Computing representation matrices for the Frobenius on cohomology groups

Momonari Kudo

Computing representation matrices for the Frobenius on cohomology groups

Momonari Kudo

Department of Mathematical Informatics, The University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan

Abstract

In algebraic geometry, the *Frobenius map* F^* on cohomology groups play an important role in the classification of algebraic varieties over a field of positive characteristic. In particular, representation matrices for F^* give rise to many important invariants such as p-rank and a-number. Several methods for computing representation matrices for F^* have been proposed for specific curves.

In this paper, we present an algorithm to compute representation matrices for F^* of general projective schemes over a perfect field of positive characteristic. We also propose an efficient algorithm specific to complete intersections; it requires to compute only certain coefficients in a power of a multivariate polynomial. Our algorithms shall derive fruitful applications such as computing Hasse-Witt matrices, and enumerating superspecial curves. In particular, the second algorithm provides a useful tool to judge the superspeciality of an algebraic curve, which is a key ingredient to prove main results in Kudo and Harashita (2017a,b, 2020) on the enumeration of superspecial genus-4 curves.

Keywords: Cohomology groups, Frobenius maps, Hasse-Witt matrices

1. Introduction

Let K be a perfect field of positive characteristic p. For a positive integer r, let $\mathbf{P}^r = \mathrm{Proj}(S)$ denote the projective r-space, where $S = K[x_0, \ldots, x_r]$ is the polynomial ring of $r+1$ variables over K. Given a projective scheme $X \subset \mathbf{P}^r$ and $q \in \mathbb{Z}$, we denote by $\mathcal{O}_X$ and $H^q(X, \mathcal{O}_X)$ its structure sheaf and its q-th cohomology group, respectively. Note that in this paper we do not assume, unless otherwise noted, any condition (e.g., smoothness, irreducibility) on X other than projectivity, while a curve means a smooth projective variety of dimension 1. Let F be the absolute Frobenius on X, and let F^* denote the induced Frobenius map on the q-th cohomology group. Computing F^* is significant to classify algebraic varieties over a positive characteristic field. For example, we can check whether a curve is superspecial or ordinary, by computing whether the Frobenius on its first cohomology group is zero or bijective. As another example, a matrix representing F^* enables us to compute several invariants

such as p-rank and a-number, which are also important in the classification of curves. This classification is of interest not only in theory, but also for applications to cryptography using algebraic curves, see e.g., Castryck et al. (2019), where superspecial genus-2 curves are used.

In case of specific (non-singular) curves, there are many previous works for computing F^* on $H^1(X, \mathcal{O}_X)$, e.g., Manin (1961), Yui (1978), Bostan et al. (2003), Komoto et al. (2010) and Harvey and Sutherland (2014) for hyperelliptic curves, González (1997) for Fermat curves, Stöhr and Voloch (1987) for plane projective curves given by an affine equation, Kudo and Harashita (2017a) and Celik et al. (2018) for non-hyperelliptic curves of low genera (see also Tuitman (2014) for computing the Zeta function of a curve over a finite field). For a simple example, when X is an elliptic curve E defined by a cubic form f in $K[x, y, z]$, the Frobenius $F^* : H^1(E, \mathcal{O}_E) \longrightarrow H^1(E, \mathcal{O}_E)$ is determined from the $(xyz)^{p-1}$-coefficient in f^{p-1}, see Hartshorne (1977), Chapter IV for more details. For another example, we consider when X is a hyperelliptic curve of genus g defined by an affine equation $y^2 = f(x)$ with $f(x) \in K[x]$ of degree $2g + 1$. Instead of F^*, one can compute the matrix representing an operator on $H^0(X, \Omega^1_X)$ dual to F^*, where Ω^1_X denotes the sheaf of differential 1-forms on X. The operator on $H^0(X, \Omega^1_X)$ is called the *Cartier operator*. It is shown in Manin (1961) and the proof of Yui (1978), Proposition 2.1 that the Cartier operator is determined from the coefficients c_{ip-j} with $1 \leq i, j \leq g$ in $f^{(p-1)/2} = \sum_{k=0}^{2g+1} c_k x^k$, where p is odd (see also Theorem 1.1 of Stöhr and Voloch (1987) for a more general statement over plane curves). In the proof of this fact, the image of a basis of $H^0(X, \Omega^1_X)$ by the Cartier operator is converted into a K-linear combination of the same basis, by using the relation $y^2 = f(x)$. Based on this fact, several algorithms specific to the Cartier operator over a hyperelliptic curve have been proposed, e.g., Bostan et al. (2003), Komoto et al. (2010) and Harvey and Sutherland (2014).

While several algorithms have been proposed for specific cases as above, the aim of this paper is to present a general-purpose algorithm, which works for *arbitrary* projective schemes X (of dimension ≥ 1). For constructing such an algorithm, it suffices to implement the following:

Step 1. Compute an explicit basis of the cohomology group $H^q(X, \mathcal{O}_X)$.

Step 2. Make a representation matrix of F^* with respect to the computed basis.

For Step 1, we can apply an algorithm proposed in Kudo (2017). In general, there are two main strategies to compute cohomology groups; the polynomial ring-based method, and the exterior algebra-based method. The first (resp. second) method is realized in Eisenbud (1998), Smith (2000), Maruyama (2002), Kudo (2017) (resp. in Decker and Eisenbud (2002), Eisenbud et al. (2003)). In particular, the algorithm in Kudo (2017), Section 3 based on a method from Maruyama (2002) computes an explicit basis of $H^q(X, \mathcal{O}_X)$.

For Step 2, to devise a concrete procedure, there exist difficulties in representing the image of each basis element by F^* as a K-linear combination of the original basis, since the map F^* is not K-linear but *p-linear*. We here

briefly describe these difficulties (we will explain details in Subsection 2.2); The desired representation matrix can not be computed only by linear algebra techniques such as solving a system of linear equations over K. In fact, no algebraic equation on unknown entries is derived directly from the image of each basis element by F^*. Besides that, different from the Cartier operator over hyperelliptic curves, using defining equations for X is not necessarily appropriate to convert the image under F^* of each basis element. This also means that we have to find a *non-trivial algebraic relation* satisfied in $H^q(X, \mathcal{O}_X)$ between the image of each basis and the original basis. Here we list our solution below;

(i) We decompose F^* into two *computable* maps; the first one raises all coordinates of elements in $H^q(X, \mathcal{O}_X)$ to their p-th powers, and the second one corresponds to multiplying a matrix of homogeneous polynomials.

(ii) We give a method to compute a matrix corresponding to the second map in (i). In particular, we prove that such a matrix always exists; it can be computed from a free resolution of S/I.

(iii) We obtain our desired representation by multiplying the matrix computed in (ii) to each basis element, and also by using linear algebra techniques. This corresponds to find non-trivial algebraic relations between elements of the original basis and those of its image.

Our solution (i)-(iii) is also a generalization of conventional approaches to specific projective curves defined by a few homogeneous polynomials; Hartshorne (1977), Chapter IV for elliptic curves defined by one polynomial, and Kudo and Harashita (2017a) for genus-4 curves (resp. Celik et al. (2018) for genus-3 curves) defined by two polynomials. See also Baker (2000), Section 3 for further examples over curves defined by (at most two) concrete polynomials.

Based on the above (i)-(iii), we present two algorithms (Algorithm (I) in Section 3 and Algorithm (II) in Section 4 below) to compute a representation matrix for F^*. Our algorithms are also based on theories of computational algebraic geometry, where Gröbner bases play a key role. The first algorithm (Algorithm (I)) works for arbitrary projective schemes.

Main Result 1. With notation as above, we fix r the dimension of $\mathbf{P}^r$. Given $1 \leq q \leq r-1$, the characteristic p and an projective scheme $X \subset \mathbf{P}^r = \mathrm{Proj}(S)$ with defining homogeneous polynomials $f_1, \ldots, f_m \in S = K[x_0, \ldots, x_r]$, there exists an algorithm (Algorithm (I) in Section 3) to compute the representation matrix for $F^* : H^q(X, \mathcal{O}_X) \longrightarrow H^q(X, \mathcal{O}_X)$. The algorithm (not counting the computation of free resolutions and lifting homomorphisms) terminates in

$$\tilde{O}\left(D^4 + D^2 \cdot \mathsf{P}(p) + D^3 p^r\right). \tag{1.1}$$

arithmetic operations over K, where D is the maximal value of the dimensions of the cohomology groups over $\mathbf{P}^r$ appearing in the computation of $H^q(X, \mathcal{O}_X)$. For each $e \in \mathbb{Z}_{\geq 1}$, we denote by $\mathsf{P}(e)$ the number of arithmetic operations over K for computing the e-th power.

Remark 1.1. Note that Algorithm (I) in Section 3 requires to compute a (minimal) *free resolution*, which can be constructed by successively computing syzygies with Gröbner basis algorithms. However, as in Kudo (2017), we do not count the cost of computing a free resolution for S/I with $I := \langle f_1, \ldots, f_m \rangle_S$, and set D as a complexity parameter for the following reason: First, it is reasonable that the complexity is measured by properties of the algebraic structure which we target (in our case, X, I, S/I, $\mathcal{O}_X$, $H^q(X, \mathcal{O}_X)$ and so on). Second, no precise complexity bound of computing a free resolution (in general case) is known (see Subsection 2.4 for a review on this issue). This comes (partly) from the difficulty on estimating the complexity of Gröbner basis computation for syzygy modules. Furthermore, a minimal free resolution for S/I derives many invariants such as projective dimension, (graded) Betti numbers, regularity, and the invariant D (see Subsection 3.4 for its definition). Thus, once the minimal free resolution is computed, D is appropriate for a complexity parameter of the remaining computation.

By a similar reason to the case of free resolutions, any explicit complexity bound of computing lifting homomorphisms has not been determined yet, see Sub-algorithm $\texttt{LIFT}(\varphi, \varphi')$ of Subsection 3.2 and Remark 3.1 for details.

The second one (Algorithm (II)) is a simplified version of the first one, and is specific to complete intersections. An advantage of Algorithm (II) is that the cost (1.1) is negligible since all the entries of the matrix representing F^* are obtained in the computation of free resolutions and lifting homomorphisms.

Main Result 2. With notation as above, we fix r the dimension of $\mathbf{P}^r$. Let $S = K[x_0, \ldots, x_r]$, and $X = V(f_1, \ldots, f_m)$ a complete intersection embedded in $\mathbf{P}^r$ with an S-regular sequence $(f_1, \ldots, f_m) \in S^m$. Assume $d_{j_1 \ldots j_{m-1}} := \sum_{k=1}^{m-1} \deg(f_{j_k}) \leq r$ for all $1 \leq j_1 < \cdots < j_{m-1} \leq m$ and $\gcd(f_i, f_j) = 1$ in S for $i \neq j$. Given the characteristic p and $(f_1, \ldots, f_m)$, there exists an algorithm (Algorithm (II) in Section 4) to compute the representation matrix for $F^* : H^q(X, \mathcal{O}_X) \longrightarrow H^q(X, \mathcal{O}_X)$ with $q = \dim(X) = r - m$. The complexity of the algorithm is bounded by the cost of computing $(f_1 \cdots f_m)^{p-1}$.

The rest of this paper is organized as follows: Section 2 introduces the notion of the cohomology groups of projective schemes, and the Frobenius map on the cohomology groups, and proves some properties of graded modules and their homomorphisms. We also briefly review known algorithms to compute free resolutions, and describe some properties of the Frobenius functor for the category of modules. In Section 3, we present an explicit algorithm to compute representation matrices for the action of Frobenius over general projective schemes, and estimate its complexity. In Section 4, we give an efficient algorithm specific to complete intersections as a simplified version of the first algorithm proposed in Section 3. Section 5 shows computational examples and experimental results obtained from our implementation over Magma (Bosma et al. (1997), Cannon et al. (2016)). In Section 6, we give some concluding remarks.

2. Preliminaries

This section lists some mathematical facts which will be used in the rest of this paper. The first and second subsections introduce the notion of the cohomology groups of projective schemes, and the Frobenius action to the cohomology groups, respectively. The third subsection shows some properties of graded homomorphisms, which are necessary to construct our main algorithm in Section 3. In the fourth subsection, we also recall the definition of free resolutions, and briefly discuss known algorithms to compute free resolutions, and their complexities. The fifth subsection reviews the Frobenius functor for the category of modules.

Throughout this section, let K be a field. For a positive integer n and variables $x = (x_1, \ldots, x_n)$, we denote by $K[x] = K[x_1, \ldots, x_n]$ the polynomial ring of n variables over K.

2.1. Cohomology groups

Let $X \subset \mathbf{P}^r = \mathrm{Proj}(S)$ be a projective scheme over K, where $S = K[x] = K[x_0, \ldots, x_r]$. In general, the cohomology groups of a sheaf on X are defined by its flabby resolution, but in a view of computational points, the notion of *Čech cohomology* gives a useful tool for computing the cohomology groups $H^q(X, \mathcal{O}_X)$. Here we review the Čech cohomology. Let $\mathcal{F}$ be a (coherent) sheaf on X, and let $\mathcal{U} = \{U_i\}_{i \in I}$ with $I \subset \mathbb{Z}$ be an open covering for X. To simplify the notation, we set $U_{i_0,\ldots,i_q} := U_{i_0} \cap U_{i_1} \cap \cdots \cap U_{i_q}$ for each $(i_0, \ldots, i_q) \in I^{q+1}$. For an integer $q \geq 0$, the *Čech q-cochain* is defined by

$$C^q(\mathcal{U}, \mathcal{F}) := \prod_{(i_0,\ldots,i_q) \in I^{q+1} \text{ with } i_0 < \cdots < i_q} \mathcal{F}(U_{i_1,\ldots,i_q}).$$

We define the q-th differential map $d^{(q)}$ by

$$d^{(q)} \ : \ C^q(\mathcal{U}, \mathcal{F}) \longrightarrow C^{q+1}(\mathcal{U}, \mathcal{F}) \ ; \ (f_{i_0,\ldots,i_q})_{i_0,\ldots,i_q} \mapsto ((d^{(q)} f)_{i_0,\ldots,i_{q+1}})_{i_0,\ldots,i_{q+1}}$$

with

$$(d^{(q)} f)_{i_0,\ldots,i_{q+1}} := \sum_{j=0}^{q+1} (-1)^j f_{i_0,\ldots,\widehat{i_j},\ldots,i_{q+1}},$$

where the hat means to omit i_j. One can verify that $d^{(q+1)} \circ d^{(q)} = 0$, and thus the sequence

$$0 \longrightarrow C^0(\mathcal{U}, \mathcal{F}) \xrightarrow{d^{(0)}} C^1(\mathcal{U}, \mathcal{F}) \xrightarrow{d^{(1)}} C^2(\mathcal{U}, \mathcal{F}) \xrightarrow{d^{(2)}} C^3(\mathcal{U}, \mathcal{F}) \xrightarrow{d^{(3)}} \cdots$$

is a complex. Here, the q-th Čech cohomology group is defined as follows:

$$H^q(\mathcal{U}, \mathcal{F}) := \mathrm{Ker}\left(d^{(q)}\right) / \mathrm{Im}\left(d^{(q-1)}\right).$$

Note that the group $H^q(\mathcal{U}, \mathcal{F})$ depends on one's choice of the covering $\mathcal{U}$ in general. However, for any separable scheme X with an affine open covering $\mathcal{U}$, the group does not depend on such a choice. In other words, we have

$$H^q(X, \mathcal{F}) \cong H^q(\mathcal{U}, \mathcal{F})$$

for any affine open covering $\mathcal{U}$ for X.

For the r-projective space $X = \mathbf{P}^r = \mathrm{Proj}(S)$ with $S = K[x] = K[x_0, \ldots, x_r]$ and a line bundle $\mathcal{F} = \mathcal{O}_X(m)$ with $m \in \mathbb{Z}$, taking $\mathcal{U}$ to be the Zariski open covering, we can compute a basis of $H^q(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m))$. In the following, we collect some basic facts on the cohomology groups $H^q(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m))$. We grade S by taking S_d to be the set of homogeneous polynomials of degree d. For an integer $m \in \mathbb{Z}$, let $S(m)$ denote the m-twist of S defined by $S(m)_t = S_{m+t}$. Let $S(m)_{x_0 \cdots x_r}$ denote the localization of $S(m)$ by the powers of $x_0 \cdots x_r$. We write $x = x_0 \cdots x_r$ and denote simply by x^α the monomial $x_0^{\alpha_0} \cdots x_r^{\alpha_r}$ of total degree $|\alpha| := \sum_{i=0}^r \alpha_i$ for $\alpha = (\alpha_0, \ldots, \alpha_r) \in \mathbb{Z}^{r+1}$. For an integer d, we also denote by $(S(m)_{x_0 \cdots x_r})_d$ the homogeneous part of degree d of the localization $S(m)_{x_0 \cdots x_r}$. In particular, the homogeneous part $(S(m)_{x_0 \cdots x_r})_0$ of degree zero is the vector space over K spanned by the set

$$\left\{ ax^\alpha : a \in K, \text{ and } \alpha \in \mathbb{Z}^{r+1} \text{ with } |\alpha| = m \right\}.$$

We define L_m to be the subspace

$$\left\langle x^\alpha : \alpha \in \mathbb{Z}^{r+1} \text{ with } \alpha_i \geq 0 \text{ for some } 0 \leq i \leq r \text{ and } |\alpha| = m \right\rangle_K$$

of the vector space $(S(m)_{x_0 \cdots x_r})_0$.

Theorem 2.1 (Hartshorne (1977), Theorem 5.1)**.** *With notation as above, we have the following:*

(1) *We have the following isomorphisms of vector spaces over K:*

$$H^0\left(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m)\right) \cong \begin{cases} S_m & \text{if } m \geq 0, \\ 0 & \text{if } m < 0. \end{cases}$$

In particular, for each $m \geq 0$, the set

$$\left\{ x^\alpha : \alpha \in (\mathbb{Z}_{\geq 0})^{r+1} \text{ with } |\alpha| = m \right\}$$

is a basis of the K-vector space $H^0\left(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m)\right)$.

(2) *For $0 < q < r$ and arbitrary m, we have $H^q\left(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m)\right) = 0$.*

(3) *One has the isomorphism*

$$H^r\left(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m)\right) \quad \cong \quad (S(m)_{x_0 \cdots x_r})_0 \ / \ L_m \tag{2.1}$$

of vector spaces over K. Thus for each $m < 0$, the set

$$\left\{ x^\alpha : \alpha \in (\mathbb{Z}_{<0})^{r+1} \text{ with } |\alpha| = m \right\}$$

gives rise to a basis of $H^r\left(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m)\right)$ via the above isomorphism (2.1)*.*

Corollary 2.2 (Hartshorne (1977), Theorem 5.1)**.** *One has the following:*

$$\begin{aligned}\dim_K H^0\left(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m)\right) &= \begin{cases} \binom{m+r}{r} & \text{if } m \geq 0, \\ 0 & \text{if } m < 0. \end{cases} \\ \dim_K H^r\left(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m)\right) &= \begin{cases} \binom{-m-1}{r} & \text{if } m \leq -r-1, \\ 0 & \text{if } m > -r-1. \end{cases}\end{aligned}$$

For $q \notin \{0, r\}$ and $m \in \mathbb{Z}$, one has $\dim_K H^q\left(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(m)\right) = 0$.

2.2. Frobenius action on cohomology groups

For a scheme X over a perfect field K of positive characteristic p, the *absolute Frobenius* on X is a morphism $F : X \to X$ with the identity map on X and $a \mapsto a^p$ on sections. The absolute Frobenius F induces the action on the q-th cohomology group $H^q(X, \mathcal{O}_X)$. We write F^* for the induced action, say

$$F^* : H^q(X, \mathcal{O}_X) \longrightarrow H^q(X, \mathcal{O}_X).$$

The induced map F^* is a p-linear map, i.e., we have $F^*(af) = a^p F^*(f)$ for all $a \in K$ and all $f \in H^q(X, \mathcal{O}_X)$. As we mentioned in Section 1, a representation matrix for F^* can define a number of invariants, which are important in the classification of X.

Our goal is to compute a matrix representing F^* algorithmically. For this, recall from Section 1 that the first step (Step 1) is to compute a basis $\mathcal{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_g\}$ of $H^q(X, \mathcal{O}_X)$ explicitly, with the algorithm proposed in Kudo (2017), Section 3 from a method by Maruyama (2002). Each basis element $\mathbf{b}_i$ computed by the algorithm is the equivalence class of a vector of the form

$$\sum_{j=1}^{t} \left(\sum_{\alpha \in (\mathbb{Z}_{<0})^{r+1},\ |\alpha| = -d_j} b_{i,j,\alpha} x^\alpha \right) \mathbf{e}_j \tag{2.2}$$

for some $b_{i,j,\alpha} \in K$, where $x = x_0 \cdots x_r$ is a multiple of variables $x_0, \ldots, x_r$. We also denote by $\mathbf{e}_j$ the vector with 1 in the j-th coordinate and 0's elsewhere.

The second step (Step 2 in Section 1) is to make the matrix representing F^* with respect to $\mathcal{B}$. Since the absolute Frobenius F sends a section f to f^p, the Frobenius F^* acts on $H^q(X, \mathcal{O}_X)$ by raising all coordinates of (2.2) to their p-th powers. From this, constructing the desired matrix representing F^* with respect to $\mathcal{B}$ means to compute elements $y_{i,k} \in K$ such that

$$\sum_{j=1}^{t} \left(\sum_{|\alpha| = -d_j} b_{i,j,\alpha}^p x^{p\alpha} \right) \mathbf{e}_j = \sum_{k=1}^{g} y_{i,k} \sum_{j=1}^{t} \left(\sum_{|\alpha| = -d_j} b_{k,j,\alpha} x^\alpha \right) \mathbf{e}_j \tag{2.3}$$

holds in $H^q(X, \mathcal{O}_X)$ for each $1 \leq i \leq g$, where α runs through elements in $(\mathbb{Z}_{<0})^{r+1}$. However, finding such $y_{i,k}$ is non-trivial for the following reasons:

- While the representation matrix for a linear map is computed by solving a system of linear equations, no algebraic equation on $y_{i,k}$ is obtained from (2.3), admitting that one compares the j-th coordinates for each $1 \leq j \leq t$. Indeed, the degree of all monomials in the j-th coordinate of the left hand side is $-pd_j$, while that of the right hand side is $-d_j$.

- Another considerable approach is using defining equations $f_i = 0$ with $1 \leq i \leq m$ for X to convert the left hand side into the right hand side of (2.3). However, a monomial dividing some $x^{p\alpha}$ does not necessarily appear in f_i's in general. Even if such a monomial x^β in f_i exists, replacing $x^{p\alpha}$ by $(-(f_i - x^\beta))x^{p\alpha-\beta}$ does not change the degree of monomials since f_i is homogeneous of degree $|\beta|$.

Thus, we need an alternative approach to the above two for finding the algebraic relation (2.3).

On the other hand, there exist some methods to find (2.3) if X is a specific projective curve defined by a few homogeneous polynomials. Here we introduce a method in Hartshorne (1977), Chapter Ⅳ for elliptic curves defined by one polynomial (see also Kudo and Harashita (2017a) for genus-4 curves (resp. Celik et al. (2018) for genus-3 curves) defined by two polynomials).

Example 2.3. Let E be an elliptic curve defined by a cubic form f in $K[x, y, z]$. In this case, the decomposition $E \longrightarrow E_p \longrightarrow E$ with $E_p := V(f^{p-1})$ induces the following commutative diagram:

$$\begin{array}{ccc}
H^1(E, \mathcal{O}_E) & \xrightarrow{\cong} & H^2(\mathbf{P}^2, \mathcal{O}_{\mathbf{P}^2}(-3)) \\
\downarrow & & \downarrow{\scriptstyle F_1^*} \\
H^1(E_p, \mathcal{O}_{E_p}) & \xrightarrow{\cong} & H^2(\mathbf{P}^2, \mathcal{O}_{\mathbf{P}^2}(-3p)) \\
\downarrow & & \downarrow{\scriptstyle \times f^{p-1}} \\
H^1(E, \mathcal{O}_E) & \xrightarrow{\cong} & H^2(\mathbf{P}^2, \mathcal{O}_{\mathbf{P}^2}(-3))
\end{array}$$

where F_1 denotes the absolute Frobenius on $\mathbf{P}^2$. It follows from Theorem 2.1 that the cohomology group $H^1(E, \mathcal{O}_E) \cong H^2(\mathbf{P}^2, \mathcal{O}_{\mathbf{P}^2}(-3))$ has one basis element $(xyz)^{-1}$, and thus its image by F^* is $f^{p-1} \cdot (xyz)^{-p}$. Since $f^{p-1} \cdot (xyz)^{-p}$ is a K-linear combination of monomials of degree -3, we obtain a relation corresponding to (2.3) by regarding monomials other than $(xyz)^{-1}$ in $f^{p-1} \cdot (xyz)^{-p}$ as zeros in $H^2(\mathbf{P}^2, \mathcal{O}_{\mathbf{P}^2}(-3))$.

In Section 3, to device a concrete procedure for a general projective scheme X, we will generalize the method of Example 2.3, and also those proposed in Kudo and Harashita (2017a) and Celik et al. (2018). In particular, we construct two maps corresponding to F_1^* and $H^1(E_p, \mathcal{O}_{E_p}) \longrightarrow H^1(E, \mathcal{O}_E)$ multiplying f^{p-1}, and prove that computing the image of each basis element by the composite map always gives the relation (2.3).

2.3. Properties of graded homomorphisms

We show some properties of graded homomorphisms. These properties are necessary to construct our main algorithm in Section 3. Let S_d denote the d-th homogeneous part of $S = K[x_0, \ldots, x_r]$. For $f \in S$, we denote by $f_{(d)}$ its homogeneous part of degree d.

Lemma 2.4. *Let $h \in S$, and g a homogeneous polynomial of degree d_1. Then we have $(gh)_{(d)} = g(h_{(d-d_1)})$.*

Proof. Put $d_2 := \deg(h)$. Writing $h = \sum_{j=0}^{d_2} h_{(j)}$, we have $gh = g\sum_{j=0}^{d_2} h_{(j)} = \sum_{j=0}^{d_2} gh_{(j)}$. Since each $gh_{(j)}$ is homogeneous of degree $d_1 + j$ and since all $gh_{(j)}$ have distinct degrees, the right hand side gives the decomposition of gh into its homogeneous parts, say $(gh)_{(d)} = g(h_{(d-d_1)})$. □

Lemma 2.5. *Let $M = \bigoplus_{i=1}^{t_M} S(m_i)$, $N = \bigoplus_{k=1}^{t_N} S(n_k)$ and $P = \bigoplus_{j=1}^{t_P} S(p_j)$ be graded free S-modules. Let $\psi : M \to P$ be a (not necessarily graded) homomorphism of S-modules. Let $\varphi : M \to N$ and $\tau : P \to N$ be graded homomorphisms of degree zero. Assume that the following diagram commutes:*

$$\begin{array}{ccc} M & & \\ \downarrow{\scriptstyle \psi} & \searrow{\scriptstyle \varphi} & \\ P & \xrightarrow{\tau} & N \end{array}$$

Then there exists a graded homomorphism $\psi' : M \to P$ of degree zero such that $\tau \circ \psi' = \varphi$. In fact, if $(h_{i,j})_{i,j}$ is the matrix representing ψ via standard bases, such a ψ' is given by

$$\psi'(\mathbf{e}_i) \quad = \quad \sum_{j=1}^{t_P} (h_{i,j})_{(p_j - m_i)} \mathbf{e}_j, \tag{2.4}$$

where each $(h_{i,j})_{(p_j-m_i)}$ denotes the homogeneous part of $h_{i,j}$ of degree $p_j - m_i$.

Proof. Let $(f_{i,k})_{i,k}$, $(g_{j,k})_{j,k}$ and $(h_{i,j})_{i,j}$ be the matrices representing φ, τ and ψ respectively via standard bases. By our assumption, each $f_{i,k}$ is homogeneous of degree $n_k - m_i$, and each $g_{j,k}$ is homogeneous of degree $n_k - p_j$. We define a graded homomorphism $\psi' : M \to P$ by (2.4). In the following, we show $\tau(\psi'(\mathbf{e}_i)) = \varphi(\mathbf{e}_i)$. Note that

$$\tau(\psi'(\mathbf{e}_i)) = \sum_{k=1}^{t_N} \left(\sum_{j=1}^{t_P} g_{j,k} (h_{i,j})_{(p_j - m_i)} \right) \mathbf{e}_k, \text{ and } \varphi(\mathbf{e}_i) = \sum_{k=1}^{t_N} f_{i,k} \mathbf{e}_k.$$

Thus it suffices to show $f_{i,k} = \sum_{j=1}^{t_P} g_{j,k} (h_{i,j})_{(p_j - m_i)}$ for each (i, k). First we

have

$$\begin{aligned}\tau(\psi(\mathbf{e}_i)) &= \tau\left(\sum_{j=1}^{t_P} h_{i,j}\mathbf{e}_j\right) = \sum_{j=1}^{t_P} h_{i,j}\tau(\mathbf{e}_j) = \sum_{j=1}^{t_P} h_{i,j}\left(\sum_{k=1}^{t_N} g_{j,k}\mathbf{e}_k\right) \\ &= \sum_{j=1}^{t_P}\left(\sum_{k=1}^{t_N} h_{i,j}g_{j,k}\mathbf{e}_k\right) = \sum_{k=1}^{t_N}\left(\sum_{j=1}^{t_P} h_{i,j}g_{j,k}\right)\mathbf{e}_k.\end{aligned}$$

On the other hand, it follows from $\tau(\psi(\mathbf{e}_i)) = \varphi(\mathbf{e}_i)$ that we have $f_{i,k} = \sum_{j=1}^{t_P} h_{i,j}g_{j,k}$ for each (i,k). Since $f_{i,k}$ is homogeneous of degree $n_k - m_i$, one has

$$f_{i,k} = \left(\sum_{j=1}^{t_P} h_{i,j}g_{j,k}\right)_{(n_k-m_i)} = \sum_{j=1}^{t_P}\left(h_{i,j}g_{j,k}\right)_{(n_k-m_i)}.$$

Moreover, since the polynomial $g_{j,k}$ is homogeneous of degree $n_k - p_j$, it follows from Lemma 2.4 that

$$\sum_{j=1}^{t_P}\left(h_{i,j}g_{j,k}\right)_{(n_k-m_i)} = \sum_{j=1}^{t_P} g_{j,k}\left(h_{i,j}\right)_{(n_k-m_i-d_{j,k})} = \sum_{j=1}^{t_P} g_{j,k}\left(h_{i,j}\right)_{(p_j-m_i)},$$

where $d_{j,k} := \deg(g_{j,k}) = n_k - p_j$. Hence we have $f_{i,k} = \sum_{j=1}^{t_P} g_{j,k}(h_{i,j})_{(p_j-m_i)}$ for each (i,k), as desired. □

2.4. Free resolutions of finitely generated graded modules

As we will describe in Subsection 3.2, our main algorithm first computes a (graded) *free resolution* (in fact *minimal* one) for the homogeneous coordinate ring of a projective scheme. For the reader's convenience, we here recall the definition of free resolutions, and briefly discuss known algorithms to compute free resolutions (over a polynomial ring), and their complexities. See also Cox et al. (1998), Section 4.8 of Kreuzer and Robbiano (2005), or Section 2.5 of Greuel and Pfister (2007) for the computation of free resolutions.

Let R be a commutative ring with unity. For a finitely generated graded R-module M, a (graded) *free resolution* $\mathbf{F}_\bullet$ for M is an exact complex

$$\cdots \longrightarrow \mathbf{F}_s \xrightarrow{\varphi_s} \cdots \xrightarrow{\varphi_2} \mathbf{F}_1 \xrightarrow{\varphi_1} \mathbf{F}_0 \xrightarrow{\varphi_0} M \longrightarrow 0, \quad (2.5)$$

where $\mathbf{F}_i$ is a graded free R-module, and where φ_i is a graded homomorphism of degree zero for each $i \geq 0$. The greatest integer n with $\mathbf{F}_n \neq 0$ is called the *length* of $\mathbf{F}_\bullet$ if it exists. We also call $\mathbf{F}_\bullet$ *minimal* if each φ_i sends the standard basis of $\mathbf{F}_i$ to a minimal set of generators for $\mathrm{Im}(\varphi_i)$. If R is the polynomial ring of n variables over a field, it is well-known that every finitely generated graded R-module has a minimal free resolution of length $\leq n$, and that minimal free resolutions are uniquely determined up to a graded isomorphism of free resolutions. In this case, minimal resolutions define many invariants of M, e.g., the length of a minimal resolution for M is the *projective dimension* of M.

So far, several algorithms to compute free resolutions have been proposed, e.g., Schreyer (1980), Schreyer (1991), Capani et al. (1997), La Scala and Stillman (1998), Erocal et al. (2016), La Scala (2017), where many experimental data showing practical behavior have been also reported. However, as we mentioned in Section 1, it is still an open problem to determine the complexity of computing free resolutions. Schreyer's algorithm(s) and La Scala-Stillman's one have been considered to be the most efficient algorithms in general case, and they have been already widely implemented in computer algebra systems such as Macaulay2, Magma, Singular and CoCoA. These algorithms construct free resolutions called *Schreyer resolutions* by successively computing syzygy modules with Gröbner bases over free modules. The complexities of these algorithms would be measured by invariants such as projective dimension, (graded) Betti numbers and so on, but they have not been estimated yet since Schreyer resolutions are not minimal in general.

2.5. Frobenius functor for the category of modules

This subsection collects properties of the Frobenius functor for the category of modules, see e.g., Miler (2003) for more details. Let R be a ring of positive characteristic p, M an R-module, and f the Frobenius endomorphism on R. We denote by ${}^fM$ the left R-module structure defined on M by restriction of scalars via f, that is, for $r \in R$ and $m \in M$, we define $r \cdot m := r^p m$.

The Frobenius functor is defined as a functor from the category of R-modules to itself, and it is defined by $F_R(M) := M \otimes_R {}^fR$. In the following lemma, we enumerate some fundamental properties of the Frobenius functor $F_R(\cdot)$:

Lemma 2.6. *Let R be a ring of positive characteristic p, f the Frobenius endomorphism on R, and $F_R(\cdot)$ the Frobenius functor from the category of R-modules to itself. Then we have the following:*

(1) *The functor $F_R(\cdot)$ is right exact.*

(2) *There exist isomorphisms $F_R(R) = R \otimes_R {}^fR \cong {}^fR \cong R$ as R-modules via $a \otimes b \mapsto a \cdot b = a^p b$. For free modules, one has $F_R(R^t) = (\bigoplus_{i=1}^t R) \otimes_R {}^fR \cong ({}^fR)^t \cong R^t$ via $(a_1, \ldots, a_t) \otimes b \mapsto (a_1 \cdot b, \ldots, a_t \cdot b) = (a_1^p b, \ldots, a_t^p b)$.*

(3) *For any ideal $J \subset R$, we have $F_R(R/J) = (R/J) \otimes_R {}^fR \cong R/J_p$, where J_p denotes the ideal generated by the p-th powers of elements in J.*

(4) *Let $\varphi : R^t \to R^s$ be a homomorphism of R-modules, and $(r_{i,j})_{i,j}$ a $t \times s$ matrix which represents φ via standard bases. Then $F_R(\varphi) : R^t \to R^s$ is given by $(r_{i,j}^p)_{i,j}$.*

Proof. (1) Since tensor product is right exact, the claim holds. (2) Straightforward. (3) The claim follows from $F_R(R/J) = (R/J) \otimes_R {}^fR \cong R/(J \cdot {}^fR)$, where in this case $J \cdot {}^fR := \langle a^p x : a \in J,\ x \in {}^fR \rangle_R = J_p$. (4) Let $\mathbf{e}_i$ be an element of the standard basis of R^t. By (2), we identify $\mathbf{e}_i$ with $\mathbf{e}_i \otimes 1$, and it follows that $F_R(\varphi)(\mathbf{e}_i) = (\varphi \otimes \mathrm{id}_{{}^fR})(\mathbf{e}_i \otimes 1) = (\sum_{j=1}^s r_{i,j}\mathbf{e}_j) \otimes 1 = \sum_{j=1}^s (r_{i,j}\mathbf{e}_j \otimes 1) = \sum_{j=1}^s (r_{i,j} \cdot 1)\mathbf{e}_j = \sum_{j=1}^s r_{i,j}^p \mathbf{e}_j$. □

Theorem 2.7 (Kunz's Theorem, Kunz (1969), Theorems 2.1 and 3.3)**.** *Let R be a local ring of characteristic p. Then R is a regular ring if and only if f^n is flat for all $n > 0$, where f denotes the Frobenius endomorphism on R.*

Since regularity and flatness can be each checked locally, we have the following corollary:

Corollary 2.8. *Let R be a ring of positive characteristic p. Then R is a regular ring if and only if f^n is flat for all $n > 0$, where f denotes the Frobenius endomorphism on R.*

Lemma 2.9. *Let L be a field of characteristic $p > 0$, and $R := L[y_1, \ldots, y_n]$ the polynomial ring with n variables over L. Let $f_1, \ldots, f_m \in R$ be homogeneous polynomials with $d_j^{(1)} := \deg(f_j)$ for $1 \leq j \leq m$, and $J \subset R$ the ideal generated by $f_1, \ldots, f_m$. Suppose that R/J has the following graded free resolution:*

$$0 \longrightarrow \mathbf{F}_n \xrightarrow{\varphi_n} \cdots \xrightarrow{\varphi_2} \mathbf{F}_1 \xrightarrow{\varphi_1} R \xrightarrow{\varphi_0} R/J \longrightarrow 0, \quad (2.6)$$

where each $\mathbf{F}_i$ is a graded free R-module given by $\mathbf{F}_i = \bigoplus_{j=1}^{t_i} R(-d_j^{(i)})$ for some integers t_i and $d_j^{(i)}$, and where each φ_i is a graded homomorphism of degree zero. Let $J_p := \langle f_1^p, \ldots, f_m^p \rangle_R$ be the ideal in R generated by $f_1^p, \ldots, f_m^p$. Then R/J_p has a graded free resolution of the form

$$0 \longrightarrow \mathbf{F}_n^{(p)} \xrightarrow{\varphi_n^{(p)}} \cdots \xrightarrow{\varphi_2^{(p)}} \mathbf{F}_1^{(p)} \xrightarrow{\varphi_1^{(p)}} R \xrightarrow{\varphi_0^{(p)}} R/J_p \longrightarrow 0, \quad (2.7)$$

where each $\mathbf{F}_i^{(p)}$ is a graded free R-module given by $\mathbf{F}_i^{(p)} = \bigoplus_{j=1}^{t_i} R(-d_j^{(i)}p)$, and $\varphi_i^{(p)}$ is given by the matrix with entries equal to the p-th powers of the entries of the matrix for φ_i for each $0 \leq i \leq n$.

Proof. By Lemma 2.6 and Corollary 2.8 together with the fact that R is a regular ring of dimension n, the sequence

$$0 \longrightarrow \mathbf{F}_n^{(p)} \xrightarrow{\varphi_n^{(p)}} \cdots \xrightarrow{\varphi_2^{(p)}} \mathbf{F}_1^{(p)} \xrightarrow{\varphi_1^{(p)}} R \xrightarrow{\varphi_0^{(p)}} R/J_p \longrightarrow 0 \quad (2.8)$$

is exact. It is straightforward that the sequence (2.8) gives a graded free resolution for R/J_p. □

3. Algorithm

As in the previous section, let $S = K[x] = K[x_0, \ldots, x_r]$ denote the polynomial ring of $r + 1$ variables over a perfect field K of characteristic $p > 0$. Let $I = \langle f_1, \ldots, f_m \rangle_S$ denote the ideal in S generated by homogeneous polynomials $f_1, \ldots, f_m \in S$. Let $X = V(I) \subset \mathbf{P}^r$ be the projective scheme defined by I, and let $\mathcal{O}_X$ denote its structure sheaf. Given $f_1, \ldots, f_m$ and an integer $1 \leq q \leq r - 1$, we present an explicit algorithm to compute a representation matrix for the Frobenius F^* on the q-th cohomology group $H^q(X, \mathcal{O}_X)$.

Here, we recall our procedures in Section 1 to compute F^*.

Step 1. Compute an explicit basis of the cohomology group $H^q(X, \mathcal{O}_X)$.

Step 2. Make a representation matrix of F^* with respect to the computed basis.

After fixing the notation in Subsection 3.1 below, we shall construct the algorithm in Subsection 3.2. The correctness with our solution (i)-(iii) in Section 1 and the complexity of the algorithm will be discussed respectively in Subsections 3.3 and 3.4.

3.1. Notation

Until the end of this section, we will keep the notation below for simplicity. Let $\mathbf{F}_\bullet$ denote the minimal free resolution for S/I given by

$$0 \longrightarrow \mathbf{F}_{r+1} \xrightarrow{\varphi_{r+1}} \cdots \xrightarrow{\varphi_2} \mathbf{F}_1 \xrightarrow{\varphi_1} S \xrightarrow{\varphi_0} S/I \longrightarrow 0, \quad (3.1)$$

where each $\mathbf{F}_i$ is of the form $\mathbf{F}_i = \bigoplus_{j=1}^{t_i} S(-d_j^{(i)})$ for some integers t_i and $d_j^{(i)}$, and where each φ_i is a graded homomorphism of degree zero with a representation matrix $\left(g_{k,\ell}^{(i)}\right)_{k,\ell}$. Note that each entry $g_{k,\ell}^{(i)}$ is a homogeneous polynomial in S of degree $d_k^{(i)} - d_\ell^{(i)}$. Computing $\mathbf{F}_\bullet$ means to compute all the elements

$$t_i,\ d_j^{(i)},\ \text{and}\ \left(g_{k,\ell}^{(i)}\right)_{k,\ell}\ \text{for}\ 1 \le i \le r+1. \quad (3.2)$$

The above resolution (3.1) induces an exact sequence of sheaves as follows:

$$0 \longrightarrow \mathcal{F}_{r+1} \xrightarrow{\varphi_{r+1}^\sim} \cdots \xrightarrow{\varphi_2^\sim} \mathcal{F}_1 \xrightarrow{\varphi_1^\sim} \mathcal{O}_{\mathbf{P}^r} \xrightarrow{\varphi_0^\sim} \mathcal{O}_X \longrightarrow 0,$$

where we set $\mathcal{F}_i := \mathbf{F}_i^\sim = \bigoplus_{j=1}^{t_i} \mathcal{O}_{\mathbf{P}^r}(-d_j^{(i)})$ for $1 \le i \le r+1$, and where each $\varphi_i^\sim$ is the morphism induced from φ_i. We denote by $H^r(\mathbf{P}^r, \mathcal{F}_\bullet)$ the complex of cohomology groups given by $H^r(\varphi_i^\sim) : H^r(\mathbf{P}^r, \mathcal{F}_i) \longrightarrow H^r(\mathbf{P}^r, \mathcal{F}_{i-1})$. We set

$$g := \dim_K \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet)), \qquad g_0 := \dim_K H^r(\mathbf{P}^r, \mathcal{F}_{r-q}),$$

$$g_1 := \dim_K \mathrm{Ker}\left(H^r(\varphi_{r-q}^\sim)\right), \qquad g_2 := \dim_K \mathrm{Im}\left(H^r(\varphi_{r-q+1}^\sim)\right),$$

$$t := t_{r-q},\ \text{and}\ d_j := d_j^{(r-q)}\ \text{for}\ 1 \le j \le t,$$

where $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet)) := \mathrm{Ker}\left(H^r(\varphi_{r-q}^\sim)\right) / \mathrm{Im}\left(H^r(\varphi_{r-q+1}^\sim)\right)$. It follows from Theorem 2.1 that the K-vector space $H^r(\mathbf{P}^r, \mathcal{F}_{r-q})$ has the basis

$$\mathcal{V} := \{\mathbf{v}_k : 1 \le k \le g_0\} = \{x^\alpha \mathbf{e}_j : 1 \le j \le t,\ \alpha \in (\mathbb{Z}_{<0})^{r+1},\ |\alpha| = -d_j\},$$

where we may assume the following: For each k, there exists $1 \le j(k) \le t$ such that $\mathbf{v}_k = x^{\alpha(k)}\mathbf{e}_{j(k)}$ for some $\alpha(k) \in (\mathbb{Z}_{<0})^{r+1}$ with $|\alpha(k)| = -d_{j(k)}$.

3.2. Main algorithm

In the following, we give our main algorithm. We first fix r, which is the dimension of $\mathbf{P}^r = \mathrm{Proj}(S)$ with $S = K[x] = K[x_0, \ldots, x_r]$. The inputs are a tuple of homogeneous polynomials $(f_1, \ldots, f_m) \in S^m$ with $I = \langle f_1, \ldots, f_m \rangle_S$ and $X = V(I) \subset \mathbf{P}^r$, the characteristic p, and an integer $1 \le q \le r-1$.

Algorithm (I). Given $(f_1, \ldots, f_m)$, p and q as above, this algorithm computes a representation matrix for the Frobenius $F^* : H^q(X, \mathcal{O}_X) \longrightarrow H^q(X, \mathcal{O}_X)$ with respect to a suitable basis. This algorithm conducts the following two main procedures Steps 1 and 2, where Step 1 (resp. Step 2) consists of two (resp. four) sub-procedures (1-1)–(1-2) (resp. (2-1)–(2-4)):

Step 1. Compute a free resolution for S/I and an explicit basis of $H^q(X, \mathcal{O}_X)$. More precisely this step is divided into the following two steps:

(1-1) Set $\mathbf{F}_\bullet \leftarrow$ (the minimal free resolution for S/I), where a free resolution is given by the elements (3.2) in Subsection 3.1.

(1-2) Applying the algorithm given in Kudo (2017), Section 3, we obtain the following four bases:

- $\mathcal{V} := \{\mathbf{v}_k : 1 \leq k \leq g_0\}$ for $H^r(\mathbf{P}^r, \mathcal{F}_{r-q})$,
- $\mathcal{A} := \{\mathbf{a}_i := \sum_{k=1}^{g_0} a_{i,k}\mathbf{v}_k : 1 \leq i \leq g_2\}$ for $\mathrm{Im}\left(H^r(\varphi^{\sim}_{r-q+1})\right)$,
- $\mathcal{B} := \{\mathbf{b}_i := \sum_{k=1}^{g_0} b_{i,k}\mathbf{v}_k : 1 \leq i \leq g\}$ for the right hand side of $H^q(X, \mathcal{O}_X) \cong \mathrm{Ker}\left(H^r(\varphi^{\sim}_{r-q})\right) / \mathrm{Im}\left(H^r(\varphi^{\sim}_{r-q+1})\right)$,
- $\mathcal{A} \cup \mathcal{B}$ for $\mathrm{Ker}\left(H^r(\varphi^{\sim}_{r-q})\right)$,

where $A := (a_{i,k})_{i,k}$ (resp. $B = (b_{i,k})_{i,k}$) is a $g_2 \times g_0$ (resp. $g \times g_0$) matrix over K.

Step 2. For $\mathbf{F}_\bullet$ and $(\mathcal{V}, \mathcal{A}, \mathcal{B})$ obtained in Step A, this step computes the representation matrix for $F^* : H^q(X, \mathcal{O}_X) \longrightarrow H^q(X, \mathcal{O}_X)$ with respect to the basis $\mathcal{B}$. This step is divided into the following four steps:

(2-1) Set $\mathbf{F}^{(p)}_\bullet \leftarrow$ (the minimal free resolution for S/I_p), which is computed by taking the p-th powers of the entries of each representation matrix for $\mathbf{F}_\bullet$, see Lemma 2.9.

(2-2) Set $\psi_\bullet \leftarrow$ `ComputeAll_LIFT`$(\mathbf{F}_\bullet, \mathbf{F}^{(p)}_\bullet)$.

(2-3) Set $\mathcal{B}^{(p)} := \{\mathbf{b}_i^{(p)} : 1 \leq i \leq g\} \leftarrow$ `Make_image_by_Frobenius`$(\mathcal{B})$.

(2-4) Set $Y \leftarrow$ `REP`$(\mathcal{B}^{(p)}, \psi_{r-q}, \mathcal{V}, \mathcal{A}, \mathcal{B})$, and return Y as the final output of Algorithm (I).

Sub-algorithm `ComputeAll_LIFT`$(\mathbf{F}_\bullet, \mathbf{F}^{(p)}_\bullet)$. This sub-algorithm computes a morphism of complexes $\psi_\bullet : \mathbf{F}^{(p)}_\bullet \longrightarrow \mathbf{F}_\bullet$, which is given by a sequence of graded homomorphisms ψ_i of degree zero such that the following diagram commutes:

$$\begin{array}{ccccccccccccc}
0 & \longrightarrow & \mathbf{F}^{(p)}_{r+1} & \xrightarrow{\varphi^{(p)}_{r+1}} & \cdots & \xrightarrow{\varphi^{(p)}_2} & \mathbf{F}^{(p)}_1 & \xrightarrow{\varphi^{(p)}_1} & S & \xrightarrow{\varphi^{(p)}_0} & S/I_p & \longrightarrow & 0 \\
 & & \downarrow{\scriptstyle \psi_{r+1}} & & & & \downarrow{\scriptstyle \psi_1} & & \downarrow{\scriptstyle \psi_0} & & \downarrow{\scriptstyle \psi} & & \\
0 & \longrightarrow & \mathbf{F}_{r+1} & \xrightarrow{\varphi_{r+1}} & \cdots & \xrightarrow{\varphi_2} & \mathbf{F}_1 & \xrightarrow{\varphi_1} & S & \xrightarrow{\varphi_0} & S/I & \longrightarrow & 0
\end{array}$$

where ψ_0 is the identity map on S, and ψ is given by $h + I_p \mapsto h + I$ for $h \in S$. More specifically, compute all representation matrices for ψ_i with $1 \leq i \leq r+1$ by repeating the three procedures below until $i = r + 1$: Set $i \leftarrow 1$.

1. Set $\varphi \leftarrow \varphi_i$, $\varphi' \leftarrow \psi_{i-1} \circ \varphi_i^{(p)}$ and $t' \leftarrow t_i$
2. Execute $\texttt{LIFT}(\varphi, \varphi')$ below to obtain a matrix C_i representing a graded homomorphism ψ_i of degree zero such that $\varphi' = \varphi \circ \psi_i$.
3. Set $i \leftarrow i + 1$.

Sub-algorithm $\texttt{LIFT}(\varphi, \varphi')$. Compute $\varphi(\mathbf{e}_j)$ and $\varphi'(\mathbf{e}_j)$ for each $1 \leq j \leq t'$. By a module membership algorithm (see Section 2.8.1 in Greuel and Pfister (2007) and Section 4.1 of Decker and Lossen (2000)), compute $h_{j,k} \in S$ such that $\varphi'(\mathbf{e}_j) = \sum_{k=1}^{t} h_{j,k}\varphi(\mathbf{e}_k)$, where we note that this computation of $h_{j,k}$'s requires to compute a Gröbner basis of the syzygy module for $\varphi'(\mathbf{e}_j)$ and $-\varphi(\mathbf{e}_k)$ with $1 \leq k \leq t'$. We may assume that each $h_{j,k}$ is homogeneous; if not homogeneous, by Lemma 2.5 we replace it by its homogeneous part of some degree so that the map ψ below is a graded homomorphism of degree zero. Define ψ_i by $\psi_i(\mathbf{e}_j) := \sum_{k=1}^{t} h_{j,k}\mathbf{e}_k$, and output its representation matrix $(h_{j,k})_{j,k}$. Note that ψ_i is a graded homomorphism of degree zero, see Lemma 2.5.

Sub-algorithm $\texttt{Make\_image\_by\_Frobenius}(\mathcal{B})$. For the basis $\mathcal{B}$ computed in Step (1-2), compute the image of each element of $\mathcal{B}$ by F_1^*, where F_1 denotes the absolute Frobenius on $\mathbf{P}^r$. Note that the map F_1^* acts just like raising the coordinates of each $\mathbf{b}_i$ to their p-th powers. Thus each image is simply computed as $\mathbf{b}_i^{(p)} = \sum_{k=1}^{g_0} b_{i,k}^p \mathbf{v}_k^{(p)}$ with $\mathbf{v}_k^{(p)} := x^{p\cdot\alpha(k)}\mathbf{e}_{j(k)}$, but for the next step (Step 2-4) we also compute each coordinate of $\mathbf{b}_i^{(p)}$ as follows: Repeat the three procedures below until $i = g$: Set $i \leftarrow 1$.

1. Set
$$g_{i,j} \leftarrow \sum_{1\leq k\leq g_0,\ j(k)=j} b_{i,k}x^{\alpha(k)}$$
for each $1 \leq j \leq t$, where we have $\mathbf{b}_i = \sum_{j=1}^{t} g_{i,j}\mathbf{e}_j$.
2. Compute $g_{i,j}^p$ for all $1 \leq j \leq t$, and set $\mathbf{b}_i^{(p)} \leftarrow \sum_{j=1}^{t} g_{i,j}^p \mathbf{e}_j$.
3. Set $i \leftarrow i + 1$.

Sub-algorithm $\texttt{REP}(\mathcal{B}^{(p)}, \psi_{r-q}, \mathcal{V}, \mathcal{A}, \mathcal{B})$. Let $(h_{j,k})_{j,k}$ be the representation matrix for the graded homomorphism ψ_{r-q}. For each $1 \leq i \leq g$, compute the right hand side of

$$\mathbf{b}_i^{(p)} \cdot C_{r-q} = \sum_{k=1}^{t} \left(\sum_{j=1}^{t} g_{i,j}^p h_{j,k} \right) \mathbf{e}_k, \tag{3.3}$$

and write it as

$$\sum_{k=1}^{t} \left(\sum_{j=1}^{t} g_{i,j}^p h_{j,k} \right) \mathbf{e}_k = \sum_{k=1}^{g_0} b'_{i,k}\mathbf{v}_k \tag{3.4}$$

for some $b'_{i,k} \in K$. Thanks to Lemma 3.5 below, such $b'_{i,k}$'s are definitely found. Solve the linear system

$$\begin{pmatrix} b'_{1,1} & \cdots & b'_{1,g_0} \\ \vdots & & \vdots \\ b'_{g,1} & \cdots & b'_{g,g_0} \end{pmatrix} = \begin{pmatrix} x_{1,1} & \cdots & x_{1,g_2} & y_{1,1} & \cdots & y_{1,g} \\ \vdots & & \vdots & \vdots & & \vdots \\ x_{g,1} & \cdots & x_{g,g_2} & y_{g,1} & \cdots & y_{g,g} \end{pmatrix} \begin{pmatrix} A \\ B \end{pmatrix} \quad (3.5)$$

over K, and output the representation matrix $Y = (y_{i,j})_{i,j}$

Remark 3.1. As we stated in Remark 1.1, the complexity of computing lifting homomorphisms has not been estimated yet due to the difficulty of estimating the cost of computing a Gröbner basis of a syzygy module. The complexity might be measured by invariants such as (graded) Betti numbers appearing in minimal free resolutions together with our lemma (Lemma 2.5) on the degrees of entries of matrices representing lifting homomorphisms. (Also in Bayer and Stillman (1988), a bound on the degrees of the entries of elements in the syzygy module for a homogeneous ideal is given, and it could be extended to the case of graded submodules.)

3.3. Correctness of the main algorithm

This subsection shows the correctness of Algorithm (I) in the previous subsection, based on mathematical foundations given in Section 2. In particular, we realize (i)-(iii) described in Section 1. We set $I_p := \langle f_1^p, \ldots, f_m^p \rangle_S$ and $X_p := V(I_p)$.

First, we give a key lemma, by which $F^* : H^q(X, \mathcal{O}_X) \longrightarrow H^q(X, \mathcal{O}_X)$ can be decomposed into two *computable* maps; a p-linear map followed by a K-linear map between cohomology groups over $\mathbf{P}^r$.

Lemma 3.2. *With notation as above, there exist a resolution $\mathcal{F}_\bullet^{(p)}$ for $\mathcal{O}_{X_p}$ and the following commutative diagram:*

$$\begin{array}{ccc} H^q(X, \mathcal{O}_X) & \xrightarrow{\cong} & \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet)) \\ \downarrow & & \downarrow{\scriptstyle F_1^*} \\ H^q(X_p, \mathcal{O}_{X_p}) & \xrightarrow{\cong} & \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)})) \\ \downarrow & & \downarrow \\ H^q(X, \mathcal{O}_X) & \xrightarrow{\cong} & \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet)) \end{array}$$

where F_1 denotes the absolute Frobenius map on $\mathbf{P}^r$. In particular, the map $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)})) \longrightarrow \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet))$ in the above diagram is a K-linear map canonically induced by the homomorphism $H^r(\psi_\bullet^{\sim}) : H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)}) \longrightarrow H^r(\mathbf{P}^r, \mathcal{F}_\bullet)$ of complexes.

Proof. We decompose the absolute Frobenius $F : X \longrightarrow X$ into the composition of the following two morphisms $X \longrightarrow X_p$ with $\psi^{\sim} : \mathcal{O}_{X_p} \longrightarrow \mathcal{O}_X$ followed by $X_p \longrightarrow X$ with $\sigma^{\sim} : \mathcal{O}_X \longrightarrow \mathcal{O}_{X_p}$, where $\sigma : S/I \longrightarrow S/I_p$ is defined by $h + I \mapsto h^p + I_p$. In particular, the p-th power map on the coordinate ring S/I defined by F is decomposed into $\psi \circ \sigma$. Let $\psi_\bullet : \mathbf{F}_\bullet^{(p)} \longrightarrow \mathbf{F}_\bullet$ be a homomorphism of complexes computed by executing the sub-algorithm `ComputeAll_LIFT`$(\mathbf{F}_\bullet, \mathbf{F}_\bullet^{(p)})$. We have the following commutative diagram:

$$\begin{array}{ccccccccccccc}
0 & \longrightarrow & \mathbf{F}_{r+1} & \xrightarrow{\varphi_{r+1}} & \cdots & \xrightarrow{\varphi_2} & \mathbf{F}_1 & \xrightarrow{\varphi_1} & S & \xrightarrow{\varphi_0} & S/I & \longrightarrow & 0 \\
 & & \downarrow{\scriptstyle \sigma_{r+1}} & & & & \downarrow{\scriptstyle \sigma_1} & & \downarrow{\scriptstyle \sigma_0} & & \downarrow{\scriptstyle \sigma} & & \\
0 & \longrightarrow & \mathbf{F}_{r+1}^{(p)} & \xrightarrow{\varphi_{r+1}^{(p)}} & \cdots & \xrightarrow{\varphi_2^{(p)}} & \mathbf{F}_1^{(p)} & \xrightarrow{\varphi_1^{(p)}} & S & \xrightarrow{\varphi_0^{(p)}} & S/I_p & \longrightarrow & 0 \\
 & & \downarrow{\scriptstyle \psi_{r+1}} & & & & \downarrow{\scriptstyle \psi_1} & & \downarrow{\scriptstyle \psi_0} & & \downarrow{\scriptstyle \psi} & & \\
0 & \longrightarrow & \mathbf{F}_{r+1} & \xrightarrow{\varphi_{r+1}} & \cdots & \xrightarrow{\varphi_2} & \mathbf{F}_1 & \xrightarrow{\varphi_1} & S & \xrightarrow{\varphi_0} & S/I & \longrightarrow & 0
\end{array}$$

where each σ_i is an S-homomorphism raising the coordinates to their p-th powers. Namely, we have the composition of the following two homomorphisms of complexes: $\sigma_\bullet : \mathbf{F}_\bullet \longrightarrow \mathbf{F}_\bullet^{(p)}$ and $\psi_\bullet : \mathbf{F}_\bullet^{(p)} \longrightarrow \mathbf{F}_\bullet$, which also induce morphisms of coherent sheaves $\sigma_\bullet^{\sim} : \mathcal{F}_\bullet \longrightarrow \mathcal{F}_\bullet^{(p)}$ and $\psi_\bullet^{\sim} : \mathcal{F}_\bullet^{(p)} \longrightarrow \mathcal{F}_\bullet$, where $\mathcal{F}_i^{(p)} := (\mathbf{F}_i^{(p)})^{\sim} = \bigoplus_{j=1}^{t_i} \mathcal{O}_{\mathbf{P}^r}\left(-pd_j^{(i)}\right)$ for $1 \leq i \leq r+1$. Hence we have the following homomorphisms of complexes of cohomology groups: $H^r(\sigma_\bullet^{\sim}) : H^r(\mathbf{P}^r, \mathcal{F}_\bullet) \longrightarrow H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)})$ and $H^r(\psi_\bullet^{\sim}) : H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)}) \longrightarrow H^r(\mathbf{P}^r, \mathcal{F}_\bullet)$. Finally we obtain the desired commutative diagram in the statement, as well as the proof of the isomorphism

$$H^q(X, \mathcal{O}_X) \cong \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet)) = \mathrm{Ker}\left(H^r(\varphi_{r-q}^{\sim})\right) / \mathrm{Im}\left(H^r(\varphi_{r-q+1}^{\sim})\right)$$

given in Kudo (2017), Theorem 5. □

Remark 3.3. Each element in $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)}))$ is given as the equivalence class of an element in $\mathrm{Ker}\left(H^r((\varphi_{r-q}^{(p)})^{\sim})\right)$, which is a subspace of $H^r(\mathbf{P}^r, \mathcal{F}_{r-q}^{(p)})$ with $\mathcal{F}_{r-q}^{(p)} = \bigoplus_{j=1}^{t} \mathcal{O}_{\mathbf{P}^r}(-pd_j)$. It follows from Theorem 2.1 that an element in $\mathrm{Ker}\left(H^r((\varphi_{r-q}^{(p)})^{\sim})\right)$ is of the form $\sum_{j=1}^{t} g_j' \mathbf{e}_j$, where each g_j' is a K-linear combination of monomials of negative degree $-d_j p$.

The following corollary from Lemma 3.2 shows that the image of each element in $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)}))$ by $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)})) \longrightarrow \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet))$ is computed just by multiplying the representation matrix C_{r-q} for ψ_{r-q}.

Corollary 3.4. *Let* $\sum_{j=1}^{t} g_j' \mathbf{e}_j$ *be an element in* $\mathrm{Ker}\left(H^r((\varphi_{r-q}^{(p)})^{\sim})\right)$ *given in Remark 3.3. The map* $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)})) \longrightarrow \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet))$ *in Lemma*

3.2 sends the class of $\sum_{j=1}^{t} g'_j \mathbf{e}_j$ *to that of the element* $\sum_{k=1}^{t} \left(\sum_{j=1}^{t} g'_j h_{j,k} \right) \mathbf{e}_k$ *in* $\mathrm{Ker}\left(H^r(\varphi^{\sim}_{r-q})\right)$*, where* $(h_{j,k})_{j,k}$ *is a matrix over* S *representing* ψ_{r-q}*.*

Lemma 3.5. *With the same notation as in Corollary 3.4, computing* $g'_j h_{j,k}$ *for* $1 \leq j, k \leq t$ *enables us to find* $b'_k \in K$ *with* $1 \leq k \leq g_0$ *such that*

$$\sum_{k=1}^{t} \left(\sum_{j=1}^{t} g'_j h_{j,k} \right) \mathbf{e}_k = \sum_{k=1}^{g_0} b'_k \mathbf{v}_k.$$

Proof. Let C_i denote the representation matrix for ψ_i for $1 \leq i \leq r+1$. From our construction of ψ_i in the sub-algorithm `ComputeAll_LIFT`, it follows from Lemma 2.5 that each (j,k)-entry of C_i is homogeneous of degree $(-d_k^{(i)}) - (-d_j^{(i)} p) = d_j^{(i)} p - d_k^{(i)}$. In particular, the (j,k)-entry $h_{j,k}$ of C_{r-q} is of degree $d_j p - d_k$. Since g'_j is a K-linear combination of monomials of negative degree $-d_j p$, the k-th coordinate $g'_j h_{j,k}$ is that of monomials of negative degree $-d_k$.

On the other hand, the set $\mathcal{V} = \{v_1, \ldots, v_{g_0}\}$ consists of elements of the form $x^{\alpha(k)} \mathbf{e}_k$ for all $1 \leq k \leq t$ and all $\alpha(k) \in (\mathbb{Z}_{<0})^{r+1}$ with $|\alpha(k)| = -d_k$. Therefore we can find desired $b'_k \in K$. □

Proof of the correctness of Algorithm (I). We show that Algorithm (I) outputs the representation matrix for the composite map of the right column of the diagram in Lemma 3.2. In Step (1), the basis $\mathcal{B}$ of $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet))$ is constructed. From Steps (2-1) to (2-2), the representation matrix $C_{r-q} = (h_{j,k})_{j,k}$ for ψ_{r-q} is computed. The set $\mathcal{B}^{(p)}$ computed in Step (2-3) is a subset of $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)}))$, and by the definition of F_1^* it is the image of $\mathcal{B}$ by F_1^*, see Sub-algorithm `Make_image_by_Frobenius`$(\mathcal{B})$. It suffices to show that the sub-algorithm `REP`$(\mathcal{B}^{(p)}, \psi_{r-q}, \mathcal{V}, \mathcal{A}, \mathcal{B})$ computes the desired matrix representing F^*. In `REP`$(\mathcal{B}^{(p)}, \psi_{r-q}, \mathcal{V}, \mathcal{A}, \mathcal{B})$, we compute $b'_{i,k} \in K$ such that (3.4) holds. By Lemma 3.5, such $b'_{i,k}$'s are found by computing $g^p_{i,j} h_{j,k}$ for $1 \leq j, k \leq t$. Since the image of each $F_1^*(\mathbf{b}_i)$ by the map $\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)})) \longrightarrow \mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet))$ is included in $\mathrm{Ker}\left(H^r(\varphi^{\sim}_{r-q})\right)$, there exist the $(g \times g_2)$ matrix $X = (x_{i,j})_{i,j}$ and the $(g \times g)$ matrix $Y = (y_{i,j})_{i,j}$ over K such that

$$\begin{aligned} \sum_{k=1}^{g_0} b'_{i,k} \mathbf{v}_k &= \sum_{j=1}^{g_2} x_{i,j} \mathbf{a}_j + \sum_{j=1}^{g} y_{i,j} \mathbf{b}_j \\ &= \sum_{j=1}^{g_2} x_{i,j} \left(\sum_{k=1}^{g_0} a_{j,k} \mathbf{v}_k \right) + \sum_{j=1}^{g} y_{i,j} \left(\sum_{k=1}^{g_0} b_{j,k} \mathbf{v}_k \right), \end{aligned}$$

where $\mathcal{A} \cup \mathcal{B} = \{\mathbf{a}_1, \ldots, \mathbf{a}_{g_2}, \mathbf{b}_1, \ldots, \mathbf{b}_g\}$ is a basis of $\mathrm{Ker}\left(H^r(\varphi^{\sim}_{r-q})\right)$ computed in Step (1-2). Thus X and Y are definitely computed. Here we have $\sum_{k=1}^{g_0} b'_{i,k} \mathbf{v}_k = \sum_{j=1}^{g} y_{i,j} \mathbf{b}_j$ in $\mathrm{Ker}\left(H^r(\varphi^{\sim}_{r-q})\right) / \mathrm{Im}\left(H^r(\varphi^{\sim}_{r-q+1})\right)$ for each $1 \leq i \leq g$, and thus Y is the desired representation matrix. □

3.4. Complexity analysis

In this subsection, we investigate the complexity of Algorithm (I) given in Subsection 3.2. Recall that the input objects of the algorithm are a tuple of homogeneous polynomials $(f_1, \ldots, f_m) \in S^m$ with $S = K[x] = K[x_0, \ldots, x_r]$, the characteristic p, and an integer $1 \leq q \leq r-1$. As we described in Remark 1.1, the complexities of the computations of free resolutions and lifting homomorphisms (Steps (1-1), (2-1) and (2-2)) have not been determined yet in general. For fixed r and q, the output object is determined by p and the elements of (3.2) and the $t \times t$ matrix C_{r-q} over S, which are computed in Steps (1-1), (2-1) and (2-2). From this, we estimate the complexity of Algorithm (I) according to the parameters p, $t^{(\max)} := \max\{t_i : r-q-1 \leq i \leq r-q+1\}$, and $d^{(\max)} := \max\{d^{(i,\max)} : r-q-1 \leq i \leq r-q+1\}$, where $d^{(i,\max)} := \max\{d_j^{(i)} : 1 \leq j \leq t_i\}$. In the following, we denote by $\mathsf{P}(e)$ the number of arithmetic operations over K for computing the e-th power for each $e \in \mathbb{Z}_{\geq 1}$. We assume that $\mathsf{MultMono}(s)$ (resp. $\mathsf{PowerMono}(m)$) is a function such that the product of two monomials x^α and x^β (resp. the power $(x^\alpha)^n$) can be computed in time $\mathsf{MultMono}(|\text{total. deg}(x^{\alpha+\beta})|)$ (resp. $\mathsf{MultMono}(|\text{total. deg}(x^{n\alpha})|)$), where α and β can take elements in $\mathbb{Z}^{r+1}$ (not only in $(\mathbb{Z}_{\geq 0})^{r+1}$).

Proposition 3.6. *With notation as above, the total complexity of* Steps (1-2), (2-3) *and* (2-4) *of* Algorithm (I) *in* Subsection 3.2 (*not counting the computation of a basis of* $H^r(\mathbf{P}^r, \mathcal{F}_i)$ *for* $r-q-1 \leq i \leq r-q+1$) *is*

$$\tilde{O}\left(\left(t^{(\max)}(d^{(\max)})^r\right)^4 + \left(t^{(\max)}(d^{(\max)})^r\right)^2 \mathsf{P}(p) + (t^{(\max)})^2 (d^{(\max)})^{3r} p^r\right) \tag{3.6}$$

arithmetic operations over K.

Proof. First consider Step (1-2), where we compute the basis $\mathcal{B}$ of $H^q(X, \mathcal{O}_X)$. From Kudo (2017), Corollary 16, the complexity for Step (1-2) is estimated as

$$O\left(\left(t^{(\max)}(d^{(\max)})^r\right)^4\right) \tag{3.7}$$

arithmetic operations over K. Note that we also used the following fact to obtain (3.7): The number of the non-zero terms of each (j,k)-entry of the representation matrix for φ_i is bounded by $\binom{d_j^{(i)} - d_k^{(i)} + r}{r} = O((d^{(\max)})^r)$, since the (j,k)-entry is homogeneous of degree $d_j^{(i)} - d_k^{(i)}$ in $r+1$ variables.

Next we consider Step (2-3). This step computes $\mathbf{b}_i^{(p)} = \sum_{k=1}^{g_0} b_{i,k}^p \mathbf{v}_k^{(p)}$ for each $\mathbf{b}_i \in \mathcal{B}$. For this, we first compute $\mathbf{v}_k^{(p)}$ for all $1 \leq k \leq g_0$, which can be done in $O(g_0 \mathsf{PowerMono}(pd^{(\max)}))$ bit operations. For each $1 \leq i \leq g$, it follows from $g_0 = O(t^{(\max)}(d^{(\max)})^r)$ that computing $\mathbf{b}_i^{(p)}$ is estimated to be done in $g_0 \cdot \mathsf{P}(p) = O\left(t^{(\max)}(d^{(\max)})^r \cdot \mathsf{P}(p)\right)$ arithmetic operations over K. Since we have $g = O(t^{(\max)}(d^{(\max)})^r)$, the complexity for Step (2-3) is estimated as

$$O\left(\left(t^{(\max)}(d^{(\max)})^r\right)^2 \cdot \mathsf{P}(p)\right) \tag{3.8}$$

arithmetic operations over K, plus

$$O\left(t^{(\max)}(d^{(\max)})^r \mathsf{PowerMono}(pd^{(\max)})\right) \tag{3.9}$$

bit operations.

We determine the complexity for Step (2-4). In this step, one first computes the right hand side of (3.3), which gives the representation (3.4). Note that each $g_{i,j}^p$ has been computed in Step (2-3). Recall that we have $g = \#\mathcal{B}^{(p)} = O(t^{(\max)}(d^{(\max)})^r)$. The representation matrix C_{r-q} is a $(t \times t)$ matrix over S with $t = t_{r-q}$, and its (j,k)-entry $h_{j,k}$ is homogeneous of degree $d_j p - d_k$ with $d_j = d_j^{(r-q)}$ and $d_k = d_k^{(r-q)}$. Thus the number of the non-zero terms of each $h_{j,k}$ is bounded by $\binom{d_j p - d_k + r}{r} = O((d^{(\max)}p)^r)$, whereas that of $g_{i,j}^p$ is bounded by $g_0 = O((d^{(\max)})^r)$. Moreover, $g_{i,j}^p h_{j,k}$ is a K-linear combination of monomials of negative degree $-d_k$. Hence, the computation of $g_{i,j}^p h_{j,k}$ for all $1 \le j \le t$ requires $O(t^{(\max)}(d^{(\max)})^{2r}p^r)$ arithmetic operations over K, plus $t^{(\max)}(d^{(\max)})^{2r}p^r\mathsf{MultMono}(d_k) = O(t^{(\max)}(d^{(\max)})^{2r}p^r\mathsf{MultMono}(d^{(\max)}))$ bit operations. For each $1 \le k \le g$, the k-th entry of (3.3) is given as the sum of $g_{i,j}^p h_{j,k}$ for all $1 \le j \le t$. Therefore, we have that the total cost of computing right hand side of (3.3) for all $1 \le i \le g$ is $O((t^{(\max)})^2(d^{(\max)})^{3r}p^r)$ arithmetic operations over K, plus $O((t^{(\max)})^2(d^{(\max)})^{3r}p^r\mathsf{MultMono}(d^{(\max)}))$ bit operations.

Next we solve the linear system (3.5) over K, where the coefficient matrices A and B have been computed in Step (1-2). The size of the coefficient matrix $\begin{pmatrix} A \\ B \end{pmatrix}$ (resp. the matrix $\begin{pmatrix} X & Y \end{pmatrix}$) is $g_0 = O(t^{(\max)}(d^{(\max)})^r)$ (resp. $g = O(t^{(\max)}(d^{(\max)})^r)$), and hence the computation can be done in $O\left((t^{(\max)})^4(d^{(\max)})^{4r}\right)$ arithmetic operations over K. As a consequence, the complexity for Step (2-4) is estimated as

$$O\left((t^{(\max)})^2(d^{(\max)})^{3r}p^r + \left(t^{(\max)}(d^{(\max)})^r\right)^4\right) \tag{3.10}$$

arithmetic operations over K, plus

$$O\left((t^{(\max)})^2(d^{(\max)})^{3r}p^r\mathsf{MultMono}(d^{(\max)})\right) \tag{3.11}$$

bit operations.

Considering (3.7)-(3.11), we have that the cost of Steps (1-2), (2-3) and (2-4) is upper-bounded by (3.6) plus the sum of the bit complexities (3.9) and (3.11). These bit complexities are estimated as follows: $\mathsf{PowerMono}(pd^{(\max)})$ (resp. $\mathsf{MultMono}(d^{(\max)})$) is upper-bounded by $O(r(\log(p)+\log(d^{(\max)}))) = \tilde{O}(1)$ (resp. $O(r\log(d^{(\max)})) = \tilde{O}(1)$), where we used that r is fixed. Thus the sum of the bit complexities (3.9) and (3.11) is $\tilde{O}\left((t^{(\max)})^2(d^{(\max)})^{3r}p^r\right)$, which is lower than the third term of (3.6). □

Putting $D := \max\{\dim_K H^r(\mathbf{P}^r, \mathcal{F}_i) : r-q-1 \leq i \leq r-q+1\}$, we can simply write (3.6) as (3.13) in Corollary 3.7 below. Note that we have $D = O(t^{(\max)}(d^{(\max)})^r)$ since

$$\dim_K H^r(\mathbf{P}^r, \mathcal{F}_i) = \sum_{j=1}^{t_i} \binom{d_j^{(i)} - 1}{r} = O\left(t^{(\max)}(d^{(\max)})^r\right) \tag{3.12}$$

for $r-q-1 \leq i \leq r-q+1$. The value D is also appropriate as an asymptotic parameter since it is determined from the values t_i and $d_j^{(i)}$ in (3.2).

Corollary 3.7. *The notation is same as in* Proposition 3.6. *We fix r and set $D := \max\{\dim_K H^r(\mathbf{P}^r, \mathcal{F}_i) \ ; \ r-q-1 \leq i \leq r-q+1\}$. Then the total arithmetic complexity of* Steps (1-2), (2-3) *and* (2-4) *of* Algorithm (I) *in* Subsection 3.2 *over K is*

$$\tilde{O}\left(D^4 + D^2\mathsf{P}(p) + D^3 p^r\right). \tag{3.13}$$

For $K = \mathbb{F}_p$, the arithmetic complexity is

$$\tilde{O}(D^4 + D^3 p^r), \tag{3.14}$$

and its binary complexity is

$$\tilde{O}(D^4\mathsf{M}^{\sim}(\log(p)) + D^3 p^r \mathsf{M}^{\sim}(\log(p))) \tag{3.15}$$

with $\mathsf{M}^{\sim}(n) := \mathsf{M}(n)\log(n)$, where $\mathsf{M}(n)$ denotes the time required to multiply two n-bit integers.

4. Algorithm specific to complete intersections

As in the previous section, let K be a perfect field of characteristic p, and let $S = K[x] = K[x_0, \ldots, x_r]$ denote the polynomial ring of $r+1$ variables over K. Let $X \subset \mathbf{P}^r = \mathrm{Proj}(S)$ be a projective scheme of dimension $q = \dim(X)$.

In this section, we give a specific method to compute a representation matrix for the Frobenius F^* with $q = \dim(X)$ when $X = V(f_1, \ldots, f_m)$ is a *complete intersection* defined by an *S-regular* sequence $(f_1, \ldots, f_m) \in S^m$ (in this case, we have $\dim(X) = r - m$). This method for $\dim(X) = 1$ (i.e., X is a curve) was already proposed in Kudo and Harashita (2017a), Appendix B, whereas it works also for $\dim(X) > 1$ and is viewed as a special case of Algorithm (I) in Subsection 3.2. After verifying these stronger arguments in the first subsection below, we shall give an algorithm in the second subsection. In particular, we prove that the entries of the representation matrix for F^* with respect to a suitable basis are coefficients in $(f_1 \cdots f_m)^{p-1}$.

4.1. The Frobenius action for complete intersections

The proposition below (Proposition 4.1) provides a method to compute F^* over complete intersections, and it includes Kudo and Harashita (2017a), Proposition B.2.2 for $\dim(X) = 1$. Both the propositions are proved in essentially the same way, whereas no proof is given in Kudo and Harashita (2017a). For the reader's convenience, we here give a complete proof which works also for $\dim(X) > 1$. Another reason for writing the proof here is to describe how the method is viewed as a special case of Algorithm (I) in Subsection 3.2.

Proposition 4.1. *Let K be a perfect field with $\mathrm{char}(K) = p > 0$. Let $f_1, \ldots, f_m$ be homogeneous polynomials with $d_{j_1 \ldots j_{m-1}} := \sum_{k=1}^{m-1} \deg(f_{j_k}) \leq r$ for all $1 \leq j_1 < \cdots < j_{m-1} \leq m$ such that $\gcd(f_i, f_j) = 1$ in $S := K[x] = K[x_0, \ldots, x_r]$ for $i \neq j$. Suppose that the sequence $(f_1, \ldots, f_m)$ is S-regular. Let $X = V(f_1, \ldots, f_m)$ be the projective scheme defined by the equations $f_1 = 0, \ldots, f_m = 0$ in $\mathbf{P}^r = \mathrm{Proj}(S)$, and $q := \dim(X) = r - m$. Write $(f_1 \cdots f_m)^{p-1} = \sum_\alpha c_\alpha x^\alpha$ with $x^\alpha = x_0^{\alpha_0} \cdots x_r^{\alpha_r}$ for $\alpha = (\alpha_0, \ldots, \alpha_r) \in (\mathbb{Z}_{\geq 0})^{r+1}$, and*

$$\Lambda \;:=\; \left\{ \beta \in (\mathbb{Z}_{<0})^{r+1} : |\beta| = -\sum_{j=1}^{m} \deg(f_j) \right\} = \{\beta^{(1)}, \ldots, \beta^{(g)}\}, \quad (4.1)$$

where $g := \#\Lambda$ and $|\beta| := \sum_{i=0}^{r} \beta_i$ for $\beta = (\beta_0, \ldots, \beta_r) \in (\mathbb{Z}_{<0})^{r+1}$. Then we have $g = \dim_K H^q(X, \mathcal{O}_X)$, and a representation matrix for F^* is given by

$$\begin{bmatrix} c_{-p\cdot\beta^{(1)}+\beta^{(1)}} & \cdots & c_{-p\cdot\beta^{(g)}+\beta^{(1)}} \\ \vdots & & \vdots \\ c_{-p\cdot\beta^{(1)}+\beta^{(g)}} & \cdots & c_{-p\cdot\beta^{(g)}+\beta^{(g)}} \end{bmatrix}, \quad (4.2)$$

where $p \cdot \beta := (p\beta_0, \ldots, p\beta_r)$.

Proof. Since $(f_1, \ldots, f_m)$ is S-regular, and so is the sequence $(f_1^p, \ldots, f_m^p)$, see e.g., Eisenbud (1995), Corollary 17.8. Hence both the minimal free resolutions $\mathbf{F}_\bullet$ and $\mathbf{F}_\bullet^{(p)}$ in Subsection 3.2 are canonically given as graded Koszul complexes (see Kudo and Harashita (2017a), Section B.2) of length m in the following way: For each $1 \leq i \leq m$, we set

$$\begin{aligned} \mathbf{F}_i &:= \bigoplus_{1 \leq j_1 < \cdots < j_i \leq m} S(-d_{j_1 \ldots j_i}) \mathbf{e}_{j_1 \ldots j_i}, \text{ and} \\ \mathbf{F}_i^{(p)} &:= \bigoplus_{1 \leq j_1 < \cdots < j_i \leq m} S(-p d_{j_1 \ldots j_i}) \mathbf{e}_{j_1 \ldots j_i}, \end{aligned}$$

with $d_{j_1 \ldots j_i} := \sum_{k=1}^{i} \deg(f_{j_k})$, and the graded homomorphisms $\varphi_i : \mathbf{F}_i \longrightarrow \mathbf{F}_{i-1}$ and $\varphi_i^{(p)} : \mathbf{F}_i^{(p)} \longrightarrow \mathbf{F}_{i-1}^{(p)}$ of degree zero are given by

$$\begin{aligned} \varphi_i(\mathbf{e}_{j_1 \ldots j_i}) &:= \sum_{k=1}^{i} (-1)^{k-1} f_{j_k} \mathbf{e}_{j_1 \ldots \hat{j_k} \ldots j_i}, \text{ and} \\ \varphi_i^{(p)}(\mathbf{e}_{j_1 \ldots j_i}) &:= \sum_{k=1}^{i} (-1)^{k-1} f_{j_k}^p \mathbf{e}_{j_1 \ldots \hat{j_k} \ldots j_i}, \end{aligned}$$

where the hat means to omit j_k. Moreover, $\psi_\bullet : \mathbf{F}_\bullet^{(p)} \longrightarrow \mathbf{F}_\bullet$ is constructed without using the division algorithm over S; we define each $\psi_i : \mathbf{F}_i^{(p)} \longrightarrow \mathbf{F}_i$ by

$$\psi_i(\mathbf{e}_{j_1 \dots j_i}) := (f_{j_1} \cdots f_{j_i})^{p-1} \mathbf{e}_{j_1 \dots j_i}.$$

In particular, ψ_{r-q} with $r-q=m$ is a map just multiplying an element in $S(-\sum_{j=1}^m d_j)$ by $(f_1 \cdots f_m)^{p-1}$. By the assumption $d_{j_1 \dots j_{m-1}} \leq r$ for all $1 \leq j_1 < \cdots < j_{m-1} \leq m$, we also have $\mathrm{Im}\left(H^r(\varphi^{\sim}_{r-q+1})\right) = \mathrm{Im}\left(H^r((\varphi^{(p)}_{r-q+1})^{\sim})\right) = 0$, and thus

$$\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet)) = \mathrm{Ker}\left(H^r(\varphi^{\sim}_{r-q})\right) = H^r(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(-\sum_{j=1}^m d_j)), \text{ and}$$

$$\mathrm{H}^{r-q}(H^r(\mathbf{P}^r, \mathcal{F}_\bullet^{(p)})) = \mathrm{Ker}\left(H^r((\varphi^{(p)}_{r-q})^{\sim})\right) = H^r(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(-\sum_{j=1}^m p d_j))$$

in the diagram in Lemma 3.2. It follows from Theorem 2.1 that the basis of $H^r(\mathbf{P}^r, \mathcal{O}_{\mathbf{P}^r}(-\sum_{j=1}^m d_j))$ is given by $\mathcal{B} = \{x^\beta : \beta \in \Lambda\}$. For each basis element $x^{\beta^{(i)}} \in \mathcal{B}$ with $\beta^{(i)} \in \Lambda$, its image by the composite map of the right column of the diagram in Lemma 3.2 is computed as

$$\begin{aligned}(f_1 \cdots f_m)^{p-1} \cdot \left(x^{\beta^{(i)}}\right)^p &= (f_1 \cdots f_m)^{p-1} \cdot x^{p \cdot \beta^{(i)}} = \sum_\alpha c_\alpha x^{\alpha + p \cdot \beta^{(i)}} \\ &= \sum_{j=1}^g c_{-p \cdot \beta^{(i)} + \beta^{(j)}} x^{\beta^{(j)}}.\end{aligned}$$

Hence our claim holds. □

Proposition 4.1 gives a simplification of Algorithm (I) in Subsection 3.2 if the input $(f_1, \dots, f_m)$ is S-regular and if $q = \dim(X) = r - m$; to compute the representation matrix for F^*, we not necessarily compute any free resolution, but only certain coefficients in $(f_1 \cdots f_m)^{p-1}$. Moreover this method is viewed as a generalization of a standard method to compute F^* with $q=1$ for elliptic curves, see Example 2.3 in this paper or Hartshorne (1977), Chapter IV.

4.2. Algorithm and Complexity

With the same notation as in Proposition 4.1, we here write down an efficient algorithm specific to complete intersections. Thanks to Proposition 4.1, Algorithm (I) in Subsection 3.2 is quite simply written as follows:

Algorithm (II) (algorithm for complete intersections). Given an S-regular sequence $(f_1, \dots, f_m)$ and the characteristic p, this algorithm computes a representation matrix for F^* on $H^q(X, \mathcal{O}_X)$ with $q = r - m = \dim(X)$.

1. Compute the coefficients $c_{-p \cdot \beta^{(i)} + \beta^{(j)}}$ for $1 \leq i, j \leq g$ in $(f_1 \cdots f_m)^{p-1}$.
2. Output the matrix (4.2).

Complexity. The complexity is upper-bounded by the cost of computing the multiple $(f_1 \cdots f_m)^{p-1}$. The cost of computing $(f_1 \cdots f_m)^{p-1}$ heavily depends on one's choice of algorithms for computing the multiplication and the power computation over the multivariate polynomial ring $K[x_0, \ldots, x_r]$; for a fixed r, the complexity can be upper-bounded in polynomial time with respect to $\max_{1 \leq j \leq m}(\deg(f_j))$ and p, see e.g., Horowitz (1973), Theorem 3.1.

Open problems. Improving Algorithm (II) and its complexity bound is an open problem since it is not necessary to compute all the non-zero coefficients in $(f_1 \cdots f_m)^{p-1}$, but only the g^2 coefficients $c_{-p \cdot \beta^{(i)} + \beta^{(j)}}$ for $1 \leq i, j \leq g$.

One possible way is to generalize a method adopted in computing the Cartier operator for a hyperelliptic curve $y^2 = f(x)$, where $f(x)$ is a univariate polynomial in x over K of degree $2g + 1$. Recall from Section 1 that the entries of the matrix representing the Cartier operator are the x^{ip-j}-coefficients in $h := f^{(p-1)/2}$ for $1 \leq i, j \leq g$. In this case, computing the Cartier operator is reduced into multiplying matrices over the ring $\mathbb{Z}_p$ of p-adic integers, see e.g., Bostan et al. (2003), p. 52 for more details. The reduction is based on the following fact: The coefficients in h satisfy a linear recurrent sequence of order $2g + 1$. From this, a linear recurrence with the coefficients in $(f_1 \cdots f_m)^{p-1}$, if it exists, may be the key to the generalization.

Here we list the problems described above;

Problem 4.2. *Find an algorithm to efficiently compute* $c_{-p \cdot \beta^{(i)} + \beta^{(j)}}$ *for* $1 \leq i, j \leq g$ *in* $(f_1 \cdots f_m)^{p-1}$.

Problem 4.3. *Find a linear recurrence with the coefficients in* $(f_1 \cdots f_m)^{p-1}$.

5. Computational examples and experimental results

This section shows computational examples and experimental results obtained by our implementation over Magma (Bosma et al. (1997), Cannon et al. (2016)). We implemented the main algorithms (Algorithms (I) and (II)) over Magma V2.24-5 in its 32-bit version on a laptop with Windows OS, a 2.60 GHz Inter Core i5-4210M processor and 8.00 GB memory.

5.1. Examples

Example 5.1. Let K be a perfect field of characteristic $p > 2$. Put

$$\begin{aligned} f &:= 5vz - 2wx - 3wy + wz, \\ g &:= 10v^2 + 5wv - 5w^2 + 4x^2 - 12xy + 2xy - 2y^2 - 35yz - 12z^2, \\ h &:= 15v^2 - 5wv + 5w^2 + 8x^2 - 12xy - 14xz - 11y^2 - 3yz + 15z^2, \end{aligned}$$

and $C := V(I) \subset \mathbf{P}^4$ with $S = K[x, y, z, v, w]$ and $I = \langle f, g, h \rangle_S$. The non-singular curve C is the (classical) modular curve of level 67, say $C = X_0(67)$. For defining equations for modular curves, see e.g., Galbraith (1996). In the following, we compute the representation matrix for the Frobenius on the first

cohomology group $H^1(C, \mathcal{O}_C)$ for the case of $p = 3$. In this case, we have the following commutative diagram:

$$\begin{array}{ccccccccccccc}
0 & \xrightarrow{\varphi_4^{(p)}} & S(-6p) & \xrightarrow{\varphi_3^{(p)}} & \bigoplus_{j=1}^{3} S(-4p) & \xrightarrow{\varphi_2^{(p)}} & \bigoplus_{j=1}^{3} S(-2p) & \xrightarrow{\varphi_1^{(p)}} & S & \xrightarrow{\varphi_0^{(p)}} & S/I_p & \longrightarrow & 0 \\
 & & \downarrow{\scriptstyle \psi_3} & & \downarrow{\scriptstyle \psi_2} & & \downarrow{\scriptstyle \psi_1} & & \downarrow{\scriptstyle \psi_0} & & \downarrow{\scriptstyle \psi} & & \\
0 & \xrightarrow{\varphi_4} & S(-6) & \xrightarrow{\varphi_3} & \bigoplus_{j=1}^{3} S(-4) & \xrightarrow{\varphi_2} & \bigoplus_{j=1}^{3} S(-2) & \xrightarrow{\varphi_1} & S & \xrightarrow{\varphi_0} & S/I & \longrightarrow & 0
\end{array}$$

For the representation matrices of the above homomorphisms, see the text files on the web page of the author (Kudo (2020)). The first cohomology group $H^1(C, \mathcal{O}_C) \cong H^4(\mathbf{P}^4, \mathcal{O}_{\mathbf{P}^4}(-6))$ has a basis

$$\left\{ \frac{1}{x^2yzvw}, \frac{1}{xy^2zvw}, \frac{1}{xyz^2vw}, \frac{1}{xyzv^2w}, \frac{1}{xyzvw^2} \right\},$$

which indicates that C is a curve of genus 5. From the output of our program, the representation matrix for F^* with $q = 1$ is

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 2 & 0 & 2 & 0 & 0 \\ 0 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix},$$

and its rank is equal to 3. The Eigen polynomial is $a^5 + a^4 + a^3$, where a is an indeterminate.

Remark 5.2. In Example 5.1, the tuple (f, g, h) is an S-regular sequence with $\deg(f) = \deg(g) = \deg(h) = 2$.

Example 5.3. Let K be a perfect field of characteristic $p > 0$. Put

$$\begin{aligned} f_1 &:= y^2 + (-x_3 - x_1 - x_0)y + 2x_3x_2 + 3x_1^2 - 2x_1x_0 + 2x_0^2, \\ f_2 &:= x_1^2 - x_0x_2, \quad f_3 := x_2^2 - x_1x_3, \quad f_4 := x_3x_0 - x_2x_1, \end{aligned}$$

and $C := V(f_1, f_2, f_3, f_4) \subset \mathbf{P}^4$ with $S = K[x_0, x_1, x_2, x_3, y]$. The non-singular curve C is a normalization of the modular curve $X_0(23)$, which is a hyperelliptic curve of genus 2 given as an affine model in Bruin and Najman (2015). For a method of the normalization of hyperelliptic curves, see Galbraith (2012), Chapter 10. In what follows, we compute the representation matrix for the Frobenius on $H^1(C, \mathcal{O}_C)$ for the case of $p = 5$. By a way similar to Example 5.1, we can compute a basis of $H^1(C, \mathcal{O}_C)$. (For more information of the computation, see the text files on the web page of the author (Kudo (2020)).) The output basis of $H^1(C, \mathcal{O}_C)$ is

$$\left\{ \left(\frac{1}{x_0x_1x_2x_3y}, \quad 0 \right), \left(0, \quad \frac{1}{x_0x_1x_2x_3y} \right) \right\},$$

which indicates that C is a curve of genus 2. From the output of our program, the representation matrix for F^* with $q = 1$ is

$$\begin{pmatrix} 2 & 2 \\ 3 & 1 \end{pmatrix},$$

and it has full-rank. The Eigen polynomial is $a^2 + 2a + 1$, where a is an indeterminate.

Different from Examples 5.1 and 5.3, we shall give an example which is not a curve in Example 5.4:

Example 5.4. Let K be a perfect field of characteristic $p > 0$. We consider a *Horrocks-Mumford surface*, which is an abelian variety X embedded into $\mathbf{P}^4$ as the (non-complete) intersection of 3 quintics and 15 sextics (cf. Manolache (1988)). It was proved in Theorem 5.2 of Horrocks and Mumford (1973) that any abelian surface in $\mathbf{P}^4$ is projectively equivalent to one of Horrocks-Mumford surfaces. Note that X has degree 10, see also Section 7.1 of Liedtke (2013) for other invariants of (general) abelian surfaces over (perfect) fields of positive characteristic.

We randomly choose 3 quintics and 15 sextics defining X by using Magma's built-in functions `RandomAbelianSurface_d10g6` and `DefiningPolynomials`. For reasons of space, the chosen defining equations are omitted here, but they are written in a text file (`HM-surface_example1.txt`) which is available at the author's web page Kudo (2020). For this (specific) chosen X, we demonstrate Algorithm (I) in Subsection 3.2 to compute the representation matrix for the Frobenius F^* on $H^1(X, \mathcal{O}_X)$ in the case of $p = 3$. Note that representation matrices for φ_i and ψ_i, the basis $\{\mathbf{b}_1, \mathbf{b}_2\}$ and the matrix Y below depend on the choice of defining equations. We also use the same notation as in Section 3.

In Step (1-1) of Algorithm (I), the minimal free resolution $\mathbf{F}_\bullet$ is computed as follows:

$$0 \to S(-10)^2 \xrightarrow{\varphi_4} S(-8)^{20} \xrightarrow{\varphi_3} S(-7)^{35} \xrightarrow{\varphi_2} S(-5)^3 \oplus S(-6)^{15} \xrightarrow{\varphi_1} S \xrightarrow{\varphi_0} S/I \to 0,$$

where $S := K[x, y, z, w, v]$ and where we simply denote by M^n the direct sum $\bigoplus_{i=1}^n M$ for a (graded) module M and an integer $n \geq 1$. Step (1-2) computes a basis of $H^1(X, \mathcal{O}_X)$ via the isomorphy $H^1(X, \mathcal{O}_X) \cong \mathrm{H}^3(H^4(\mathbf{P}^4, \mathcal{F}_\bullet)) = \mathrm{Ker}\left(H^4(\varphi_3^\sim)\right) / \mathrm{Im}\left(H^4(\varphi_4^\sim)\right)$, where $\mathrm{Ker}\left(H^4(\varphi_3^\sim)\right)$ and $\mathrm{Im}\left(H^4(\varphi_4^\sim)\right)$ are subspaces of the 700-dimensional space $H^4(\mathbf{P}^4, \mathcal{F}_3) \cong H^4(\mathbf{P}^4, \mathcal{O}_{\mathbf{P}^4}(-8))^{\oplus 20}$ with the basis

$$\mathcal{V} = \left\{ x^{\alpha_1} y^{\alpha_2} z^{\alpha_3} w^{\alpha_4} v^{\alpha_5} \mathbf{e}_j : 1 \leq j \leq 20 \text{ and } \alpha_k \in \mathbb{Z}_{<0} \text{ with } \textstyle\sum_{k=1}^5 \alpha_k = -8 \right\}.$$

Similarly to $H^4(\mathbf{P}^4, \mathcal{F}_3)$, we have that $H^4(\mathbf{P}^4, \mathcal{F}_4)$ and $H^4(\mathbf{P}^4, \mathcal{F}_2)$ are of dimension 252 and 525 respectively, and we can compute their bases explicitly. With these bases together with representation matrices for φ_4 and φ_3, we can compute a basis of $H^1(X, \mathcal{O}_X)$. For more information of the computation, see

the text file `result_code_HM-surface_example1.txt` on the web page of the author (Kudo (2020)). The output basis $\mathcal{B}$ for $H^1(X, \mathcal{O}_X)$ consists of two elements represented explicitly by the following two vectors:

$$\begin{aligned}\mathbf{b}_1 &= \frac{2}{x^4yzwv}\mathbf{e}_1 + \frac{2}{x^3y^2zwv}\mathbf{e}_2 + \frac{2}{x^2y^3zwv}\mathbf{e}_3 + \frac{2}{xy^4zwv}\mathbf{e}_4 + \frac{2}{x^3yz^2wv}\mathbf{e}_5 \\ &\quad + \frac{2}{xy^3z^2wv}\mathbf{e}_6 + \frac{2}{x^2yz^3wv}\mathbf{e}_7 + \frac{2}{xy^2z^3wv}\mathbf{e}_8 + \frac{2}{xyz^4wv}\mathbf{e}_9, \\ \mathbf{b}_2 &= \frac{2}{xy^3zw^2v}\mathbf{e}_1 + \frac{2}{x^3yzw^2v}\mathbf{e}_4 + \frac{1}{xyzw^4v}\mathbf{e}_6 + \frac{1}{x^4yzwv}\mathbf{e}_8 + \frac{2}{x^2yzw^3v}\mathbf{e}_9 \\ &\quad + \frac{2}{x^3y^2zwv}\mathbf{e}_{10} + \frac{2}{x^2y^3zwv}\mathbf{e}_{11} + \frac{2}{xy^4zwv}\mathbf{e}_{12} + \frac{2}{xy^2zw^3v}\mathbf{e}_{13}.\end{aligned}$$

In Step (2-1) (resp. (2-3)), the resolution $\mathbf{F}_\bullet^{(p)}$ for S/I_p (resp. the image $\mathcal{B}^{(p)}$) is computed from $\mathbf{F}_\bullet$ (resp. $\mathcal{B}$), by considering the p-th power operation. By the matrix representing $\psi_3 : S(-8p)^{20} \to S(-8)^{20}$ computed in Step (2-2), we compute the representation matrix Y for F^* on $H^1(X, \mathcal{O}_X)$ in Step (2-4) with simple linear algebra. From the output of our program, Y is a rank 2-matrix for this sample (this means that X is *ordinary*).

We also executed Algorithm (I) in Subsection 3.2 for some other samples generated by `RandomAbelianSurface_d10g6`. The computational results show that both the cases of $\mathrm{rank}(Y) = 1$ and 0 also occur, where X with $\mathrm{rank}(Y) = 0$ is called *superspecial* (in fact, supersingular, see Section 7.1 of Liedtke (2013)). See `HM-surface_example2.txt` and `HM-surface_example3.txt` for the defining equations of these samples at the web page Kudo (2020). It would be an interesting problem to determine the (non-)existence of a superspecial Horrocks-Mumford surface for large p.

5.2. Experimental results

To confirm practical time performance of our algorithm, we compute the representation matrix for the Frobenius F^* on the first cohomology group of $X_0(23)$ for $3 \leq p \leq 17$. Table 1 shows our experimental results for $X_0(23)$ of Example 5.3. We use the same notation as in Section 3. Note that $r = 4$ and $D = 2$ are fixed in our experiments.

Time performance: For each $3 \leq p \leq 17$, the computation of free resolutions and lifting homomorphisms (Steps (1-1), (2-1) and (2-2)) is clearly dominant; in particular, Step (2-2) takes so much time to terminate, e.g., 85.074 seconds for $p = 17$. Unlike the steps (1-1), (2-1) and (2-2), it only takes a few seconds to proceed with Steps (1-2), (2-3) and (2-4). From this, we see that once one gets free resolutions and lifting homomorphisms, one may efficiently compute the matrix representing the Frobenius F^*. On the other hand, for examining that Steps (1-2), (2-3) and (2-4) behave in the complexity estimated in (3.14) of Corollary 3.7 for large p, it is necessary to improve the computation of free resolutions and lifting homomorphisms (Steps (1-1), (2-1) and (2-2)).

Table 1: Experimental results to examine time performance on Algorithm (I) for $X_0(23)$ in $\mathbf{P}^4$ with $r = 4$. For each $3 \leq p \leq 17$, the Frobenius F^* has full-rank ($= 2$), which means that $X_0(23)$ is an ordinary curve.

p	D	Eigen polynomial	Total time for Steps (1-1), (2-1) and (2-2)	Total time for Steps (1-2), (2-3) and (2-4)
3	2	$a^2 + 1$	0.118	0.037
5	2	$a^2 + 2a + 1$	0.445	0.046
7	2	$a^2 + 5a + 3$	1.796	0.089
11	2	$a^2 + 6a + 4$	12.106	0.516
13	2	$a^2 + 7a + 9$	39.634	1.316
17	2	$a^2 + 11a + 4$	85.129	3.142

Correctness: An affine model of $X_0(23)$ given in Bruin and Najman (2015) is

$$y^2 + (-x^3 - x - 1)y = -2x^5 - 3x^2 + 2x - 2,$$

ans its genus is 2. With Yui's method (Yui (1978)), one can compute the rank and the Eigen polynomial of the Cartier operator, which is dual to the Frobenius operator F^*. The rank and the Eigen polynomial computed by Yui's method coincide with those in Table 1 for each $3 \leq p \leq 17$.

6. Concluding remarks

We proposed two *explicit* algorithms to compute the representation matrix for the Frobenius F^* on the cohomology groups $H^q(X, \mathcal{O}_X)$. The first algorithm (main algorithm) works for arbitrary projective schemes, and is based on the following three techniques: (i) Decompose F^* into two computable maps between cohomology groups over $\mathbf{P}^r$. (ii) Compute a matrix corresponding to one of the two maps, where its entries are homogeneous polynomials in S. (iii) Multiplying each basis element by the matrix computed in (ii) together with linear algebra techniques, we represent the each image element by F^* as a K-linear combination of the same basis. The second algorithm is a simplified version of the first one, and is specific to complete intersections. Our algorithms also provide effective methods to classify algebraic varieties in positive characteristic, e.g., whether curves are ordinary or superspecial.

However, the first algorithm works well under the assumption that one has computed free resolutions and lifting homomorphisms. To make the algorithm more efficient, we have to improve the efficiency of computing free resolutions and lifting homomorphisms, or design a new method to efficiently compute them. The second algorithm has room for improvement if we have a method to efficiently extract specific coefficients in a power of a multivariate polynomial. Constructing such a method and improving the second algorithm are open problems.

Acknowledgement

The author thanks the anonymous referees for their comments and suggestions, which have helped the author significantly improve the paper.

The contents of this paper was presented at Effective Methods in Algebraic Geometry (MEGA 2019). The author also thanks the anonymous referees assigned by the organization of this conference for their comments and suggestions. All of them are taken into account for improving the presentation of the paper.

The author is also very grateful to Shushi Harashita, Masaya Yasuda and Kazuhiro Yokoyama for many helpful comments, corrections, and suggestions on this work. This work was supported by JSPS Grant-in-Aid for Research Activity Startup 18H05836 and 19K21026, and JSPS Grant-in-Aid for Young Scientists 20K14301.

References

Baker, M.: *Cartier points on curves*, International Mathematics Research Notices, Volume **2000**, Issue 7, pp. 353–370, 2000.

Bayer, D. and Stillman, M.: *On the complexity of computing syzygies*, Journal of Symbolic Computation, **6**, pp. 135–147, 1988.

Bosma, W., Cannon, J. and Playoust, C.: *The Magma algebra system. I. The user language*, Journal of Symbolic Computation, **24**, pp. 235–265, 1997.

Bostan, A., Gaudry, P. and Schost, É.: *Linear recurrences with polynomial coefficients and computation of the Cartier-Manin operator on hyperelliptic curves*, In: Mullen G. L., Poli A., Stichtenoth H. (eds) Finite Fields and Applications. Fq 2003, Lecture Notes in Computer Science, **2948**, pp. 40–58, Springer, Berlin, Heidelberg.

Bruin, P. and Najman, F.: *Hyperelliptic modular curves $X_0(n)$ and isogenies of elliptic curves of quadratic fields*, LMS J. Compute. Math., **18** (1), pp. 578–602, 2015.

Cannon, J., et al.: *Magma A Computer Algebra System*, School of Mathematics and Statistics, University of Sydney, 2016.
`http://magma.maths.usyd.edu.au/magma/`

Capani, A., De Dominicis, G., Niesi, G. and Robbiano, L.: *Computing minimal finite free resolutions*, Journal of Pure and Applied Algebra, **117-118**, pp. 105–117, 1997.

Castryck, W., Decru, T. and Smith, B.: *Hash functions from superspecial genus-2 curves using Richelot isogenies*, to appear in: Proceedings of Number-Theoretic Methods in Cryptology 2019 (NutMiC 2019), arXiv: 1903.06451 [cs.CR], 2019.

Celik, T. O., Elias, Y., Gunes, B., Newton, R., Ozman, E., Pries, R. and Thomas, L.: *Non-Ordinary curves with a Prym variety of low p-rank*, In: Bouw I., Ozman E., Johnson-Leung J., Newton R. (eds), Women in Numbers Europe II. Association for Women in Mathematics Series, vol. **11**. Springer, Cham, 2018.

Cox D., Little J., O ' Shea D. (1998) Free Resolutions. In: Using Algebraic Geometry. GTM **185**, Springer, New York, NY.

Decker, W. and Eisenbud, D.: *Sheaf Algorithm Using the Exterior Algebra*, pp. 215–249, In: Eisenbud et al. (2002), 2002.

Decker, W. and Lossen, C.: *Computing in Algebraic Geometry, A Quick Start using SINGULAR*, ACM **16**, Springer, 2000.

Eisenbud, D.: *Commutative Algebra: With a View Toward Algebraic Geometry*, GTM **150**, Springer, 1995.

Eisenbud, D.: Chapter 8: Computing cohomology, pp. 219–226, In: Vasconcelos (1998), 1998.

Eisenbud, D., Grayson, D. R., Stillman, M. and Sturmfels, B. eds.: *Computations in Algebraic Geometry with Macaulay2*, Springer-Verlag, 2002.

Eisenbud, D., Fløystad, G. and Schreyer F.-O.: *Sheaf Cohomology and Free Resolutions over Exterior Algebras*, Trans. Amer. Math. Soc., **355**, no. 11, pp. 4397–4426, 2003.

Erocal, B., Motsak, O., Schreyer, F.-O., Steenpass, A.: *Refined Algorithms to Compute Syzygies*, J. Symb. Comput. **74** (2016), 308-327.

Galbraith, S. D.: *Equations for modular curves*, Doctoral Thesis, Oxford University, 1996.

Galbraith, S. D.: *Mathematics in Public Key Cryptography*, Cambridge University Press, 2012.

González, J.: *Hasse-Witt matrices for the Fermat curves of prime degree*, Tohoku Math. J. (2) **49** (1997), no. 2, pp. 149–163. MR 1447179 (98b:11064)

Greuel, G.-M. and Pfister, G.: *A Singular Introduction to Commutative Algebra*, Second edition, Springer (2007).

Hartshorne, R.: *Algebraic Geometry*, GTM **52**, Springer-Verlag, 1977.

Harvey, D. and Sutherland, A. V.: *Computing Hasse-Witt matrices of hyperelliptic curves in average polynomial time*, LMS Journal of Computation and Mathematics, **17**, pp. 257–273, 2014.

Horowitz, E.: *The Efficient Calculation of Powers of Polynomials*, Journal of Computer and System Science, **7**, pp. 469–480, 1973.

Horrocks, G. and Mumford, D.: *A rank* 2 *vector bundle on* $\mathbb{P}^4$ *with* 15,000 *symmetries*, Topology, **12** (1), pp. 63–81, 1973.

Komoto, H., Kozaki, S. and Matsuo, K.: *Improvements in the computation of the Hasse-Witt matrix*, JSIAM Letters Vol. **2** (2010), pp. 17–20, 2010.

Kreuzer, M. and Robbiano, L.: *Computational Commutative Algebra 2*, Springer–Verlag Berlin Heidelberg, 2005.

Kudo, M.: *Analysis of an algorithm to compute the cohomology groups of coherent sheaves and its applications*, Japan Journal of Industrial and Applied Mathematics, Vol. **34**, Issue 1, pp. 1–40, 2017.

Kudo, M. and Harashita, S.: *Superspecial curves of genus* 4 *in small characteristic*, Finite Fields and Their Applications, **45**, pp. 131–169, 2017.

Kudo, M. and Harashita, S.: *Computational approach to enumerate non-hyperelliptic superspecial curves of genus 4*, Tokyo Journal of Mathematics, in press, 2020 (preprint version: *Enumerating superspecial curves of genus* 4 *over prime fields*, arXiv: 1702.05313 [math.AG], 2017).

Kudo, M. and Harashita, S.: *Enumerating Superspecial Curves of Genus* 4 *over Prime Fields* (abstract version of Kudo and Harashita (2020)), In: Proceedings of The Tenth International Workshop on Coding and Cryptography 2017 (WCC2017), September 18-22, 2017, Saint-Petersburg, Russia, `http://wcc2017.suai.ru/proceedings.html`

Kunz, E.: *Characterization of regular local rings of characteristic p*, American Journal of Mathematics, **41**, pp. 772–784, 1969.

La Scala, R.: *Computing minimal free resolutions of right modules over non-commutative algebras*, Journal of Algebra, **478**, 458–483, 2017.

La Scala, R. and Stillman, M.: *Strategies for Computing Minimal Free Resolutions*, J. Symb. Comput. **26**(4) (1998), pp. 409–431.

Liedtke C.: *Algebraic Surfaces in Positive Characteristic*, In: Bogomolov F., Hassett B., Tschinkel Y. (eds) Birational Geometry, Rational Curves, and Arithmetic. Springer, New York, NY, 2013.

Manin, J. I.: *The Hasse-Witt matrix of an algebraic curve*, AMS Translations, Series 2 **45** (1965), pp. 245–264, (originally published in Izv. Akad. Nauk SSSR Ser. Mat. **25** (1961) 153–172). MR 0124324 (23 #A1638)

Manolache, N.: *Syzygies of abelian surfaces embedded in* $\mathbf{P}^4(\mathbf{C})$, J. Reine Angew. Math., **384**, (1988) 180–191.

Maruyama, M.: *Gröbner Bases and their Application* (in Japanese). Kyoritsu Publisher, Tokyo (2002).

Miler, C.: *The Frobenius endomorphism and homological dimensions*, arXiv, math. AC: 0301208v3, 2003.

Schreyer, F.-O.: Die Berechnung von Syzygien mit dem verallgemeinerten Weierstrasschen Divisionssatz. Diplomarbeit, Hamburg (1980).

Schreyer, F.-O.: A Standard Basis Approach to Syzygies of Canonical Curves. J. Reine Angew. Math. **421** (1991), 83–123.

Smith, G. G.: *Computing Global Extension Module*, Journal of Symbolic Computation, **29**, pp. 729–746, 2000.

Tuitman, J.: *Counting points on curves using a map to* $\mathbf{P}^1$*, II*, arXiv: 1412.7217 [math.NT], 2014.

Vasconcelos, W.: *Computational Methods in Commutative Algebra and Algebraic Geometry*, Algorithms and Computation in Mathematics, **2**, Springer, 1998.

Stöhr, K.-O. and Voloch, J. F.: *A formula for the Cartier operator on plane algebraic curves*, J. Reine Angew. Math. **377** (1987), 49–64.

Yui, N.: *On the Jacobian varieties of hyperelliptic curves over fields of characterisctic* $p > 2$. Journal of algebra, **52**, pp. 378–410, 1978.

Computation programs and log files for the paper "Computing representation matrices for the Frobenius on cohomology groups", available on the web page `https://sites.google.com/view/m-kudo-official-website/english/code/comp_frobenius?authuser=0`