

Proof. Note that $J = \operatorname{rad}(A \otimes_K L)$ and $\langle C\rangle_K = \iota^{-1}(J) = \operatorname{rad}(A)$ by Proposition 4.5. Since $A/\operatorname{rad}(A) \cong K[\underline{x}]/\sqrt{I}$ by Remark 4.1, the algorithm is correct. We prove the complexity statement. The FGLM data can be computed in $O(nd^3)$ field operations, and each minimal polynomial can be computed in $O(d^3)$ field operations; thus Step 1 can be computed in $O(nd^3)$ field operations. Using classical polynomial multiplication, Step 2 can be performed using $O(nd^2\log(d))$, $O(nd^2\log(d) + nd\log(q/p))$, or $O(nd^2\log(d))$ field operations plus nd computations of pth roots, respectively, depending on the field, by Proposition 4.7. Let $\sigma_i = c_0 + c_1T + \cdots + c_kT^k$. Then $\sigma_i(x_i \otimes 1) = e_d(c_0I_d + c_1M_i + \cdots + c_kM_i^k)$, where e_d is the dth standard basis vector. Thus $\sigma_i(x_i \otimes 1)$ can be computed in at most k vector-matrix multiplications and k vector additions. Since $k \le d$, the n elements $\sigma_1(x_1 \otimes 1),\dots,\sigma_n(x_n \otimes 1)$ can be computed using $O(nd^3)$ field operations. The basis computation costs again $O(nd^3)$ operations by Proposition 3.5, so Step 3 can be performed in $O(nd^3)$ field operations. Step 4 is $O(1)$ if K is perfect and $O(mrd^\omega)$ otherwise by Proposition 3.18. But $r \le \max\{\deg(\mu_i) \mid 1 \le i \le n\} \le d$; thus Step 4 costs $O(md^{\omega+1})$ field operations if $K = \mathbb{F}_q(t_1,\dots,t_m)$. Finally, Step 5 can be performed using $O(nd^3)$ field operations by Proposition 3.3, which finishes the proof. □

5 Primary decomposition over infinite fields

In this section, we describe an algorithm for the primary decomposition of zero-dimensional ideals over infinite fields. The algorithm is based on the following basic fact.

Proposition 5.1. *Let A be a finite-dimensional commutative K-algebra and $a \in A$ with minimal polynomial μ_a. Assume that μ_a factors as $\mu_a = \mu_1 \cdots \mu_\ell$ such that the μ_i are pairwise coprime (not necessarily irreducible). Set $A_i := \ker(\mu_i(a))$ (the kernel of multiplication by $\mu_i(a)$). Then $A_i \trianglelefteq A$ for all i, and $A = A_1 \oplus \cdots \oplus A_\ell$ is a direct sum decomposition of algebras.*

Also note that if $I \trianglelefteq K[\underline{x}]$ is a zero-dimensional ideal with primary decomposition $I = \bigcap_{i=1}^k Q_i$, then

$$A := K[\underline{x}]/I \cong K[\underline{x}]/Q_1 \oplus \cdots \oplus K[\underline{x}]/Q_k =: A_1 \oplus \cdots \oplus A_k$$

by the Chinese Remainder Theorem. Thus computing the primary decomposition of I is equivalent to decomposing A into a direct sum of algebras which are not decomposable any further. The key component of the algorithm is the computation and factorization of the minimal polynomial of an element $a := \lambda_1 x_1 + \cdots + \lambda_n x_n + I \in K[\underline{x}]/I$, with $\lambda \in K^n$. If a suitable element is found, then the ideal is split into ideals of smaller residue class dimension using Proposition 5.1; the smaller ideals are handled recursively. This idea was already used in [GTZ88], [Kre89], [BW93], and [Ste05], where the first three algorithms assume that the field has characteristic zero, and the last algorithm assumes positive characteristic. In [GTZ88] and [Ste05] the vector λ is chosen at random until a suitable element is found, whereas [Kre89] and [BW93] choose λ in a deterministic manner. In [Kre89], no approximation is given to how many elements will suffice, but [BW93] show that at most $\prod_{i=1}^n \left(\binom{d^i}{2} + 1\right)$ choices are necessary.

We will develop all results in the language of algebras. This allows us to give relatively short proofs and unify the description for characteristic zero and positive characteristic. Our approach is deterministic, and we show that $(d-1)^2+1$ choices suffice to find a suitable element. This enables us to give a complete complexity analysis of the algorithms.

5.1 Theoretical results

To estimate the number of necessary choices for $a \in A$, we need the following combinatorial lemma.

Lemma 5.2. *Let A/K be a separable extension of fields with $[A : K] = d < \infty$. There are at most $d-1$ maximal subfields F/K of A/K.*

Proof. Let N/K be the normal closure of A/K with Galois group $G := \operatorname{Aut}_K(N)$, and let $H := \operatorname{Aut}_A(N) \leq G$; there is a Galois correspondence between the subfields of A/K and the subgroups of G containing H. In particular, the maximal subfields of A/K are in bijection to the subgroups of G which contain H minimally, that is, the $U \leq G$ with $H \lneqq U$ such that there is no $V \leq G$ with $H \lneqq V \lneqq U$. Thus it suffices to prove that there are at most $d-1$ subgroups of G which contain H minimally.

Let G act on $\Omega := G/H$ by left multiplication; then H is the stabilizer of the coset $1 \cdot H$. There is a bijection between the subgroups of G which contain H minimally, and the minimal blocks of Ω which contain H, see for example [DM96, Theorem 1.5A]. The minimal blocks of Ω containing H are generated by H and one other element of Ω. Thus there are at most $|\Omega| - 1 = [G : H] - 1 = d - 1$ minimal blocks of Ω containing H. □

Let A be a finite-dimensional commutative K-algebra. Recall that A is *semi-simple* if $\operatorname{rad}(A) = \{0\}$, or equivalently, if A is isomorphic to a direct sum of field extensions of K. A semi-simple algebra is *separable* if all of those field extensions are separable.

Definition 5.3. Let $A = F_1 \oplus \cdots \oplus F_k$ for finite field extensions F_i/K. Let $K \leq L \leq F_1$ be a subfield, and let $\sigma_i \colon L \to F_i$ be field monomorphisms for $i = 2, \ldots, k$; set $\sigma := (\sigma_2, \cdots, \sigma_k)$. Then

$$\Delta(L, \sigma) := \{x + \sigma_2(x) + \cdots + \sigma_k(x) \mid x \in L\} \subseteq A$$

is the *twisted diagonal* with respect to L and σ.

Note that a twisted diagonal is a subalgebra of A.

Theorem 5.4. *Let $I \trianglelefteq K[\underline{x}]$ be a zero-dimensional ideal and $A = K[\underline{x}]/I$ such that $A/\operatorname{rad}(A)$ is separable. Let $\nu \colon K[\underline{x}] \to A$ be the natural epimorphism and $d := \dim_K(A)$. For $\lambda = (\lambda_1, \ldots, \lambda_n) \in K^n$ denote by μ_λ the minimal polynomial of $\nu(\lambda_1 x_1 + \cdots \lambda_n x_n)$.*

1. *If I is a prime ideal, then $C = \{\lambda \in K^n \mid \deg(\mu_\lambda) < d\}$ is a finite union of at most $d-1$ proper subspaces of K^n.*

2. *If I is a primary ideal but not prime, then $C = \{\lambda \in K^n \mid \mu_\lambda$ is irreducible$\}$ is a proper subspace of K^n.*

3. *If I is not primary, then $C = \{\lambda \in K^n \mid \mu_\lambda$ is the power of an irreducible polynomial$\}$ is a finite union of proper subspaces of K^n, and is contained in a finite union of at most $d-1$ proper subspaces of K^n.*

Proof. Let $\varphi\colon K^n \to A\colon \lambda \mapsto \sum_{i=1}^n \lambda_i \nu(x_i)$.

To prove 1, note that a zero-dimensional prime ideal is a maximal ideal, so A is a field. Clearly, $\deg(\mu_\lambda) < d$ if and only if $\varphi(\lambda)$ lies in a proper subfield F of A/K, so $C = \bigcup \varphi^{-1}(F)$, where F/K runs over all maximal subfields of A/K. Since A is generated by $\nu(x_1), \dots, \nu(x_n)$ we see $\varphi^{-1}(F) \subsetneqq K^n$ for all F, and by Lemma 5.2 there are at most $d-1$ possibilities for F; this proves part 1.

We now prove part 2. By the Wedderburn-Malcev Theorem, see for example [Jac89, pp. 374-375], there exists a subalgebra S of A with $S \cong A/\operatorname{rad}(A)$ such that $A = S \oplus \operatorname{rad}(A)$ as a vector space; moreover, every separable subalgebra T of A is contained in S. Thus the elements of A with irreducible minimal polynomial are precisely the elements of S, hence $C = \varphi^{-1}(S)$.

It remains to prove part 3. Consider $\rho\colon A \to A/\operatorname{rad}(A)$. The minimal polynomial of $a := \nu(\lambda_1 x_1 + \dots + \lambda_n x_n) \in A$ is a power of an irreducible polynomial if and only if the minimal polynomial of $\rho(a)$ is irreducible. Hence we can assume that A is semi-simple, so $A = F_1 \oplus \dots \oplus F_k$ is a direct sum of separable field extensions of K.

Let $a = a_1 + \dots + a_k$ with $a_i \in F_i$ for all i; then $\mu_a = \operatorname{lcm}(\mu_{a_1}, \dots, \mu_{a_k})$, and since μ_a is irreducible this yields $\mu_{a_i} = \mu_a$ for all i. In particular, $K[a_1] \cong K[a_i]$ for all i, and there exist field isomorphisms $\sigma_i\colon K[a_1] \to K[a_i]$ with $\sigma_i(a_1) = a_i$. Thus $a = a_1 + \sigma_2(a_1) + \dots + \sigma_k(a_1)$, that is, a lies in a twisted diagonal of A. Clearly there are only finitely many twisted diagonals. Taking preimages under φ as in the other cases shows that C is a finite union of proper subspaces.

We conclude the proof by showing that C is contained in a union of at most $d-1$ proper subspaces. We may assume $[F_1 : K] \le d/2$. Assume first $F_1 \cong F_2$. Then every twisted diagonal is contained in some subalgebra $A_\sigma := \{a_1 + \sigma(a_1) + a_3 + \dots + a_k \mid a_i \in F_i\}$, where $\sigma\colon F_1 \to F_2$ is an isomorphism. There are at most $d/2$ isomorphisms, hence at most $d/2 \le d-1$ subalgebras A_σ. Since A_σ is a subalgebra, $\varphi^{-1}(A_\sigma) \subsetneqq K^n$.

Now assume that F_1 is isomorphic to a proper subfield of F_2. Then every twisted diagonal is contained in some subalgebra $F_1 \oplus M_2 \oplus F_3 \oplus \dots \oplus F_k$, where M_2 is a maximal subfield of F_2/K. By Lemma 5.2, there are at most $d-2$ maximal subfields of F_2/K.

Finally, assume that F_1 is not isomorphic to a subfield of F_2. Then every twisted diagonal is contained in some subalgebra $M_1 \oplus F_2 \oplus \dots \oplus F_k$, where M_1 is a maximal subfield of F_1/K, of which there are at most $d/2 - 1 < d - 1$. □

Note that in the last part, C is not necessarily a union of at most $d-1$ proper subspaces. For example, let $F_1 = \dots = F_k$ be a quadratic extension of K and $A = F_1 \oplus \dots \oplus F_k$. Then $d = \dim_K(A) = 2k$, but there are 2^{k-1} twisted diagonals. However, C is always *contained* in a union of at most $d-1$ proper subspaces.

Corollary 5.5. *Let $\Lambda \subseteq K^n$ be a set of size $(d-1)^2 + 1$ such that any subset of size n is linearly independent. If I is prime, then μ_λ has degree d for at least one $\lambda \in \Lambda$; if I is primary but not prime, then μ_λ is a proper power for at least one $\lambda \in \Lambda$; if I is not primary, then μ_λ has at least two coprime factors for at least one $\lambda \in \Lambda$.*

Proof. It suffices to show that Λ cannot be a subset of a union of $d-1$ proper subspaces of K^n. Suppose this statement is false, so $\Lambda \subseteq \bigcup_{i=1}^{d-1} V_i$ with $(n-1)$-dimensional subspaces $V_i \le K^n$. Then $|V_i \cap \Lambda| \le d-1$ by assumption on Λ, hence $|\Lambda| \le \sum_{i=1}^{d-1} |V_i \cap \Lambda| \le (d-1)^2$, a contradiction. □

The following two results show how to construct such sets Λ if K is an infinite field.

Proposition 5.6. *Let K be a field of characteristic zero and $n \in \mathbb{N}$. Let $\lambda^{(i)} := (\binom{i}{0}, \dots, \binom{i}{n-1})$ for $i \in \mathbb{N}$; set $\Lambda := \{\lambda^{(i)} \mid i \in \mathbb{N}\}$. Then every n-element subset of Λ is linearly independent.*

Proof. This follows by [GV85, Corollary 2]. □

Remark 5.7. The sequence $\lambda^{(0)}, \lambda^{(1)}, \ldots$ is Pascal's triangle, truncated after n entries, so $\lambda^{(i+1)} = (1, \lambda_1^{(i)} + \lambda_2^{(i)}, \lambda_2^{(i)} + \lambda_3^{(i)}, \ldots, \lambda_{n-1}^{(i)} + \lambda_n^{(i)})$. Thus $\lambda^{(i+1)}$ can be computed from $\lambda^{(i)}$ in $O(n)$ field operations.

Proposition 5.8. *Let K be a field of positive transcendence degree, and let $n \in \mathbb{N}$; let $x \in K$ be transcendent over the prime field of K. For $i \in \mathbb{N}$ let $\lambda^{(i)} := (m_{i1}, \ldots, m_{in}) \in K^n$, where*

$$m_{ij} = \begin{cases} x^{(j-1)(i-j)} & \text{if } i \geq j, \\ 0 & \text{otherwise}; \end{cases}$$

set $\Lambda := \{\lambda^{(i)} \mid i \in \mathbb{N}\}$. Then every n-element subset of Λ is linearly independent.

Proof. For $I = \{r_1 < \cdots < r_n\} \subseteq \Lambda$ let $M_I := (m_{r_i j})_{i,j} \in K^{n \times n}$. We prove $\deg(\det(M_I)) = \sum_{j=1}^{n}(j-1)(r_j - j)$ by induction on n. The claim is trivially true for $n = 1$, so assume now $n > 1$. By Laplace expansion, $\det(M_I) = \sum_{i=1}^{n}(-1)^{n-i} m_{r_i n} \det(M_{I \setminus \{r_i\}})$. If $m_{r_i n} \neq 0$, then

$$\deg(m_{r_i n} \det(M_{I \setminus \{r_i\}})) = \sum_{j=1}^{i-1}(j-1)(r_j - j) + \sum_{j=i}^{k-1}(j-1)(r_{j+1} - j) + (k-1)(r_i - k),$$

so if $i < n$, then

$$\begin{aligned} \deg(m_{r_n n} \det(M_{I \setminus \{r_n\}})) - \deg(m_{r_i n} \det(M_{I \setminus \{r_i\}})) &= \sum_{j=i}^{k-1}(j-1)(r_j - r_{j+1}) + (k-1)(r_k - r_i) \\ &> (k-1)\sum_{j=i}^{k-1}(r_j - r_{j+1}) + (k-1)(r_k - r_i) = 0. \end{aligned}$$

Thus the last term in the Laplace expansion is the unique term of maximal degree $\sum_{j=1}^{n}(j-1)(r_j - j)$. □

Remark 5.9. Note that $\lambda^{(i+1)} = (1, \lambda_1^{(i)} x^{i-1}, \ldots, \lambda_{n-1}^{(i)} x^{i-n+1})$, so $\lambda^{(i+1)}$ can be computed from $\lambda^{(i)}$ in $O(n + i)$ field operations.

Example 5.10. Although Theorem 5.4 is valid for arbitrary fields, its application to finite fields is limited. The problem is that $C = K^n$ is possible. Let $K = \mathbb{F}_2$, and let b_1, b_2, b_3 be primitive elements of $\mathbb{F}_{2^2}$, $\mathbb{F}_{2^3}$, $\mathbb{F}_{2^5}$, respectively; set $a_1 := b_1 + b_3$ and $a_2 := b_2 + b_3$. Then $A := K[a_1, a_2] = \mathbb{F}_{2^{30}}$, but the elements $0, a_1, a_2, a_1 + a_2$ all lie in proper subfields.

Theorem 5.4 is the basis for the primary decomposition algorithm. If K is an infinite perfect field, it asserts that if A is decomposable, then we will always find an element yielding a decomposition; and if A is not decomposable, then we will always find an element which proves this fact. This is no longer true over non-perfect fields, since there exist algebraic field extensions which are not primitive. This problem was first successfully overcome by Steel [Ste05], by moving the inseparable parts of $K[\underline{x}]/I$ into the field K. We follow this approach, and again formulate and prove the results in terms of algebras.

For an algebraic extension F/K let F_{s}/K be the maximal separable subfield of F/K and F_{i}/K the maximal purely inseparable subfield of F/K. Let $[F : K]_{\mathrm{s}} := [F_{\mathrm{s}} : K]$ be the *separable degree* of F/K, and $[F : K]_{\mathrm{i}} := [F : K]/[F : K]_{\mathrm{s}}$ the *inseparable degree* of F/K. Recall that if N/K is a normal field extension then $N = N_{\mathrm{s}} \cdot N_{\mathrm{i}}$ (see for example [Jac89, Theorem 8.19]); in particular, $[N : K]_{\mathrm{i}} = [N_{\mathrm{i}} : K]$.

Proposition 5.11. *Let A/K be a field generated by $a_1, \ldots, a_n$, and let N/K be the normal closure of A/K. Let $\mu_i \in K[T]$ be the minimal polynomial of a_i, and let L/K be minimal with $\mathrm{sep}(\mu_i) \in L[T]$ for all i. Then $L = N_{\mathrm{i}}$.*

Proof. Set $\sigma_i := \mathrm{sep}(\mu_i)$. Since every root of μ_i in a splitting field has the same multiplicity, there exist $\ell_i \in \mathbb{N}$ with $\mu_i = \sigma_i^{p^{\ell_i}}$. It follows that L/K is purely inseparable, since it is generated by the coefficients of the μ_i, so $L \subseteq N_{\mathrm{i}}$; in particular, $[L : K] \leq [N : K]_{\mathrm{i}}$. On the other hand, N/L is the splitting field of the separable polynomials $\sigma_1, \ldots, \sigma_n$, hence separable; so $[N : L] \leq [N : K]_{\mathrm{s}}$. Thus $[N : K] = [N : L][L : K] \leq [N : K]_s [N : K]_{\mathrm{i}} = [N : K]$, so $[L : K] = [N : K]_{\mathrm{i}} = [N_{\mathrm{i}} : K]$. This proves $L = N_{\mathrm{i}}$. □

Theorem 5.12. *Let A be a finite-dimensional commutative K-algebra, generated by elements $a_1, \ldots, a_n$; let $\mu_i \in K[T]$ be the minimal polynomial of a_i. Let E/K be a purely inseparable extension such that $\operatorname{sep}(\mu_i) \in E[T]$. Set $B := A \otimes_K E / \operatorname{rad}(A \otimes_K E)$, and let $\varepsilon \colon A \to B \colon a \mapsto a \otimes 1 + \operatorname{rad}(A \otimes_K E)$.*

1. *Write $A/\operatorname{rad}(A) = F_1 \oplus \cdots \oplus F_k$ with finite field extensions F_i/K. Let S_i/K be the maximal separable subfield of F_i/K. Then $B \cong S_1 \cdot E \oplus \cdots \oplus S_k \cdot E$, where $S_i \cdot E$ is the compositum of S_i and E. In particular, B/E is separable. Furthermore, ε factors over $\widetilde{\varepsilon} \colon A/\operatorname{rad}(A) \to B \colon a + \operatorname{rad}(A) \mapsto a \otimes 1 + \operatorname{rad}(A \otimes_K E)$, and $\widetilde{\varepsilon}$ is an embedding.*

2. *Let $a \in A$; let $\mu \in K[T]$ be the minimal polynomial of a over K, and let $\widetilde{\mu} \in E[T]$ be the minimal polynomial of $\varepsilon(a)$ over E. Then $\widetilde{\mu} = \operatorname{sep}(\mu)$. In particular, μ is the power of an irreducible polynomial if and only if $\widetilde{\mu}$ is irreducible, and μ has at least two coprime factors if and only if $\widetilde{\mu}$ has at least two coprime factors.*

Proof. We prove part 1. Since $A \otimes_K E/\operatorname{rad}(A \otimes_K E) \cong A/\operatorname{rad}(A) \otimes_K E/\operatorname{rad}(A/\operatorname{rad}(A) \otimes_K E)$, we may assume that A is semi-simple. If K is perfect, then $E = K$, and the results are trivial. So assume in the following that K is a field of positive characteristic p. Assume first $k = 1$, so $A = F_1$ is a field. Since B is semi-simple, it is a direct sum of fields. But every element of $A \otimes_K E$ is either invertible or nilpotent, see for example [Jac89, Theorem 8.46], so B has no non-trivial zero-divisors, which shows that B is a field. The map $A \otimes_K E \to A \cdot E \colon a \otimes \ell \mapsto a \cdot \ell$ is clearly surjective and factors over the radical, so it induces an isomorphism. It remains to prove that $A \cdot E = S_1 \cdot E$. Let N/K be the normal closure of A/K, and let N_{i}/K be the maximal purely inseparable subfield of N/K. Since $N_{\mathrm{i}} \leq A \cdot N_{\mathrm{i}} \leq N$ we see $[N_{\mathrm{i}} : K] \leq [A \cdot N_{\mathrm{i}} : K]_{\mathrm{i}} \leq [N : K]_{\mathrm{i}} = [N_{\mathrm{i}} : K]$, so $[A \cdot N_{\mathrm{i}} : K]_{\mathrm{i}} = [N_{\mathrm{i}} : K]$. On the other hand, S_1 is the maximal separable subfield of $A \cdot N_{\mathrm{i}} : K$, so $[A \cdot N_{\mathrm{i}} : K]_s = [S_1 : K]$. Since S_1 and N_{i} are linearly disjoint, $[S_1 \cdot N_{\mathrm{i}} : K] = [S_1 : K][N_{\mathrm{i}} : K] = [A \cdot N_{\mathrm{i}} : K]$, hence $A \cdot N_{\mathrm{i}} = S_1 \cdot N_{\mathrm{i}}$. But E/K is an extension of N_{i}/K by Proposition 5.11, thus $A \cdot E = (A \cdot N_{\mathrm{i}}) \cdot E = (S_1 \cdot N_{\mathrm{i}}) \cdot E = S_1 \cdot E$. This completes the case $k = 1$.

Now let k be arbitrary. Let $\pi_j \colon A \to F_j$ be the jth projection map. Then F_j is generated by $\pi_j(a_1), \ldots, \pi_j(a_n)$, and the minimal polynomial of $\pi_j(a_i)$ divides μ_i. Thus $F_j \otimes_K E/\operatorname{rad}(F_j \otimes_K E) \cong S_j \cdot L$ follows by the above. But $\operatorname{rad}(A \otimes_K E) = \operatorname{rad}(F_1 \otimes_K E \oplus \cdots \oplus F_k \otimes_K E) = \operatorname{rad}(F_1 \otimes_K E) \oplus \cdots \oplus \operatorname{rad}(F_k \otimes_K E)$, which completes the proof of part 1.

We now prove part 2. Let $\overline{\varepsilon} \colon A \to A/\operatorname{rad}(A)$ be the natural epimorphism. Then it is easy to see that the minimal polynomial of $\overline{\varepsilon}(a)$ is the square-free part of μ, so we may assume that A is semi-simple. Again, if K is perfect, then the statements are obvious, so we may assume that K is a field of positive characteristic p. Assume first that A is a field. Set $\sigma := \operatorname{sep}(\mu) \in L[T]$. As in the proof of Proposition 5.11, there exists $\ell \in \mathbb{N}$ such that $\mu = \sigma^{p^\ell}$. Since $0 = \mu(a) = (\sigma(a))^{p^\ell}$ we see $\sigma(\varepsilon(a)) = \varepsilon(\sigma(a)) = 0$, so $\widetilde{\mu} | \sigma$. But σ is irreducible; for if $\sigma = \sigma_1 \cdot \sigma_2$ then $\mu = \sigma_1^{p^\ell} \cdot \sigma_2^{p^\ell}$. Hence $\sigma = \widetilde{\mu}$.

In the general case, write $A = F_1 \oplus \cdots \oplus F_k$ for field extensions F_j/K and $a = a_1 + \cdots + a_k$ with $a_j \in F_j$. Then $\mu = \operatorname{lcm}(\mu_1, \ldots, \mu_k)$ and $\widetilde{\mu} = \operatorname{lcm}(\widetilde{\mu}_1, \ldots, \widetilde{\mu}_k)$, where μ_i is the minimal polynomial of a_i over K and $\widetilde{\mu}_i$ is the minimal polynomial of $\varepsilon(a_i)$ over L. But $\widetilde{\mu}_i = \operatorname{sep}(\mu_i)$ by the field case, hence $\widetilde{\mu} = \operatorname{sep}(\mu)$, which finishes the proof. □

5.2 Algorithms

Our first primary decomposition algorithm is split into two smaller algorithms. The first algorithm, IsPrimary, decides whether a zero-dimensional ideal is primary; if this is not the case, then it finds an element a which can be used to split the ideal, using Proposition 5.1. The second algorithm, PrimaryDecompositionInfinite, is the main algorithm. It uses IsPrimary to find an element which splits the ideal and then handles the smaller ideals recursively.

Algorithm 5.13 (IsPrimary)**.**
Input: The Gröbner basis of a zero-dimensional ideal $I \trianglelefteq K[\underline{x}]$ for an infinite field K; if $\operatorname{char}(K) > 0$, we assume $K = \mathbb{F}_q(t_1, \ldots, t_m)$ for a prime power q and indeterminates $t_1, \ldots, t_m$.
Output: TRUE if I is a primary ideal. Otherwise FALSE, together with the representation matrix F of an element in $K[\underline{x}]/I$ whose minimal polynomial has at least two coprime factors.

1. Let M be the FGLM data of $A := K[\underline{x}]/I$ and $d := \dim_K(K[\underline{x}]/I)$. Compute the minimal polynomials μ_i of M_i. If some μ_i has at least two coprime factors, return FALSE and M_i. If some μ_i has degree d and is the power of an irreducible polynomial, then return TRUE.

2. Compute the separable parts $\sigma_i := \operatorname{sep}(\mu_i) \in L[T]$, where $L = K$ if K is perfect and $L = \mathbb{F}_q(\sqrt[r]{t_1}, \dots, \sqrt[r]{t_m})$ for some p-power r if $K = \mathbb{F}_q(t_1, \dots, t_m)$.

3. Compute an L-basis C for the ideal $J = \langle \sigma_1(x_1 \otimes 1), \dots, \sigma_n(x_n \otimes 1)\rangle / I \otimes_K L \trianglelefteq A \otimes_K L$ using IdealBasis.

4. Let $H :=$ IdealBasisToGroebner$(I \otimes_K L, C)$, and let N be the FGLM data of H. Set $d := \dim_L(L[\underline{x}]/\langle H\rangle)$. If $\deg(\sigma_j) = d$ for some j, return TRUE. Otherwise, set $i := 2$.

5. Let $\lambda := \lambda^{(i)}$, with $\lambda^{(i)}$ as in Proposition 5.6 if $\operatorname{char}(K) = 0$ and as in Proposition 5.8 if $\operatorname{char}(K) > 0$; set $F := \lambda_1 N_1 + \cdots + \lambda_n N_n$. Compute the minimal polynomial $\mu \in L[T]$ of F.

6. If μ is the power of an irreducible polynomial and $\deg(\mu) = d$, then return TRUE. If μ has two coprime factors, then return FALSE and $\lambda_1 M_1 + \cdots + \lambda_n M_n$. Otherwise set $i := i + 1$ and go to Step 5.

If K is not finite, then the cost of factoring a polynomial cannot be measured in field operations. We therefore list the number of required factorizations separately in the complexity analysis.

Proposition 5.14. *Algorithm 5.13 is correct. It requires $O((n + d)d^4)$ field operations and at most $n + (d - 1)^2$ factorizations of univariate polynomials of degree at most d. If $K = \mathbb{F}_q(t_1, \dots, t_m)$, then it additionally requires at most nd computations of pth roots of elements in K.*

Proof. The algorithm terminates by Corollary 5.5. We show that the output is correct. This is clear if the algorithm returns in Step 1. In Steps 2–4 we compute a Gröbner basis for the kernel of $L[\underline{x}] \to B := A \otimes_K L/\operatorname{rad}(A \otimes_K L)$ (see Algorithm 4.8, Proposition 4.10 and Theorem 5.12). Note that all σ_i are irreducible, since the algorithm did not return in Step 1. Thus if σ_i has degree $d = \dim_L(B)$, then B is a field, hence $A/\operatorname{rad}(A)$ is a field, so I is primary. Hence the output is correct if the algorithm returns in Step 4. The correctness of the output in Step 6 follows by Theorem 5.12.

The computation of the FGLM data and the minimal polynomials in Step 1 can be done using $O(nd^3)$ field operations. The cost for Steps 2 and 3 is $O(nd^3)$ field operations, plus nd computations of pth roots if $\operatorname{char}(K) > 0$, see the proof of Proposition 4.10. Step 4 costs $O(nd^3)$ field operations. The computation of $\lambda^{(i)}$ costs $O(n+i)$ field operations, by Remarks 5.7 and 5.9; the sum can be computed using $O(nd^2)$ field operations and the minimal polynomial in $O(d^3)$ field operations. Note that $i \leq d^2$, so every iteration of Step 5 requires $O(nd^2 + d^3)$ field operations. Since Step 5 is executed at most d^2 times, the total cost for Step 5 is $O((n + d)d^4)$. Finally, there are at most $(d - 1)^2$ factorizations. The matrix sum and product are both only computed once in the algorithm, costing $O(nd^2)$ and $O(d^{\omega+1})$ field operations, respectively. □

Remark 5.15. Steps 2–4 are not necessary for the correctness of the algorithm; in fact, the algorithm can be replaced by the following simpler version.

1. Let M be the FGLM data of $A := K[\underline{x}]/I$ and $d := \dim_K(K[\underline{x}]/I)$. Set $i := 1$.

2. Let $\lambda := \lambda^{(i)}$, with $\lambda^{(i)}$ as in Proposition 5.6 if $\operatorname{char}(K) = 0$ and as in Proposition 5.8 if $\operatorname{char}(K) > 0$; set $F := \lambda_1 M_1 + \cdots + \lambda_n M_n$. Compute the minimal polynomial $\mu \in L[T]$ of F.

3. If μ is reducible and has at least two coprime factors, return FALSE and F. If $\mu = d$ has degree d and is the power of an irreducible polynomial, return TRUE. Otherwise set $i := i+1$. If $i > (d-1)^2+1$, then return TRUE. Otherwise go to Step 2.

The algorithm clearly terminates. Furthermore, if I is not primary, then at least one F out of $(d-1)^2+1$ must have a minimal polynomial with two coprime factors, by Corollary 5.5 and Theorem 5.12. This shows that the algorithm is correct. It is easy to see that it has a runtime complexity of $O((n+d)d^4)$ field operations and $(d - 1)^2 + 1$ factorizations, so it has a better runtime complexity than Algorithm 5.13.

However, Algorithm 5.13 has two practical advantages. First, it only has to check at most $(d'-1)^2+1$ elements in the loop, where $d' = \dim L[\underline{x}]/\sqrt{L \otimes_K I}$; this may be considerably smaller than d. Secondly, if I is primary, then $L[\underline{x}]/\sqrt{L \otimes I}$ always has a primitive element, which may allow the algorithm to terminate early. Such an element need not exist for $K[\underline{x}]/I$. For example, if $I = \langle x_1^\ell, x_2^\ell \rangle \trianglelefteq \mathbb{Q}[x_1, x_2]$, then I is primary and $d = 2\ell$, but every element $\lambda_1 x_1 + \lambda_2 x_2$ has minimal polynomial of degree at most $2\ell - 1$. Thus the variant presented here has to compute and factorize $(d-1)^2+1$ minimal polynomials to conclude that I is primary, whereas Algorithm 5.13 establishes this fact after three minimal polynomials.

Algorithm 5.16 (PrimaryDecompositionInfinite)**.**
Input: The Gröbner basis of a zero-dimensional ideal $I \trianglelefteq K[\underline{x}]$ for an infinite field K; if $\mathrm{char}(K) > 0$, we assume $K = \mathbb{F}_q(t_1, \dots, t_m)$ for a prime power q and indeterminates $t_1, \dots, t_m$.
Output: The primary decomposition of I, that is, a list of ideals $Q_1, \dots, Q_k$ given by Gröbner bases, such that each Q_i is a primary ideal, $\sqrt{Q_i} \neq \sqrt{Q_j}$ for $i \neq j$, and $\bigcap_{i=1}^k Q_i = I$.

1. Call IsPrimary with input I. If I is primary, return I. Otherwise, let F be the representation matrix of the element which proves that I is not primary. Let μ be its minimal polynomial with prime factorization $\mu = \mu_1^{e_1} \cdots \mu_r^{e_r}$.

2. Compute $F_i := \mu_i(F)^{e_i}$ and $W_i := \ker(F_i) \leq K^{1 \times d}$ for $1 \leq i \leq r$.

3. Compute $J_i :=$ IdealBasisToGroebner$(I, \bigoplus_{j \neq i} W_j)$ for $1 \leq i \leq r$.

4. Return $\bigcup_{i=1}^r$ PrimaryDecompositionInfinite(J_i).

Theorem 5.17. *Algorithm* 5.16 *is correct. It requires $O((n+d)d^5)$ field operations and $(n+(d-1)^2+1)d$ factorizations of polynomials of degree at most d; if $K \cong \mathbb{F}_q(t_1, \dots, t_m)$, then it also requires at most nd^2 computations of pth roots of elements in K.*

Proof. Let $I = \bigcap_{i=1}^k Q_i$ be the primary decomposition of I. Then $A = K[\underline{x}]/I \cong K[\underline{x}]/Q_1 \oplus \cdots \oplus K[\underline{x}]/Q_k =: A_1 \oplus \cdots \oplus A_k$ by the Chinese Remainder Theorem. We proceed by induction on k. If $k = 1$ then I is primary, which is detected in Step 1. If $k > 1$ then I is not primary, and Algorithm 5.13 yields $f \in A$ such that the minimal polynomial μ has at least two coprime factors. Let $\mu = \mu_1^{e_1} \cdots \mu_\ell^{e_\ell}$ be the prime factorization of μ. By Proposition 5.1 we can write $A = \ker(\mu_1(f)^{e_1}) \oplus \cdots \oplus \ker(\mu_\ell(f)^{e_\ell})$, and after a renumbering we can assume $W_1 := \ker(\mu_1(f)^{e_1}) = A_1 \oplus \cdots \oplus A_{i_1}$, $W_2 := \ker(\mu_2(f)^{e_2}) = A_{i_1+1} \oplus \cdots \oplus A_{i_2}$, etc. Now $W_j \cong A / \bigoplus_{\ell \neq j} W_\ell$, so the ideal J_j computed in Step 4 satisfies $K[\underline{x}]/J_j \cong W_j = A_{i_{j-1}+1} \oplus \cdots \oplus A_{i_j}$; thus $J_j = Q_{i_{j-1}+1} \cap \cdots \cap Q_{i_j}$, and the correctness follows by induction.

The complexity of Step 1 is given in Proposition 5.14. The matrix powers $I_d, F, F^2, \dots, F^{\max(e_1, \dots, e_k)}$ in Step 2 can be computed in $O(d \cdot d^\omega)$ field operations. Every F_i is the sum of $e_i \deg(\mu_i) + 1$ scaled powers of F, so the F_i can be computed in at most $O(d^3)$ field operations. Each kernel can be computed in $O(d^\omega)$ field operations. Since $k \leq d$, Step 2 has a total cost of $O(d^{\omega+1})$ field operations. Every J_i can be computed in $O(nd^\omega)$ field operations, so Step 3 has a total cost of $O(nd^{\omega+1})$ field operations. Thus Steps 1–3 have a total cost of $O((n+d)d^4)$ field operations and $n+(d-1)^2+1$ factorizations of polynomials of degree at most d, plus nd computations of pth roots of elements in K if $K \cong \mathbb{F}_q(t_1, \dots, t_m)$. To count the number of times that PrimaryDecompositionInfinite is called, consider the recursion tree. The leaves correspond to the primary components of I. Since there are at most d primary components, the tree has at most d leaves and hence at most $2d-1$ vertices. Thus PrimaryDecompositionInfinite is called at most $2d-1$ times, which proves the complexity result. □

Remark 5.18.

1. Using the variant of IsPrimary described in Remark 5.15, the algorithm has a complexity of $O((n+d)d^5)$ field operations and at most $(n+(d-1)^2+1)d$ factorizations.

2. Slightly better complexity results can be achieved by using Remark 3.14 to compute the minimal polynomial. However, this leads to a worse complexity if classical matrix multiplication (that is, $\omega = 3$) is used, and is slower in practice.

6 Primary decomposition for arbitrary fields

Algorithm 5.16 uses linear combinations of the generators of the algebra A to find a splitting element. We have seen in Example 5.10 that this approach may fail if the field is finite. An alternative approach is to use minimal polynomials of arbitrary elements of A. This is used for example in [EHV92] and [Mon02]; note however that [Mon02] uses the characteristic polynomial instead of the minimal polynomial. In both algorithms, the elements are chosen at random. Furthermore, [EHV92] assumes that K is perfect, and [Mon02] assumes $\mathrm{char}(K) = 0$.

In this section we show that it suffices to consider the elements of an arbitrary basis of A. Furthermore, this approach is valid for arbitrary fields K.

Proposition 6.1. *Let $I \trianglelefteq K[\underline{x}]$ be a zero-dimensional ideal; set $A := K[\underline{x}]/I$, and let B be a basis of A. If I is not primary, then μ_b has two coprime factors for at least one $b \in B$.*

Proof. By Theorem 5.12 we may assume that A is semi-simple and A/K is separable. Suppose μ_b is irreducible for all $b \in B$. Then every $b \in B$ lies in a twisted diagonal of A. Write $A = F_1 \oplus \cdots \oplus F_k$, where F_i/K are separable field extensions; define

$$\varphi\colon A \to K\colon a_1 + \cdots + a_k \mapsto (k-1)\,\mathrm{tr}_{F_1/K}(a_1) - \mathrm{tr}_{F_2/K}(a_2) - \cdots - \mathrm{tr}_{F_k/K}(a_k),$$

where $a_i \in F_i$ and $\mathrm{tr}_{F_i/K}$ is the trace of F_i/K. Since F_i/K is separable, $\mathrm{tr}_{F_i/K} \neq 0$, see for example [Jac89, Section 10.5, Lemma]; thus $\varphi \neq 0$, so $\ker\varphi$ is a proper subspace of A. But every twisted diagonal is a subspace of $\ker\varphi$. Hence B is contained in $\ker\varphi$, a contradiction. □

Remark 6.2. An equivalent formulation of Proposition 6.1 is: I is primary if and only if μ_b is a prime power for all $b \in B$. It is natural to assume that the following statement is also true: "I is prime if and only if μ_b is prime for all $b \in B$." But this statement is *false.* To see this, let $K = \mathbb{F}_2(t)$, and let $I := \langle x_1^2 + t^2 + t, x_2^2 + t\rangle \trianglelefteq K[x_1, x_2]$. Set $a_i := x_i + I \in K[x_1, x_2]/I =: A$; then $B := (1, a_1, a_2, a_1a_2)$ is a basis of A, and μ_b is irreducible for all $b \in B$. However, the minimal polynomial of $a_1 + a_2$ is $(T + t)^2$, which shows that I is *not* prime.

Algorithm 6.3 (SplitAlgebra).
Input: $A = \mathrm{Alg}(M, e)$ and $a \in A$.
Output: A list $(M_1, B_1), \ldots, (M_r, B_r)$ satisfying the following properties, where $A_i := \mathrm{Alg}(M_i)$: (1) $A = A_1 \oplus \cdots \oplus A_r$; (2) $\mathrm{B}(A_i) = B_i$; and (3) μ_{a_i} is the power of an irreducible polynomial, where $a = a_1 + \cdots + a_r$ with $a_i \in A_i$.

1. Compute the representation matrix F of a.
2. Compute the minimal polynomial μ of F and its factorization $\mu = \mu_1^{e_1} \cdots \mu_r^{e_r}$.
3. If $r = 1$, then return $\mathrm{Alg}(M, e)$. Otherwise, compute the kernel W_i of the representation matrix of $\mu_i^{e_i}(a)$, for $i = 1, \ldots, r$.
4. Return the output of MatphiDirectSum$(\mathrm{Alg}(M, e), \{W_1, \ldots, W_k\})$.

Proposition 6.4. *Algorithm 6.3 is correct. If μ_a is the power of an irreducible polynomial, then it requires $O(d^3)$ field operations and a factorization of a polynomial of degree at most d. If μ_a has $r > 1$ coprime factors, then it requires $O((n+r)d^3)$ field operations and a factorization of a polynomial of degree at most d.*

Proof. The correctness follows immediately by Proposition 5.1.

By Proposition 3.11, Step 1 has a cost of $O(d^3)$, and by Proposition 3.13, Step 2 has a cost of $O(d^3)$ field operations and one factorization. Note that $a^i = e \cdot F^i$, so $(e, a, \ldots, a^{d-1})$ can be computed by d vector-matrix multiplications, for a total cost of $O(d^3)$ field operations. Let $\mu_i^{e_i} = T^\ell + c_{\ell-1}T^{\ell-1} + \cdots + c_0$; then $\mu_i^{e_i}(a) = a^\ell + c_{\ell-1}a^{\ell-1} + \cdots + c_0 e$ can be computed from $(e, a, \ldots, a^{d-1})$ in $O(d^2)$ field operations. The cost for each representation matrix is $O(d^3)$, and for each kernel $O(d^\omega)$. Thus Step 3 has a total cost of $O(rd^3)$ field operations. Step 4 is $O(nd^3)$ by Proposition 3.23. □

Algorithm 6.5 (PrimaryDecomposition).
Input: The Gröbner basis of a zero-dimensional ideal $I \trianglelefteq K[\underline{x}]$.
Output: The primary decomposition of I, that is, a list of ideals $Q_1, \ldots, Q_k$ given by Gröbner bases, such that each Q_i is a primary ideal, $\sqrt{Q_i} \neq \sqrt{Q_j}$ for $i \neq j$, and $\bigcap_{i=1}^{k} Q_i = I$.

1. Let M be the FGLM data of I and $B = \mathrm{B}(I)$. Set $\mathcal{A} := \{(M, B)\}$ and $i := 1$.
2. Set $\mathcal{A}' := \emptyset$. For every pair (M', B') in $\mathcal{A}$: If $B_i \notin B'$, then add (M', B') to A'. Otherwise, add the output of SplitAlgebra$(\mathrm{Alg}(M'), e_\ell)$ to $\mathcal{A}'$, where ℓ is the index of B_i in B'.
3. Set $\mathcal{A} := \mathcal{A}'$ and set $i := i + 1$. If $i \leq d$, then go to Step 2.
4. Call FGLMDataToGroebner for every element in $\mathcal{A}$ and return the output.

Theorem 6.6. *Algorithm* 6.5 *is correct. It requires $O(nd^4)$ field operations and d factorizations of univariate polynomials over K of degree at most d.*

Proof. The correctness follows by Propositions 6.1 and 6.4.

Step 1 requires $O(nd^3)$ field operations. We analyze Step 2. Let $d_{M',i} := \dim(\mathrm{Alg}(M'))$ in iteration i, and let $r_{M',j}$ be the number of elements returned by SplitAlgebra$(\mathrm{Alg}(M'), e_\ell)$. The call to SplitAlgebra costs $O(d^3_{M',i})$ if $r_{M',i} = 1$, and $O((n + r_{M',i})d^3_{M',i})$ otherwise. The total cost for all M' with $r_{M',i} = 1$ is $O(d^3)$ in every iteration. Since the loop has d iterations, this amounts to a total cost of $O(d^4)$. Now consider all occurrences where $r_{M',i} > 1$ during the run of the algorithm. Every occurrence splits the algebra $\mathrm{Alg}(M')$ into $r_{M',i}$ subalgebras of smaller dimension; arrange all these algebras in a tree. The leaves are the irreducible algebras, so there are $k \leq d$ leaves. It is easy to see that such a tree has at most $2k - 1$ nodes in total, and at most $k - 1$ internal nodes. Since a split corresponds to an internal node, there are at most $k - 1 \leq d - 1$ splits. Furthermore, $\sum_{M',i} r_{M',i} \leq 2d - 2$, as it is the number of non-root nodes. Since $O((n + r_{M',i})d^3_{M',i}) \subseteq O((n + r_{M',i})d^3)$, this shows that the total cost for calls to SplitAlgebra with $r_{M',i} > 1$ is $O(nd^4)$ field operations. The call to FGLMDataToGroebner is $O(nd^2_{M',d})$, so Step 4 has a total cost of $O(nd^2)$ field operations. To achieve the stated number of factorizations, we must alter the run of the algorithm minimally. Note that the minimal polynomial in SplitAlgebra has degree at most $d_{M',i}$. Multiplying all minimal polynomials of one iteration yields a polynomial of degree at most d, which is then factored; taking greatest common divisors of the factors with the minimal polynomials yields the factorization of the minimal polynomials. □

Remark 6.7.

1. The main differences between Algorithm 5.16 and Algorithm 6.5 are that the former uses linear combinations of the generators and proceeds recursively, while the latter uses elements of a basis and proceeds iteratively. Replacing Algorithm 5.16 by an iterative version would yield an algorithm with complexity $O((n + d)d^4)$ and at most $(d - 1)^2 + 1$ factorizations.
2. Algorithm 6.5 suffers from the problem described in Remark 5.15: if the ideal is primary but no basis element is primitive, then it has to compute and factor d minimal polynomials. Hence although the runtime complexity is superior, it will depend on the given example which algorithm performs better in practice.

7 Examples

In this section, we compare the algorithms developed in this paper with the algorithms implemented in Magma [BCP97], which uses a variant of the algorithm described in [BW93]. As suggested by one of the referees, we also implemented a variant of the GTZ algorithm (Algorithm ZPDF in [GTZ88]) which uses the FGLM algorithm for order changes, and therefore also avoids Gröbner basis computations using Buchberger's or Faugere's algorithm. This algorithm is referenced as GTZ below. Note that this algorithm terminates with high probability only for fields of characteristic zero. For fields of positive characteristic it may never terminate.

The examples were run on a machine with four Intel Xeon E5-4617 processors with 6 cores each, running at 2.9 GHz, and 1 TB of DDR3 RAM.

The first three examples are taken from the symbolic data project [Grä10]. We choose the base field $\mathbb{F}_{65537}$ for these examples.

Example 7.1 (ilias13)**.** $\mathbb{F}_{65537}[\underline{x}] = \mathbb{F}_{65537}[d_2, s_2, D_2, S_2, d_1, s_1, S_1]$, degrevlex order with $d_2 > \cdots > S_1$. The ideal is radical, and the residue class algebra has dimension 468. The primary decomposition has 19 components.

Example 7.2 (Reimer_5a)**.** $\mathbb{F}_{65537}[\underline{x}] = \mathbb{F}_{65537}[u, t, z, y, x]$, degrevlex order with $u > \cdots > x$. The ideal is radical, and the residue class algebra has dimension 720. The primary decomposition has 12 components.

Example 7.3 (Steidel_1)**.** $\mathbb{F}_{65537}[\underline{x}] = \mathbb{F}_{65537}[x, y, z]$, degrevlex order with $x > y > z$. The residue class algebra has dimension 729, and the residue class algebra of the radical has dimension 569. The primary decomposition has 28 components.

The next group of examples comes from the L$_3$-U$_3$-quotient algorithm for finitely presented groups [Jam12]. Every finitely presented group $G = \langle a, b \mid w_1, \ldots, w_r\rangle$ defines a trace presentation ideal

$$I(G) \trianglelefteq \mathbb{Z}[x, y_1, \ldots, y_4, z_1, \ldots, z_4, \zeta].$$

The monomial order on the polynomial ring is a weighted degrevlex order with $\deg(x) = 8$, $\deg(y_i) = 2$, $\deg(z_i) = 4$, $\deg(\zeta) = 1$, and $x > y_1^4 > \cdots > y_4^4 > z_1^2 > \cdots > z_4^2 > \zeta^8$.

The ideal $I(G)$ is not zero-dimensional in general, but we will alter the ideal or the polynomial ring to get zero-dimensional ideals (over fields).

Example 7.4. $I := I(G) \otimes_{\mathbb{Z}} \mathbb{Q} \trianglelefteq \mathbb{Q}[x, \ldots, \zeta]$, where $G = \langle a, b \mid (a^5b^2)^2, [a^{-2}, b]^4, ab^{-2}a^2b\rangle$. The ideal is radical, and the residue class algebra has dimension 184. The primary decomposition has 12 components.

Example 7.5. $I := I(G) \otimes_{\mathbb{Z}} \mathbb{Q} \trianglelefteq \mathbb{Q}[x, \ldots, \zeta]$, where $G = \langle a, b \mid a^5, b^5, (ab)^5, [a, b]^5\rangle$. The residue class algebra has dimension 458, and the residue class algebra of the radical has dimension 446. The primary decomposition has 46 components.

Example 7.6. $I := I(G) \otimes_{\mathbb{Z}} \mathbb{Q} \trianglelefteq \mathbb{Q}[x, \ldots, \zeta]$, where $G = \langle a, b \mid a^4, b^4, (ab)^6, [a, b]^6\rangle$. The residue class algebra has dimension 256, and the residue class algebra of the radical has dimension 248. The primary decomposition has 57 components.

Example 7.7. $I := I(G) \otimes_{\mathbb{Z}[y_2]} \mathbb{F}_2(y_2) + \langle y_1 + y_4\rangle \trianglelefteq \mathbb{F}_2(y_2)[x, y_1, y_3, y_4, z_1, \ldots, z_4, \zeta]$, where $G = \langle a, b \mid ab^2a^{-1}b^3, [a, b]^2\rangle$. The ideal is radical, and the residue class algebra has dimension 50.

Example 7.8. $I := I(G) \otimes_{\mathbb{Z}[z_2]} \mathbb{F}_2(z_2) + \langle z_3 + z_4\rangle \trianglelefteq \mathbb{F}_2(z_2)[x, y_1, \ldots, y_4, z_1, z_3, z_4, \zeta]$, where $G = \langle a, b \mid a^4b^4, (ab)^4\rangle$. The residue class algebra has dimension 156, and the residue class algebra of the radical has dimension 16. The primary decomposition has 4 components.

The final example comes from the L$_2$-quotient algorithm for finitely presented groups [Jam14]. Every finitely presented group $G = \langle a, b, c \mid w_1, \ldots, w_r\rangle$ defines a trace presentation ideal

$$I(G) \trianglelefteq \mathbb{Z}[x, y_1, \ldots, y_3, z_1, \ldots, z_3].$$

The monomial order on the polynomial ring is a weighted degrevlex order with $\deg(x) = 3$, $\deg(y_i) = 2$, $\deg(z_i) = 1$, and $x^2 > y_1^3 > \cdots > y_3^3 > z_1^6 > \cdots > z_3^6$.

Example 7.9. $I := I(G) \otimes_{\mathbb{Z}[z_2]} \mathbb{F}_2(z_2) \trianglelefteq \mathbb{F}_2(z_2)[x, y_1, y_2, y_3, z_1, z_3]$, where

$$G = \langle a, b, c \mid a^5bac^{11}, aba^{-1}ba^{-2}c\rangle$$

The ideal is radical, and the residue class algebra has dimension 28. The minimal polynomials of some generators are inseparable, so Algorithm 4.8 must use field extensions.

The time to compute the radical is listed in Table 1, and the time to compute the primary decomposition in Table 2. Several computations did not finish after three hours and were terminated without a result. Note that the first three examples are defined over a finite field, the second three examples are defined over the rationals, and the last three examples are defined over function fields of finite fields. Since Algorithm 5.16 is only applicable to ideals defined over infinite fields, it cannot be used on the first three examples.

It should be noted that MAGMA's algorithms use p-adic and modular techniques to address the problem of coefficient growth for the computations over the rationals. The implementation of our algorithms does not use these techniques at the moment, but it can be expected that this would yield a significant speed-up. The modular approach is described in [IPS11].

Another possibility for improvement is the use of sparse linear algebra methods. At the moment, the algorithms use regular linear algebra methods, and the FGLM data of the examples are relatively dense. For sparse examples, the Gröbner based algorithms seem to perform better than the algorithms developed in this paper.

	Ex. 7.1	Ex. 7.2	Ex. 7.3	Ex. 7.4	Ex. 7.5	Ex. 7.6	Ex. 7.7	Ex. 7.8	Ex. 7.9
MAGMA	192.27	104.72	9.01	3.43	82.06	5.88	> 3h	> 3h	> 3h
Alg. 4.8	1.43	0.93	1.81	18.73	52.74	2.98	1.50	9.94	0.32

Table 1: Time (in seconds) to compute the radical.

	Ex. 7.1	Ex. 7.2	Ex. 7.3	Ex. 7.4	Ex. 7.5	Ex. 7.6	Ex. 7.7	Ex. 7.8	Ex. 7.9
MAGMA	> 3h	> 3h	2760.28	1.70	> 3h	> 3h	6.88	4350.18	3691.44
GTZ	2.35	5.78	> 3h	5867.68	> 3h	> 3h	13.57	> 3h	31.02
Alg. 5.16	n/a	n/a	n/a	1.51	2.93	5.79	14.29	0.06	0.06
Alg. 6.5	2.08	4.76	5.96	5.77	0.98	4.51	0.44	0.06	0.13

Table 2: Time (in seconds) to compute the primary decomposition.

8 Acknowledgments

I was supported by the Alexander von Humboldt Foundation via a Feodor Lynen Research Fellowship. The initial ideas for this paper were developed while I was visiting the MAGMA group at the University of Sydney. I am grateful to Allan Steel for several very helpful discussions. I am also grateful to John Cannon for inviting me to Sydney. I thank Eamonn O'Brien and Gerhard Pfister for helpful comments on an early version of this paper. The referees made very helpful suggestions, which led to a significant improvement of the paper.

References

[AKR05] J. Abbott, M. Kreuzer, and L. Robbiano. Computing zero-dimensional schemes. *J. Symbolic Comput.*, 39(1):31–49, 2005.

[BCP97] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. Computational algebra and number theory (London, 1993).

[BTBQM00] M. A. Borges-Trenard, M. Borges-Quintana, and T. Mora. Computing Gröbner bases by FGLM techniques in a non-commutative setting. *J. Symbolic Comput.*, 30(4):429–449, 2000.

[BW93] Thomas Becker and Volker Weispfenning. *Gröbner bases*, volume 141 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1993. A computational approach to commutative algebra, In cooperation with Heinz Kredel.

[DM96] John D. Dixon and Brian Mortimer. *Permutation groups*, volume 163 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1996.

[EHV92] David Eisenbud, Craig Huneke, and Wolmer Vasconcelos. Direct methods for primary decomposition. *Invent. Math.*, 110(2):207–235, 1992.

[FGLM93] J. C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *J. Symbolic Comput.*, 16(4):329–344, 1993.

[Grä10] Hans-Gert Gräbe. The SymbolicData Project, 2010.

[GTZ88] Patrizia Gianni, Barry Trager, and Gail Zacharias. Gröbner bases and primary decomposition of polynomial ideals. *J. Symbolic Comput.*, 6(2-3):149–167, 1988. Computational aspects of commutative algebra.

[GV85] Ira Gessel and Gérard Viennot. Binomial determinants, paths, and hook length formulae. *Adv. in Math.*, 58(3):300–321, 1985.

[GWW09] Shuhong Gao, Daqing Wan, and Mingsheng Wang. Primary decomposition of zero-dimensional ideals over finite fields. *Math. Comp.*, 78(265):509–521, 2009.

[IPS11] Nazeran Idrees, Gerhard Pfister, and Stefan Steidel. Parallelization of modular algorithms. *J. Symbolic Comput.*, 46(6):672–684, 2011.

[Jac89] Nathan Jacobson. *Basic algebra. II.* W. H. Freeman and Company, New York, second edition, 1989.

[Jam12] Sebastian Jambor. *An* L_3-U_3*-quotient algorithm for finitely presented groups.* PhD thesis, RWTH Aachen University, 2012.

[Jam14] Sebastian Jambor. An L_2-quotient algorithm for finitely presented groups on arbitrarily many generators. *Preprint*, 2014. arxiv:1402.6788.

[Kem02] Gregor Kemper. The calculation of radical ideals in positive characteristic. *J. Symbolic Comput.*, 34(3):229–238, 2002.

[KL91] Teresa Krick and Alessandro Logar. An algorithm for the computation of the radical of an ideal in the ring of polynomials. In *Applied algebra, algebraic algorithms and error-correcting codes (New Orleans, LA, 1991)*, volume 539 of *Lecture Notes in Comput. Sci.*, pages 195–205. Springer, Berlin, 1991.

[Kni95] Philip A. Knight. Fast rectangular matrix multiplication and QR decomposition. *Linear Algebra Appl.*, 221:69–81, 1995.

[KR00] Martin Kreuzer and Lorenzo Robbiano. *Computational commutative algebra. 1.* Springer-Verlag, Berlin, 2000.

[Kre89] Heinz Kredel. Primary ideal decomposition. In *EUROCAL '87 (Leipzig, 1987)*, volume 378 of *Lecture Notes in Comput. Sci.*, pages 270–281. Springer, Berlin, 1989.

[Lak90] Y. N. Lakshman. *On the Complexity of Computing Gröbner bases for Zero Dimensional Polynomial Ideals.* PhD thesis, Rensselaer Polytechnic Institute, 1990.

[MMM93] M. G. Marinari, H. M. Möller, and T. Mora. Gröbner bases of ideals defined by functionals with an application to ideals of projective points. *Appl. Algebra Engrg. Comm. Comput.*, 4(2):103–145, 1993.

[Mon02] Chris Monico. Computing the primary decomposition of zero-dimensional ideals. *J. Symbolic Comput.*, 34(5):451–459, 2002.

[Rou99] Fabrice Rouillier. Solving zero-dimensional systems through the rational univariate representation. *Appl. Algebra Engrg. Comm. Comput.*, 9(5):433–461, 1999.

[Sei74] A. Seidenberg. Constructions in algebra. *Trans. Amer. Math. Soc.*, 197:273–313, 1974.

[Ste05] Allan Steel. Conquering inseparability: primary decomposition and multivariate factorization over algebraic function fields of positive characteristic. *J. Symbolic Comput.*, 40(3):1053–1075, 2005.

[Sto94] Arne Storjohann. *Algorithms for Matrix Canonical Forms.* PhD thesis, Swiss Federal Institute of Technology Zurich, 1994.

[Str69] Volker Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:354–356, 1969.

[vzGG99] Joachim von zur Gathen and Jürgen Gerhard. *Modern computer algebra.* Cambridge University Press, New York, 1999.

[Wil12] Virginia Vassilevska Williams. Multiplying matrices faster than Coppersmith-Winograd [extended abstract]. In *STOC'12—Proceedings of the 2012 ACM Symposium on Theory of Computing*, pages 887–898. ACM, New York, 2012.

Department of Mathematics
The University of Auckland
Private Bag 92019
Auckland
New Zealand
E-mail address: `s.jambor@auckland.ac.nz`