MATHEMATICS OF COMPUTATION
Volume 82, Number 283, July 2013, Pages 1711–1756
S 0025-5718(2012)02650-0
Article electronically published on September 14, 2012

COMPUTING IN PICARD GROUPS OF PROJECTIVE CURVES OVER FINITE FIELDS

PETER BRUIN

Abstract. We give algorithms for computing with divisors on projective curves over finite fields, and with their Jacobians, using the algorithmic representation of projective curves developed by Khuri-Makdisi. We show that various desirable operations can be performed efficiently in this setting: decomposing divisors into prime divisors; computing pull-backs and push-forwards of divisors under finite morphisms, and hence Picard and Albanese maps on Jacobians; generating uniformly random divisors and points on Jacobians; computing Frobenius maps; and finding a basis for the l-torsion of the Picard group for prime numbers l different from the characteristic of the base field.

Introduction

Let X be a complete, smooth, geometrically connected curve of genus g over a field k. We fix a line bundle $\mathcal{L}$ on X of degree at least $2g+1$. Then X can be represented by means of the finite-dimensional k-vector spaces of global sections of the first few tensor powers of $\mathcal{L}$ and the multiplication maps between them; effective divisors on X can be represented as linear subspaces of these k-vector spaces. (These statements will be made precise in Section 2.) Using this representation of X and of divisors on it, Khuri-Makdisi [12] has developed algorithms for computing with divisors and elements of the Picard group. Taking advantage of some improvements to this basic idea, described in [13], his algorithms are currently the fastest known algorithms for general curves, asymptotically for increasing genus and measured in operations in k. A notable feature of this framework is that equations for X play a negligible role.

In the present article, we concentrate on the case where the field k is finite. Theorems A and B below summarise our main results. We assume that curves and divisors are represented as in §§2.1 and 2.2 below, respectively. We write $\mathcal{L}_X$ for the fixed line bundle on X. If D is an effective divisor on X, we may represent D as the k-vector space $\Gamma(X, \mathcal{L}_X^{\otimes i}(-D))$ (see §2.2), where i is bounded by some fixed linear function of $(\deg D)/(\deg \mathcal{L}_X)$. For simplicity, we assume in both theorems that this convention is respected. The curve is given to the algorithms in the form of a certain finite commutative k-algebra $S_X^{(h)}$, defined in §2.1 below; we implicitly assume that h is large enough with respect to the degrees of the divisors involved.

Theorem A. *There exist probabilistic algorithms that solve the following problems for projective curves X over a finite field k, with expected running time as indicated.*

Received by the editor February 4, 2011 and, in revised form, November 2, 2011.
2010 *Mathematics Subject Classification.* 11G20, 11Y16, 14Q05.
This paper evolved from one of the chapters of the author's thesis [*Modular curves, Arakelov theory, algorithmic applications*, Proefschrift, Universiteit Leiden, 2010], the research for which was supported by the Netherlands Organisation for Scientific Research (NWO).

(1) *Given an effective divisor D on X, compute the decomposition of D into prime divisors as a list of pairs (P, m_P), where P is a prime divisor and m_P is the multiplicity of P in D, in time polynomial in* $\deg \mathcal{L}_X$, $\deg D$ *and* $\log \#k$ *(Algorithm* 2.4*).*
(2) *Given a finite extension k' of k and an effective divisor D on $X_{k'}$, compute the image of D under the Frobenius map over k in time polynomial in* $\deg \mathcal{L}_X$, $\deg D$, $[k' : k]$ *and* $\log \#k$ *(Algorithm* 3.1*).*
(3) *Given the zeta function of X and a non-negative integer d such that the set of effective divisors of degree d on X is non-empty, generate a uniformly random element of this set in time polynomial in* $\deg \mathcal{L}_X$, d *and* $\log \#k$ *(Algorithm* 3.5*).*
(4) *Given a finite extension k' of k and an element $x \in \operatorname{Pic}^0 X_{k'}$, compute the image of x under the Frobenius map over k in time polynomial in* $\deg \mathcal{L}_X$, $[k' : k]$ *and* $\log \#k$ *(Algorithm* 3.6*).*
(5) *Given the zeta function of X, generate a uniformly random element of* $\operatorname{Pic} X$ *in time polynomial in* $\deg \mathcal{L}_X$ *and* $\log \#k$ *(Algorithm* 3.7*).*
(6) *Given a positive integer n dividing $\#k^\times$ and elements $x, y \in \operatorname{Pic}^0 X$ with $ny = 0$, compute the element $[x, y]_n \in \mu_n(k)$, where $\mu_n(k)$ denotes the group of n-th roots of unity in k and*

$$[\ ,\ ]_n \colon (\operatorname{Pic} X)/n \operatorname{Pic} X \times (\operatorname{Pic} X)[n] \longrightarrow \mu_n(k)$$

is the Frey–Rück pairing, in time polynomial in $\deg \mathcal{L}_X$, $\log n$ *and* $\log \#k$ *(Algorithm* 3.9*).*
(7) *Given the zeta function of X and a prime number l different from the characteristic of k, compute an $\mathbf{F}_l$-basis for* $(\operatorname{Pic} X)[l]$ *in time polynomial in* $\deg \mathcal{L}_X$, l *and* $\log \#k$ *(Algorithm* 3.12*).*

Theorem B summarises our main results about finite morphisms between projective curves. Such morphisms are assumed to be represented as in §2.5 below; in particular, if $f \colon X \to Y$ is such a morphism, then $\mathcal{L}_X$ is isomorphic to $f^*\mathcal{L}_Y$. To explain the running times in this theorem, we note that $\deg \mathcal{L}_X = \deg \mathcal{L}_Y \cdot \deg f$. The restriction to finite fields is not essential here; see the corresponding algorithms for more general statements.

Theorem B. *There exist probabilistic algorithms that solve the following problems for morphisms $f \colon X \to Y$ between projective curves over a finite field k, with expected running time as indicated.*

(1) *Given an effective divisor D on X, compute the image $f(D)$ on Y in time polynomial in* $\deg \mathcal{L}_X$, $\deg D$ *and* $\log \#k$ *(Algorithm* 2.5*).*
(2) *Given an effective divisor E on Y, compute the pull-back f^*E in time polynomial in* $\deg \mathcal{L}_X$, $\deg E$ *and* $\log \#k$ *(Algorithm* 2.6*).*
(3) *Given an effective divisor D on X, compute the push-forward f_*D in time polynomial in* $\deg \mathcal{L}_X$, $\deg D$ *and* $\log \#k$ *(Algorithm* 2.7*).*
(4) *Given an element $y \in \operatorname{Pic} Y$, compute* $(\operatorname{Pic} f)(y)$ *in time polynomial in* $\deg \mathcal{L}_X$ *and* $\log \#k$ *(Algorithm* 2.14*).*
(5) *Given an element $x \in \operatorname{Pic} X$ and a rational point $O \in X(k)$, compute* $(\operatorname{Alb} f)(x)$ *in time polynomial in* $\deg \mathcal{L}_X$ *and* $\log \#k$ *(Algorithm* 2.15*).*

The paper is organised as follows. In the preliminary Section 1 we consider some computational problems related to finite algebras over a field; these are needed

in the other two sections. In Section 2 we recall Khuri-Makdisi's algorithms for projective curves over an arbitrary base field k, and we describe a number of extensions. Some of our algorithms require that we are able to efficiently compute primary decompositions of finite k-algebras. This condition is fulfilled, for example, if k is a finite field or a number field. We give algorithms for decomposing a divisor as a linear combination of prime divisors, computing pull-backs and push-forwards of divisors under finite morphisms, and computing Picard and Albanese maps induced by finite morphisms. We also consider some more technical problems that are needed in the rest of the paper. In Section 3 we describe the rest of our algorithms, which are specific to curves over finite fields. In particular, we show how to compute the Frobenius map on points of a curve and of its Jacobian over finite extensions of the base field, how to generate uniformly random effective divisors of a given degree and uniformly random points of the Jacobian (given the zeta function of the curve), and how to compute Frey–Rück pairings on the Jacobian. By combining the above methods, we also show that if we know the zeta function of the curve, the methods of Couveignes [5] for computing Kummer maps of order l and for finding a basis for the l-torsion of the Picard group, where l is a prime number different from the characteristic of the base field, can be extended to our situation.

Remarks. (1) When the field k is finite, measuring the running time in field operations is essentially the same as measuring it in bit operations. However, if k is a number field, it is impossible to avoid numerical explosion of the data describing the divisors during computations such as multiplication by a large integer, so that the running time in bit operations is much worse than that counted in field operations. Using lattice reduction algorithms to reduce the size of the data between operations should not be expected to solve this problem; see Khuri-Makdisi [13, page 2214].
(2) Many of the algorithms we describe are probabilistic. All of these are of the *Las Vegas* type. This means that the running time depends on certain random data generated during the execution of the algorithm, but that the outcome is guaranteed to be correct. The epithet *Las Vegas* distinguishes such algorithms from those of the *Monte Carlo* type, where the randomness influences the correctness of the outcome instead of the running time.
(3) The algorithms mentioned in this paper have a running time that is bounded by some polynomial in various quantities that are indicated in each case. Obtaining more detailed estimates should not be difficult, but has at the time of writing not yet been done.
(4) The algorithms presented in this paper are relevant for computations with curves of large genus over finite fields. The author's interest in such computations was raised by the search for an algorithm for efficiently computing coefficients of modular forms. In the book [9], Edixhoven, Couveignes and others describe such an algorithm for modular forms for the group $\mathrm{SL}_2(\mathbf{Z})$. In the author's thesis [3], their methods are generalised to modular forms for other groups $\Gamma_1(n)$. The method in each case is to compute two-dimensional modular Galois representations over finite fields. The basic problem one needs to solve is to find explicit realisations of group schemes over $\mathbf{Q}$ of the form $J[\mathfrak{m}]$ with J the Jacobian of a modular curve and $\mathfrak{m}$ a maximal ideal of the Hecke algebra acting on J. As a scheme, $J[\mathfrak{m}]$ can be embedded into the affine line over $\mathbf{Q}$; the image then gets a group scheme structure described by polynomials over $\mathbf{Q}$. To compute $J[\mathfrak{m}]$ efficiently, these rational data are approximated either over the complex numbers or modulo sufficiently many small prime

numbers. The complex method has already been used by Bosman [2] in actual computations. The method using finite fields was described by Couveignes [5] for the modular curves $\mathrm{X}_1(5l)$ with l a prime number. The computations in this case can be done using singular plane models for these curves. For more general modular curves X, it is natural to embed X as a smooth curve in a higher-dimensional projective space via the line bundle of modular forms of weight 2; this is the approach used in [3]. Using modular symbols [20], one can compute q-expansions of these modular forms and the zeta function of X. This directly gives a representation of X tailored for our algorithms, without having to write down equations.

1. Algorithms for computing with finite algebras

In this section, we consider two computational problems about finite commutative algebras over a field: finding the primary decomposition of such an algebra, and reconstructing such an algebra from a certain kind of bilinear map between modules over it. The algebras to which we are going to apply these techniques in the next section are of the form $\Gamma(E, \mathcal{O}_E)$, where E is an effective divisor on a smooth curve over a field k. In this section, however, we place ourselves in the more general setting of arbitrary finite commutative k-algebras.

1.1. **Primary decomposition.** Let k be a field. We assume the following:

(1) We have a way to represent elements of k by bit strings (or, equivalently, by non-negative integers).
(2) We have algorithms that, given elements of k in this representation, perform field operations in k (addition, subtraction, multiplication, division, testing equality) in a number of bit operations that is bounded by a polynomial in the length of the input.
(3) We have an algorithm that, given a finite k-algebra A in the form of its multiplication table with respect to some k-basis, determines the primary decomposition of A, as a list of maximal ideals $\mathfrak{m}$ of A and the corresponding $\mathfrak{m}$-primary quotients $A \to A_{\mathfrak{m}}$, in a number of bit operations that is bounded by a polynomial in $[A : k]$ and the maximum of the bit lengths of the coefficients of the multiplication table.

Although every element of k must be representable by a bit string, such a representation need not be unique. The algorithms may depend on k; we only require the running time to be polynomial with respect to varying input for fixed k. Whenever we make use of the above assumptions on k in this section and the next, we will therefore specify running times for fixed k. We allow probabilistic algorithms, in which case "running time" is understood to mean "expected running time".

If k is perfect, the third assumption is fulfilled if we have a (probabilistic) algorithm that, given a polynomial $f \in k[x]$ by means of bit representations of its coefficients, factors f in an (expected) number of operations in k that is bounded by a polynomial in $\deg f$ and the maximum of the bit lengths of the coefficients. In fact, algorithms reducing the problem of primary decomposition for finite algebras over a perfect field k to the problem of factoring polynomials over k have been known for some time, but do not seem to be easily available in published form; see Khuri-Makdisi's preprint [13, draft version 2, §7]. For an algorithm to find the primary decomposition of finite algebras (not necessarily commutative) over *finite* fields, see Eberly and Giesbrecht [8].

In Section 3, we will allow the base field to vary. In this context, it is relevant to note that there exist algorithms as above for which the running time with respect to a varying finite field k is polynomial in $\log k$.

1.2. **Reconstructing an algebra from a perfect bilinear map.** Let A be a commutative ring. If M, N and O are free A-modules of rank one and

$$\mu\colon M \times N \to O$$

is an A-bilinear map, we say that μ is *perfect* if it induces an isomorphism

$$M \otimes_A N \xrightarrow{\sim} O$$

of free A-modules of rank 1.

Now let k be a field, and let a finite commutative k-algebra A be specified implicitly in the following way. We are given k-vector spaces M, N and O of the same finite dimension, together with a k-bilinear map

$$\mu\colon M \times N \to O.$$

We assume there exists a commutative k-algebra A such that M, N and O are free A-modules of rank 1 and μ is a perfect A-bilinear map. The following observation implies that A is the *unique* k-algebra with this property, and also shows how to compute A as a subalgebra of $\operatorname{End}_k M$, provided we are able to find a generator of N as an A-module. We note that the roles of M and N can be interchanged.

Lemma 1.1. *In the above situation, let g be a generator of the A-module N. The ring homomorphism $A \to \operatorname{End}_k M$ sending a to multiplication by a is, as an A-linear map, the composition of*

$$\begin{aligned} A &\xrightarrow{\sim} N \\ a &\longmapsto ag \end{aligned}$$

and

$$\begin{aligned} N &\longrightarrow \operatorname{End}_k M \\ n &\longmapsto \mu(\ \ , g)^{-1} \circ \mu(\ \ , n). \end{aligned}$$

In particular, the image of A in $\operatorname{End}_k M$ equals the image of the second map.

Proof. This is a straightforward verification. □

In the case where k is a finite field, a way to find a generator for N as an A-module is simply to pick random elements $g \in N$ until we find one that generates N. Since μ is perfect, checking whether g generates N comes down to checking whether $\mu(\ \ , g)\colon M \to O$ is an isomorphism. In particular, we can do this without knowing A.

To get a reasonable expected running time for this approach, we need to ensure that N contains sufficiently many elements n such that $N = An$. Since N is free of rank 1, the number of generators equals the number of units in A. Let us therefore estimate under what conditions a random element of A is a unit with probability at least $1/2$. Write d for the degree of A over k. Decomposing A into a product of finite local k-algebras, and noting that the proportion of units in a finite local k-algebra is equal to the proportion of units in its residue field, we see that

$$\frac{\#A^\times}{\#A} \geq \frac{(\#k^\times)^d}{\#k^d} = \left(1 - \frac{1}{\#k}\right)^d;$$

equality occurs if and only if A is a product of d copies of k. Now it is not hard to show that

$$\#k \geq 2d \implies \left(1 - \frac{1}{\#k}\right)^d \geq \frac{1}{2}.$$

Taking a finite extension k' of k of cardinality at least $2d$, we therefore see that a random element of $A_{k'}$ is a unit with probability at least $1/2$. There are well-known algorithms to generate such an extension, such as that of Rabin [17], which runs in probabilistic polynomial time and simply tries random polynomials until it finds one that is irreducible, and the deterministic algorithm of Adleman and Lenstra [1].

Algorithm 1.2 (Reconstruct an algebra from a bilinear map)**.** *Let k be a finite field, let A be a finite k-algebra, and let*

$$\mu\colon M \times N \to O$$

be a perfect A-bilinear map between free A-modules of rank 1. Given the coefficients of μ relative to some k-bases of M, N and O, this algorithm outputs a k-basis of the image of A in $\operatorname{End}_k M$ consisting of matrices relative to the given basis of M.

1. *Choose an extension k' of k of degree $\left\lceil \frac{\log \max\{2[A:k], \#k\}}{\log \#k} \right\rceil$. Let M', N', O' and μ' denote the base extensions of M, N, O and μ to k'.*
2. *Choose a uniformly random element $g \in N'$.*
3. *Check whether $\mu'(\ \ , g)\colon M' \to O'$ is an isomorphism; if not, go to step 2.*
4. *For n ranging over a k'-basis of N', compute the endomorphism*

$$a_n = \mu'(\ \ , g)^{-1} \circ \mu'(\ \ , n) \in \operatorname{End}_{k'} M'.$$

   Let $A' \subseteq \operatorname{End}_{k'} M'$ denote the k'-span of the a_n.
5. *Using k-bases for $\operatorname{End}_{k'} M'$ and its k-linear subspaces $\operatorname{End}_k M$ and A', compute a k-basis for $\operatorname{End}_k M \cap A'$ and output this basis.*

Analysis. It follows from Lemma 1.1 that A' equals the image of $k' \otimes_k A$ in $\operatorname{End}_{k'} M$. This implies that the basis returned by the algorithm is indeed a k-basis for the image of A in $\operatorname{End}_k M$. Because of the choice of k', steps 2 and 3 are executed at most twice on average. It is therefore clear that the expected running time of the algorithm, measured in operations in k, is polynomial in $[A : k]$. ♢

If k is infinite, or finite and sufficiently large, we have the following variant. Let Σ be a finite subset of k, and let V be a k-vector space of dimension d with a given basis $v_1, \ldots, v_d$. The set of Σ-*linear combinations* of $v_1, \ldots, v_d$ is

$$V_\Sigma = \left\{ \sum_{i=1}^{d} \sigma_i v_i \mid \sigma_1, \ldots, \sigma_d \in \Sigma \right\}.$$

Choosing the σ_i uniformly randomly in Σ, we get the uniform distribution on V_Σ. If $H_1, \ldots, H_l$ are proper linear subspaces of V, then a uniformly random element of V_Σ lies in at least one of the H_i with probability at most $l/\#\Sigma$. Now if A is a finite commutative k-algebra, it contains at most $[A : k]$ maximal ideals. This implies that if Σ is a finite subset of k with $\#\Sigma \geq 2[A : k]$, then a Σ-linear combination of any k-basis of A is a unit with probability at least $1/2$. This leads to the following analogue of Algorithm 1.2 for an arbitrary base field.

Algorithm 1.3 (Reconstruct an algebra from a bilinear map)**.** *Let k be a field, let A be a finite k-algebra, and let*

$$\mu\colon M \times N \to O$$

be a perfect A-bilinear map between free A-modules of rank 1. Given the coefficients of μ relative to some k-bases of M, N and O, this algorithm outputs a k-basis of the image of A in $\operatorname{End}_k M$, consisting of matrices relative to the given basis of M.

1. *Let p be the characteristic of k. Check whether $0, 1, \ldots, 2[A:k]-1$ are all distinct in k (this is the case if and only if $p = 0$ or $p \geq 2[A:k]$). If so, let $\Sigma = \{0, 1, \ldots, 2[A:k]-1\} \subseteq k$ and go to step 5. If not, let $k_0 = \mathbf{F}_p \subseteq k$.*
2. *For a running through the coefficients of the multiplication table of A over k:*
3. *Compute the set $F = \{f \in k_0[x] \mid f \text{ is monic and } (\#k_0)^{\deg f} < 2[A:k]\}$. If a is a zero of some polynomial in F, replace k_0 by $k_0(a) \subseteq k$. If not, let $\Sigma = \{0\} \cup \{af \mid a \in k^\times, f \in F\} \subseteq k$ and go to step 5.*
4. *Now k_0 is a finite field with $\#k_0 < 2[A:k]$ and A is obtained by base extension to k of the finite k_0-algebra A_0 defined by the same multiplication table. Apply Algorithm 1.2 to A_0 over k_0 and return the result, viewed as data over k.*
5. *Choose a uniformly random Σ-linear combination g of the given basis of N.*
6. *Check whether $\mu(\ ,g)\colon M \to O$ is an isomorphism; if not, go to step 5.*
7. *For n ranging over a k-basis of N, compute the endomorphism*

$$a_n = \mu(\ ,g)^{-1} \circ \mu(\ ,n) \in \operatorname{End}_k M,$$

and output the a_n.

Analysis. The set F constructed in step 3 has $1 + \#k_0 + \cdots + (\#k_0)^d$ elements, where d is the greatest integer such that $(\#k_0)^d < 2[A:k]$. This implies that if step 5 is reached, the set Σ used there has at least $2[A:k]$ elements. Using this, it is straightforward to show that the expected running time, measured in operations in k, is polynomial in $[A:k]$. ♢

2. Computing with divisors on a curve

In this section, we consider complete, smooth and geometrically connected curves over a field k. For some of the algorithms we assume that k satisfies the hypotheses of §1.1. In Section 3, we will restrict to curves over finite fields.

Of the material in this section, §§2.1–2.4 and 2.8 are largely based on Khuri-Makdisi's articles [12] and [13], while §§2.5–2.7, 2.9 and 2.10 seem to be new.

2.1. **Representing the curve.** Let X be a complete, smooth, geometrically connected curve over a field k. We fix a line bundle $\mathcal{L}$ on X such that

$$\deg \mathcal{L} \geq 2g + 1.$$

Then $\mathcal{L}$ is very ample (see for example Hartshorne [11, IV, Corollary 3.2(b)]), so it gives rise to a closed immersion

$$i_{\mathcal{L}}\colon X \longrightarrow \mathbf{P}\Gamma(X, \mathcal{L})$$

into a projective space of dimension $\deg \mathcal{L} - g$. (We write $\mathbf{P}V$ for the projective space of hyperplanes in a k-vector space V.) The assumption that $\deg \mathcal{L} \geq 2g + 1$ implies, moreover, that the multiplication maps

$$\mu_{i,j}\colon \Gamma(X, \mathcal{L}^{\otimes i}) \otimes_k \Gamma(X, \mathcal{L}^{\otimes j}) \longrightarrow \Gamma(X, \mathcal{L}^{\otimes (i+j)})$$

are surjective for all $i, j \geq 0$ or, equivalently, that the embedding $i_{\mathcal{L}}$ is projectively normal. This is a classical theorem of Castelnuovo [4, no. 5], Mattuck [15, page 194] and Mumford [16, page 55]. Below we will state a more general result due to Khuri-Makdisi [12, Lemma 2.2].

Remark. In the context of projective embeddings, the line bundle $\mathcal{L}$ is usually denoted by $\mathcal{O}_X(1)$. However, we often need to deal with line bundles of the form $\mathcal{L}^{\otimes i}(D)$ for a divisor D, and the author does not like the notation $\mathcal{O}_X(i)(D)$.

We write S_X for the homogeneous coordinate ring of X with respect to the embedding $i_{\mathcal{L}}$. There is a canonical injective homomorphism

$$S_X \longrightarrow \bigoplus_{i \geq 0} \Gamma(X, \mathcal{L}^{\otimes i})$$

of graded k-algebras, which is an isomorphism by the fact that $i_{\mathcal{L}}$ is projectively normal; see Hartshorne [11, Chapter II, Exercise 5.14]. It turns out that in order to compute with divisors on X, we do not need to know the complete structure of S_X. For all $h \geq 0$, we define the finite graded k-algebra $S_X^{(h)}$ as the quotient of S_X by the ideal generated by homogeneous elements of degree greater than h. Specifying $S_X^{(h)}$ is equivalent to giving the k-vector spaces $\Gamma(X, \mathcal{L}^{\otimes i})$ for $1 \leq i \leq h$ together with the multiplication maps $\mu_{i,j}$ for $i + j \leq h$.

When speaking of a *projective curve* X in the remainder of this section, we will assume without further mention that X is a complete, smooth and geometrically connected curve of genus $g \geq 0$, and that a line bundle $\mathcal{L}$ of degree at least $2g + 1$ has been chosen. We will often write $\mathcal{L}_X$ for this line bundle and g_X for the genus of X to emphasise that they are part of the data.

In the algorithms in this section, the curve X is part of the input in the guise of the graded k-algebra $S_X^{(h)}$ for some sufficiently large h. A lower bound for h is specified in each case. One way to specify the multiplication in $S_X^{(h)}$ is to fix bases for the spaces $\Gamma(X, \mathcal{L}^{\otimes i})$, and to give the matrices for multiplication with each basis element. However, as Khuri-Makdisi explains in [13], a more efficient representation is to choose a trivialisation of $\mathcal{L}$ (and hence of its powers) over an effective divisor of sufficiently large degree or, even better, at sufficiently many distinct rational points of X, so that the multiplication maps can be computed pointwise.

Remarks. (1) The integers g and $\deg \mathcal{L}$ can of course be stored as part of the data describing X. However, they can also be extracted from the dimensions of the k-vector spaces $\Gamma(X, \mathcal{L})$ and $\Gamma(X, \mathcal{L}^{\otimes 2})$; this follows easily from the Riemann–Roch formula.
(2) If the degree of $\mathcal{L}$ is at least $2g + 2$, then the homogeneous ideal defining the embedding $i_{\mathcal{L}}$ is generated by homogeneous elements of degree 2, according to a theorem of Fujita and Saint-Donat; see Lazarsfeld [14, §1.1]. This allows us to deduce equations for X from the k-algebra $S_X^{(2)}$, but we will not need to do this.
(3) The way of representing curves and divisors described in [12] and [13] is especially suited for modular curves. Namely, we can represent a modular curve X using the projective embedding given by a line bundle of modular forms, and computing the k-algebra $S_X^{(h)}$ for a given h comes down to computing q-expansions of modular forms of a suitable weight to a sufficiently large order. This can be done using modular symbols; see Stein [20]. If the modular curve has at least 3 cusps

(which is the case, for example, for $X_1(n)$ for all $n \geq 5$), then we can restrict ourselves to modular forms of weight 2, for which the formalism of modular symbols is particularly simple [20, Chapter 3].

2.2. **Representing divisors.** Let X be a projective curve of genus g in the sense of §2.1. To represent divisors on X, it is enough to consider effective divisors, since every divisor is a difference of two effective divisors.

If i is a positive integer, D is an effective divisor and confusion is impossible, we will use the abbreviation

$$\Gamma(\mathcal{L}_X^{\otimes i}(-D)) = \Gamma(X, \mathcal{L}_X^{\otimes i}(-D)).$$

Let D be an effective divisor on X such that $\mathcal{L}_X(-D)$ is generated by global sections. In terms of the projective embedding, this means that D is the intersection of X and a linear subvariety of $\mathbf{P}\Gamma(\mathcal{L}_X)$ or, equivalently, that D is defined by a system of linear equations. We represent D as the subspace $\Gamma(\mathcal{L}_X(-D))$ of $\Gamma(\mathcal{L}_X)$ consisting of sections vanishing on D. The codimension of $\Gamma(\mathcal{L}_X(-D))$ in $\Gamma(\mathcal{L}_X)$ is equal to the degree of D.

A sufficient condition for $\mathcal{L}_X(-D)$ to be generated by global sections is

$$\deg D \leq \deg \mathcal{L}_X - 2g; \tag{2.1}$$

see for example Hartshorne [11, IV, Corollary 3.2(a)]. However, in general not every subspace of codimension at most $\deg \mathcal{L}_X - 2g$ in $\Gamma(\mathcal{L}_X)$ is of the form $\Gamma(\mathcal{L}_X(-D))$ for an effective divisor D of the same degree.

Remark. This way of representing divisors comes down, for divisors of degree $d \leq \deg \mathcal{L}_X - 2g$, to embedding the d-th symmetric power of X into the Grassmann variety parametrising subspaces of codimension d in $\Gamma(\mathcal{L}_X)$ and viewing divisors of degree d as points on this Grassmann variety.

It will often be necessary to consider divisors D of degree larger than the bound $\deg \mathcal{L}_X - 2g$ of (2.1). In such cases we can represent D as a subspace of $\Gamma(\mathcal{L}_X^{\otimes i})$ for i sufficiently large such that

$$\deg D \leq i \deg \mathcal{L}_X - 2g, \tag{2.2}$$

provided of course that we know $S_X^{(h)}$ for some $h \geq i$.

Khuri-Makdisi's algorithms rest on the following two results. The first is a generalisation of the theorem of Castelnuovo, Mattuck and Mumford mentioned above. It says in effect that to compute the space of global sections of the tensor product of two line bundles of sufficiently large degree, it is enough to multiply global sections of those line bundles.

Lemma 2.1 (Khuri-Makdisi [12, Lemma 2.2])**.** *Let X be a complete, smooth, geometrically connected curve of genus g over a field k, and let $\mathcal{M}$ and $\mathcal{N}$ be line bundles on X whose degrees are at least $2g+1$. Then the canonical k-linear map*

$$\Gamma(X, \mathcal{M}) \otimes_k \Gamma(X, \mathcal{N}) \longrightarrow \Gamma(X, \mathcal{M} \otimes_{\mathcal{O}_X} \mathcal{N})$$

is surjective.

The second result shows how to find the space of global sections of a line bundle that vanish on a given effective divisor, where this divisor is represented as a subspace of global sections of a second line bundle.

Lemma 2.2 (Khuri-Makdisi [12, Lemma 2.3])**.** *Let X be a complete, smooth, geometrically connected curve of genus g over a field k, let $\mathcal{M}$ and $\mathcal{N}$ be line bundles on X such that $\mathcal{N}$ is generated by global sections, and let D be any effective divisor on X. Then the inclusion*

$$\Gamma(X,\mathcal{M}(-D)) \subseteq \left\{ s \in \Gamma(X,\mathcal{M}) \;\middle|\; s\Gamma(X,\mathcal{N}) \subseteq \Gamma(X,\mathcal{M}\otimes\mathcal{N}(-D)) \right\} \tag{2.3}$$

is an equality.

Thanks to these two lemmas, one can give algorithms to do basic operations on divisors; see [12, §3]. For example, we can add, subtract and intersect divisors of sufficiently small degree, and we can test whether a given subspace of $\Gamma(\mathcal{L}_X^{\otimes i})$ is of the form $\Gamma(\mathcal{L}_X^{\otimes i}(-D))$ for some effective divisor D. See also Algorithm 2.11 below for an example where Lemmas 2.1 and 2.2 are used.

2.3. **Deflation and inflation.** A method used in [13] to speed up the algorithms is *deflation* of subspaces. Suppose we want to compute the space $\Gamma(X,\mathcal{M}(-D))$ using Lemma 2.2 in the case where $\mathcal{M} = \mathcal{L}_X^{\otimes i}$ and $\mathcal{N} = \mathcal{L}_X^{\otimes j}(-E)$ with i and j positive integers and where D and E are effective divisors satisfying (2.2). On the right-hand side of (2.3), we may replace $\Gamma(X,\mathcal{N})$ by any basepoint-free subspace; this is clear from the proof of [12, Lemma 2.3]. It turns out that there always exists such a subspace of dimension $O(\log(\deg\mathcal{N}))$, and a subspace of dimension 2 exists if the base field is either infinite or finite of sufficiently large cardinality. Moreover, one can efficiently find such a subspace by random trial; see [13, Proposition/Algorithm 3.7].

Remark. This random search for small basepoint-free subspaces is the reason why the algorithms in [13] are probabilistic, as opposed to those in [12].

Suppose we are given a basepoint-free subspace W of $\Gamma(\mathcal{L}_X^{\otimes i}(-D))$ for some i and D such that $\Gamma(\mathcal{L}_X^{\otimes i}(-D))$ is basepoint-free. Then we can reconstruct the complete space $\Gamma(\mathcal{L}_X^{\otimes i}(-D))$ from W. This procedure is called *inflation*. To describe how this can be done, we first state the following slight generalisation of a result of Khuri-Makdisi [13, Theorem 3.5(2)].

Lemma 2.3. *Let X be a complete, smooth, geometrically connected curve of genus g over a field k, and let $\mathcal{M}$ and $\mathcal{N}$ be line bundles on X. Let V be a non-zero subspace of $\Gamma(X,\mathcal{M})$, and let D be the common divisor of the elements of V. If the inequality*

$$-\deg\mathcal{M} + \deg\mathcal{N} + \deg D \geq 2g-1$$

is satisfied, the canonical k-linear map

$$V \otimes_k \Gamma(X,\mathcal{N}) \longrightarrow \Gamma(X,\mathcal{M}\otimes_{\mathcal{O}_X}\mathcal{N}(-D)) \tag{2.4}$$

is surjective.

Proof. We note that $\mathcal{M}(-D)$ is generated by global sections, since we can view V as a subspace of $\Gamma(X,\mathcal{M}(-D))$ and the elements of V have common divisor 0 as sections of $\mathcal{M}(-D)$. We also note that $\deg\mathcal{M} \geq \deg D$. Therefore the assumption on the degrees of $\mathcal{M}$, $\mathcal{N}$ and D implies the inequalities

$$\deg\mathcal{N} \geq 2g-1$$

and

$$\deg(\mathcal{M}\otimes\mathcal{N}(-D)) \geq 2g-1.$$

After extending the field k, we may assume it is infinite. Then there exist elements $s, t \in V$ with common divisor D; see [13, Lemma 4.1]. The space

$$s\Gamma(X,\mathcal{N}) + t\Gamma(X,\mathcal{N})$$

lies in the image of (2.4), so it suffices to show that

$$\dim_k(s\Gamma(X,\mathcal{N}) + t\Gamma(X,\mathcal{N})) = \dim_k \Gamma(X, \mathcal{M} \otimes \mathcal{N}(-D)).$$

There exist disjoint effective divisors E and F such that

$$\operatorname{div} s = D + E \quad \text{and} \quad \operatorname{div} s = D + F.$$

We have

$$\begin{aligned}\dim_k(s\Gamma(X,\mathcal{N}) + t\Gamma(X,\mathcal{N})) &= 2\dim_k \Gamma(X,\mathcal{N}) - \dim_k(s\Gamma(X,\mathcal{N}) \cap t\Gamma(X,\mathcal{N})) \\ &= 2\dim_k \Gamma(X,\mathcal{N}) - \dim_k \Gamma(X, \mathcal{M} \otimes \mathcal{N}(-D-E-F)) \\ &= 2\dim_k \Gamma(X,\mathcal{N}) - \dim_k \Gamma(X, \mathcal{M}^\vee \otimes \mathcal{N}(D)).\end{aligned}$$

The last equality follows from the fact that multiplication by st is an isomorphism

$$\mathcal{M}^\vee(D) \xrightarrow{\sim} \mathcal{M}(-D-E-F).$$

From the fact that the various line bundles have degrees at least $2g-1$, we see that

$$\begin{aligned}\dim_k(s\Gamma(X,\mathcal{N}) + t\Gamma(X,\mathcal{N})) &= 2(1-g+\deg \mathcal{N}) - (1-g+\deg \mathcal{M}^\vee \otimes \mathcal{N}(D)) \\ &= 1-g+\deg \mathcal{M} + \deg \mathcal{N} - \deg D \\ &= \dim_k \Gamma(X, \mathcal{M} \otimes \mathcal{N}(-D)).\end{aligned}$$

This finishes the proof. □

To find the inflation of a basepoint-free subspace W of $\Gamma(\mathcal{L}_X^{\otimes i}(-D))$, we choose a positive integer j such that

$$(j-i)\deg \mathcal{L}_X + \deg D \geq 2g-1.$$

By Lemma 2.3 we can compute $\Gamma(\mathcal{L}_X^{\otimes(i+j)}(-D))$ as the image of the bilinear map

$$W \otimes_k \Gamma(\mathcal{L}_X^{\otimes j}) \longrightarrow \Gamma(\mathcal{L}_X^{\otimes(i+j)}).$$

Using Lemma 2.2, we then compute

$$\Gamma(\mathcal{L}_X^{\otimes i}(-D)) = \left\{ s \in \Gamma(\mathcal{L}_X^{\otimes i}) \;\middle|\; s\Gamma(\mathcal{L}_X^{\otimes j}) \subseteq \Gamma(\mathcal{L}_X^{\otimes(i+j)}(-D)) \right\}$$

We note that for this last step we can use a small basepoint-free subspace of $\Gamma(\mathcal{L}_X^{\otimes j})$ computed in advance.

2.4. **Decomposing divisors into prime divisors.** Let X be a complete, smooth, geometrically connected curve of genus g over a field k, with a projective embedding via a line bundle $\mathcal{L}$ as in §2.1. The problem we are now going to study is how to find the decomposition of a given divisor on X as a linear combination of prime divisors. We will see below that this can be done if we are given the algebra $S_X^{(h)}$ for sufficiently large h and if we are able to compute the primary decomposition of a finite commutative k-algebra. We recall from §1.1 that this is possible if k is a perfect field such that we can factor polynomials in one variable over k.

Let i be a positive integer, and let D be an effective divisor such that

$$\deg D \leq i \deg \mathcal{L} - 2g + 1.$$

We view D as a closed subscheme of X via the canonical closed immersion

$$j_D \colon D \to X.$$

For every line bundle $\mathcal{M}$ on X, the k-vector space $\Gamma(D, j_D^*\mathcal{M})$ is in a natural way a free module of rank one over $\Gamma(D, \mathcal{O}_D)$. The multiplication map

$$\mu_{i,i} \colon \Gamma(X, \mathcal{L}^{\otimes i}) \times \Gamma(X, \mathcal{L}^{\otimes i}) \longrightarrow \Gamma(X, \mathcal{L}^{\otimes 2i})$$

descends to a bilinear map

$$\mu_{i,i}^D \colon \Gamma(D, j_D^*\mathcal{L}^{\otimes i}) \times \Gamma(D, j_D^*\mathcal{L}^{\otimes i}) \longrightarrow \Gamma(D, j_D^*\mathcal{L}^{\otimes 2i})$$

of free modules of rank 1 over $\Gamma(D, \mathcal{O}_D)$. This map is perfect in the sense of §1.2.

We now assume that the graded k-algebra $S_X^{(h)}$ as in §2.1 is given for some $h \geq 2$. From the subspace $\Gamma(X, \mathcal{L}^{\otimes i}(-D))$ of $\Gamma(X, \mathcal{L}^{\otimes i})$ we can then determine $\Gamma(D, j_D^*\mathcal{L}^{\otimes i})$ as a k-vector space by means of the short exact sequence

$$0 \longrightarrow \Gamma(X, \mathcal{L}^{\otimes i}(-D)) \longrightarrow \Gamma(X, \mathcal{L}^{\otimes i}) \longrightarrow \Gamma(D, j_D^*\mathcal{L}^{\otimes i}) \longrightarrow 0. \tag{2.5}$$

(Note that exactness on the right follows from the assumption that $\deg \mathcal{L}^{\otimes i}(-D) \geq 2g - 1$.) Similarly, we can compute $\Gamma(D, j_D^*\mathcal{L}^{\otimes 2i})$ from $\Gamma(X, \mathcal{L}^{\otimes 2i}(-D))$ using the same sequence with i replaced by $2i$. We can then determine the bilinear map $\mu_{i,i}^D$ induced by $\mu_{i,i}$ by standard methods from linear algebra.

We then use Algorithm 1.3 to compute the k-algebra $\Gamma(D, \mathcal{O}_D)$ together with its action on $\Gamma(D, j_D^*\mathcal{L}^{\otimes i})$. Next we find the primary decomposition of $\Gamma(D, \mathcal{O}_D)$, say,

$$\Gamma(D, \mathcal{O}_D) \cong A_1 \times A_2 \times \cdots \times A_r,$$

where each factor A_i is a finite local k-algebra with maximal ideal P_i; we assume the field k is such that we can do this (see §1.1). Such a prime ideal P_i corresponds to a prime divisor in the support of D, and the corresponding multiplicity equals

$$m_i = \frac{[A_i : k]}{[A_i/P_i : k]}.$$

Algorithm 2.4 (Decomposition of a divisor)**.** *Let X be a projective curve over a field k satisfying the assumptions of* §1.1*. Let i be a positive integer, and let D be an effective divisor such that*

$$\deg D \leq i \deg \mathcal{L}_X - 2g_X + 1.$$

Given the k-algebra $S_X^{(2i)}$ and the subspaces $\Gamma(X, \mathcal{L}_X^{\otimes i}(-D))$ of $\Gamma(X, \mathcal{L}_X^{\otimes i})$ and $\Gamma(X, \mathcal{L}_X^{\otimes 2i}(-D))$ of $\Gamma(X, \mathcal{L}_X^{\otimes 2i})$, this algorithm outputs the decomposition of D as a linear combination of prime divisors as a list of pairs (P, m_P), where P is a prime divisor and m_P is the multiplicity of P in D.

1. *Compute the spaces $\Gamma(D, j_D^*\mathcal{L}_X^{\otimes i})$ and $\Gamma(D, j_D^*\mathcal{L}_X^{\otimes 2i})$ using* (2.5) *and the analogous short exact sequence with $2i$ in place of i.*
2. *Compute the k-bilinear map $\mu_{i,i}^D$ from $\mu_{i,i}$.*
3. *Using Algorithm* 1.3*, compute a k-basis for $\Gamma(D, \mathcal{O}_D)$ as a linear subspace of $\operatorname{End}_k \Gamma(D, j_D^*\mathcal{L}_X^{\otimes i})$, where elements of the latter k-algebra are expressed as matrices with respect to some fixed basis of $\Gamma(D, j_D^*\mathcal{L}_X^{\otimes i})$.*
4. *Compute the multiplication table of $\Gamma(D, \mathcal{O}_D)$ on the k-basis of $\Gamma(D, \mathcal{O}_D)$ found in the previous step.*
5. *Find the primary decomposition of $\Gamma(D, \mathcal{O}_D)$.*

6. *For each local factor A computed in the previous step, with maximal ideal P_A, output the inverse image of $P_A \cdot \Gamma(D, j_D^* \mathcal{L}_X^{\otimes i})$ in $\Gamma(X, \mathcal{L}_X^{\otimes i})$ and the integer $[A : k]/[A/P_A : k]$.*

Analysis. The correctness of the algorithm follows from the above discussion. It is straightforward to check that the running time, for fixed k, is polynomial in i, $\deg \mathcal{L}_X$ and the maximum of the bit lengths of the coefficients in the input. ♢

A special case of this algorithm is when D is the intersection of X with a hypersurface of degree $i-1$. Let s be a non-zero section of $\mathcal{L}_X^{\otimes(i-1)}$ defining this hypersurface. The subspaces used in this algorithm can then be computed as

$$\Gamma(X, \mathcal{L}_X^{\otimes i}(-D)) = s\Gamma(X, \mathcal{L}_X) \quad \text{and} \quad \Gamma(X, \mathcal{L}_X^{\otimes 2i}(-D)) = s\Gamma(X, \mathcal{L}_X^{\otimes(i+1)}).$$

2.5. **Finite morphisms between curves.** Let us now look at finite morphisms between curves. A finite morphism

$$f\colon X \to Y$$

of complete, smooth, geometrically connected curves induces two functors:

$$f^*\colon \{\text{line bundles on } Y\} \longrightarrow \{\text{line bundles on } X\},$$
$$\mathrm{N}_f\colon \{\text{line bundles on } X\} \longrightarrow \{\text{line bundles on } Y\}.$$

Here $f^*\mathcal{N}$ denotes the usual inverse image of the line bundle $\mathcal{N}$ on Y, and $\mathrm{N}_f\mathcal{M}$ is the *norm* of the line bundle $\mathcal{M}$ on X under the morphism f.

Let us briefly explain the notion of the norm of a line bundle. The norm functor is a special case (that of $\mathbf{G}_\mathrm{m}$-torsors) of the *trace of a torsor* for a commutative group scheme under a finite locally free morphism; see Deligne [19, exposé XVII, nos. 6.3.20–6.3.26]. We formulate the basic results for arbitrary finite locally free morphisms of schemes

$$f\colon X \to Y.$$

In this situation there exists a functor

$$\mathrm{N}_f\colon \{\text{line bundles on } X\} \to \{\text{line bundles on } Y\}$$

together with a collection of homomorphisms

$$\mathrm{N}_f^{\mathcal{L}}\colon f_*\mathcal{L} \to \mathrm{N}_f\mathcal{L}$$

of sheaves of sets, for all line bundles $\mathcal{L}$ on X, functorial under isomorphisms of line bundles on X, sending local generating sections on X to local generating sections on Y and such that the equality

$$\mathrm{N}_f^{\mathcal{L}}(xl) = \mathrm{N}_f(x) \cdot \mathrm{N}_f^{\mathcal{L}}(l)$$

holds for all local sections x of $f_*\mathcal{O}_X$ and l of $f_*\mathcal{L}$. Here $\mathrm{N}_f\colon f_*\mathcal{O}_X \to \mathcal{O}_Y$ denotes the usual norm map for a finite locally free morphism. Moreover, the functor N_f together with the collection of the $\mathrm{N}_f^{\mathcal{L}}$ is unique up to unique isomorphism. Instead of N_f we also write $\mathrm{N}_{X/Y}$ if the morphism f is clear from the context.

The norm functor has the following basic properties [19, exposé XVII, no. 6.3.26]:

(1) the functor N_f is compatible with any base change $Y' \to Y$;
(2) if $\mathcal{L}_1$ and $\mathcal{L}_2$ are two line bundles on X, there is a natural isomorphism

$$\mathrm{N}_f(\mathcal{L}_1 \otimes_{\mathcal{O}_X} \mathcal{L}_2) \cong \mathrm{N}_f\mathcal{L}_1 \otimes_{\mathcal{O}_Y} \mathrm{N}_f\mathcal{L}_2;$$

(3) if $X \xrightarrow{f} Y \xrightarrow{g} Z$ are finite locally free morphisms, there is a natural isomorphism

$$\mathrm{N}_{g\circ f} \xrightarrow{\sim} \mathrm{N}_g \circ \mathrm{N}_f.$$

Furthermore, there is a functorial isomorphism

$$\mathrm{N}_f\mathcal{L} \xrightarrow{\sim} \mathrm{Hom}_{\mathcal{O}_Y}(\det_{\mathcal{O}_Y} f_*\mathcal{O}_X, \det_{\mathcal{O}_Y} f_*\mathcal{L}); \tag{2.6}$$

see Deligne [19, exposé XVIII, no. 1.3.17], and compare Hartshorne [11, IV, Exercise 2.6].

We now consider projective curves X and Y as defined in §2.1. Suppose we have a finite morphism

$$f\colon X \to Y$$

with the property that f is induced by a graded homomorphism

$$f^{\#}\colon S_Y \to S_X$$

between the homogeneous coordinate rings of Y and X or, equivalently, by a morphism of the corresponding affine cones over X and Y. Then $f^{\#}$ induces the isomorphism

$$f^*\mathcal{L}_Y \xrightarrow{\sim} \mathcal{L}_X$$

of line bundles on X; see Hartshorne [11, Chapter II, Proposition 5.12(c)]. In particular, this implies

$$\deg \mathcal{L}_X = \deg f \cdot \deg \mathcal{L}_Y.$$

We represent a finite morphism $f\colon X \to Y$ by the k-algebras $S_X^{(h)}$ and $S_Y^{(h)}$ for some $h \geq 2$, together with the k-algebra homomorphism

$$f^{\#}\colon S_Y^{(h)} \to S_X^{(h)}$$

induced by $f^{\#}\colon S_Y \to S_X$, given by a system of linear maps $\Gamma(Y, \mathcal{L}_Y^{\otimes i}) \to \Gamma(X, \mathcal{L}_X^{\otimes i})$ compatible with the multiplication maps on both sides.

In the following, when we mention a *finite morphism $f\colon X \to Y$ between projective curves*, we assume that the k-algebras $S_X^{(h)}$ and $S_Y^{(h)}$ and the homomorphism $f^{\#}\colon S_Y^{(h)} \to S_X^{(h)}$ are given for some $h \geq 2$. In the algorithms below, we will indicate when necessary how large h must be.

Remark. The homomorphism $f^{\#}$ gives rise to an injective k-linear map

$$\Gamma(Y, \mathcal{L}_Y) \longrightarrow \Gamma(X, \mathcal{L}_X).$$

Given this map we can reconstruct $S(Y)$ as a subalgebra of $S(X)$ by noting that $S(Y)$ is generated as a k-algebra by $\Gamma(Y, \mathcal{L}_Y)$.

2.6. **Images, pull-backs and push-forwards of divisors.** Let us consider a finite morphism $f\colon X \to Y$ between complete, smooth, geometrically connected curves over a field k. Such a morphism f induces various maps between the groups of divisors on X and on Y.

First, for an *effective* divisor D on X, we write $f(D)$ for the schematic image of D under f. The definition implies that the ideal sheaf $\mathcal{O}_Y(-f(D))$ is the inverse image of $f_*\mathcal{O}_X(-D)$ under the natural map $\mathcal{O}_Y \to f_*\mathcal{O}_X$.

Second, for any divisor D on X, we have the "push-forward" f_*D of D by f; see, for example, Hartshorne [11, IV, Exercise 2.6]. If P is a prime divisor on X,

then its image $f(P)$ under f is a prime divisor on Y, the residue field $k(P)$ is a finite extension of $k(f(P))$, and f_*P is given by the formula

$$f_*P = [k(P) : k(f(P))] \cdot f(P). \tag{2.7}$$

The residue field extension degree at P can simply be computed as

$$\begin{aligned}[k(P) : k(f(P))] &= \frac{[k(P) : k]}{[k(f(P)) : k]} \\ &= \frac{\deg P}{\deg f(P)}.\end{aligned}$$

Third, for any divisor E on Y, we have the "pull-back" f^*E of E by f; see, for example, Hartshorne [11, page 137]. If Q is a prime divisor on Y, then f^*Q is given by the formula

$$f^*Q = \sum_{P:\ f(P)=Q} e(P) \cdot P, \tag{2.8}$$

where P runs over the prime divisors of X mapping to Q and $e(P)$ denotes the ramification index of f at P.

We extend f_* and f^* to arbitrary divisors on X and Y by linearity. The formulae (2.7) and (2.8) imply the well-known fact that if E is any divisor on Y, we have

$$f_*f^*E = (\deg f)E.$$

Furthermore, if E is an *effective* divisor on Y, we have an equality

$$f^*E = E \times_Y X$$

of closed subschemes of X, and if $\mathcal{I}_E$ denotes the ideal sheaf of E, then its inverse image $f^{-1}\mathcal{I}_E$ is the ideal sheaf of f^*E.

Remark. The map $D \mapsto f(D)$ is not in general linear in D. We do not extend it to the divisor *group* on X, and in fact will only need schematic images of *prime* divisors on X in what follows. In contrast, the maps f_* and f^* are linear by definition.

Now let f be a finite morphism between *projective* curves in the sense of §2.5. In particular, we have a homomorphism $f^{\#}\colon S_Y \to S_X$ of graded k-algebras. We will give algorithms to compute the image and the push-forward of a divisor on X as well as the pull-back of a divisor on Y.

Algorithm 2.5 (Image of a divisor under a finite morphism)**.** *Let $f\colon X \to Y$ be a finite morphism between projective curves, let i be a positive integer, and let D be an effective divisor on X. Given the k-algebras $S_X^{(i)}$ and $S_Y^{(i)}$, the homomorphism $f^{\#}\colon S_Y^{(i)} \to S_X^{(i)}$ and the subspace $\Gamma(X, \mathcal{L}_X^{\otimes i}(-D))$ of $\Gamma(X, \mathcal{L}_X^{\otimes i})$, this algorithm outputs the subspace $\Gamma(Y, \mathcal{L}_Y^{\otimes i}(-f(D)))$ of $\Gamma(Y, \mathcal{L}_Y^{\otimes i})$.*

1. *Output the inverse image of the subspace $\Gamma(X, \mathcal{L}_X^{\otimes i}(-D))$ of $\Gamma(X, \mathcal{L}_X^{\otimes i})$ under the linear map $\Gamma(Y, \mathcal{L}_Y^{\otimes i}) \to \Gamma(X, \mathcal{L}_X^{\otimes i})$.*

Analysis. The definition of $f(D)$ implies that the line bundle $\mathcal{L}_Y^{\otimes i}(-f(D))$ equals the inverse image of $f_*\mathcal{L}_X^{\otimes i}(-D)$ under the natural map $\mathcal{L}_Y^{\otimes i} \to f_*\mathcal{L}_X^{\otimes i}$. Taking global sections, we see that $\Gamma(Y, \mathcal{L}_Y^{\otimes i}(-f(D)))$ is the inverse image of $\Gamma(X, \mathcal{L}_X^{\otimes i}(-D))$ under the natural map $\Gamma(Y, \mathcal{L}_Y^{\otimes i}) \to \Gamma(X, \mathcal{L}_X^{\otimes i})$. It is clear that the algorithm needs a number of operations in k that is polynomial in $\deg \mathcal{L}_X$ and i. ◇

Remark. In the above algorithm, we have not placed any restrictions on the degrees of D and $f(D)$. However, $f(D)$ is not uniquely determined by $\Gamma(Y, \mathcal{L}_Y^{\otimes i}(-f(D)))$ if its degree is too large.

The algorithm to compute pull-backs that we will now give is based on the fact that the pull-back of an effective divisor E is simply the fibred product $E \times_Y X$, viewed as a closed subscheme of X. In particular, the algorithm does not have to compute the ramification indices, so instead we can *use* it to compute ramification indices. Namely, if P is a prime divisor on X, we see from (2.8) that the ramification index at P equals the multiplicity with which P occurs in the divisor $f^*(f(P))$.

Algorithm 2.6 (Pull-back of a divisor under a finite morphism)**.** *Let $f\colon X \to Y$ be a finite morphism between projective curves over a field k. Let i and j be positive integers, and let E be an effective divisor on Y such that*

$$\deg f \cdot \deg E \le i \deg \mathcal{L}_X - 2g_X, \quad \deg E \le i \deg \mathcal{L}_Y - 2g_Y,$$
$$(j-i)\deg \mathcal{L}_X + \deg f \cdot \deg E \ge 2g_X - 1.$$

(If we take $j \ge i+1$, the last equality does not pose an extra restriction on E.) Given the k-algebras $S_X^{(i+j)}$ and $S_Y^{(i+j)}$, the k-algebra homomorphism $f^\#\colon S_Y^{(i+j)} \to S_X^{(i+j)}$ and the subspace $\Gamma(Y, \mathcal{L}_Y^{\otimes i}(-E))$ of $\Gamma(Y, \mathcal{L}_Y^{\otimes i})$, this algorithm outputs the subspace $\Gamma(X, \mathcal{L}_X^{\otimes i}(-f^*E))$ of $\Gamma(X, \mathcal{L}_X^{\otimes i})$.

1. *Compute the image W of $\Gamma(Y, \mathcal{L}_Y^{\otimes i}(-E))$ under the linear map*
$$f^\#\colon \Gamma(Y, \mathcal{L}_Y^{\otimes i}) \longrightarrow \Gamma(X, \mathcal{L}_X^{\otimes i}).$$
2. *Compute the space $\Gamma(X, \mathcal{L}_X^{\otimes i+j}(-f^*E))$ as the product of W and $\Gamma(X, \mathcal{L}_X^{\otimes j})$ (see Lemma 2.3).*
3. *Compute $\Gamma(X, \mathcal{L}_X^{\otimes i}(-f^*E))$ using Lemma 2.2, and output the result.*

Analysis. The ideal in S_Y defining E is generated by the linear forms vanishing on E, and the ideal of S_X defining f^*E is generated by the pull-backs of these forms. This shows that f^*E is defined by the forms in W. In the second and third step, we compute the space of all forms vanishing on f^*E, i.e., the inflation of W. That the method described is correct was proved in §2.3. The running time, measured in operations in k, is clearly polynomial in $\deg \mathcal{L}_X$, i and j. ♢

Algorithm 2.7 (Push-forward of a divisor under a finite morphism)**.** *Let $f\colon X \to Y$ be a finite morphism between projective curves over a field k satisfying the assumptions of §1.1. Let i be a positive integer, and let D be an effective divisor on X such that*

$$\deg D \le i \deg \mathcal{L}_X - 2g_X - 1 \quad \text{and} \quad \deg D \le i \deg \mathcal{L}_Y - 2g_Y.$$

Given the k-algebras $S_X^{(2i)}$ and $S_Y^{(2i)}$, the k-algebra homomorphism $f^\#\colon S_Y^{(2i)} \to S_X^{(2i)}$ and the subspace $\Gamma(X, \mathcal{L}_X^{\otimes i}(-D))$ of $\Gamma(X, \mathcal{L}_X^{\otimes i})$, this algorithm outputs the subspace $\Gamma(Y, \mathcal{L}_Y^{\otimes i}(-f_*D))$ of $\Gamma(Y, \mathcal{L}_Y^{\otimes i})$.

1. *Compute $\Gamma(X, \mathcal{L}_X^{\otimes 2i}(-D))$ as the product of $\Gamma(X, \mathcal{L}_X^{\otimes i})$ and $\Gamma(X, \mathcal{L}_X^{\otimes i}(-D))$ (see Lemma 2.1).*
2. *Find the decomposition of D as a linear combination $\sum_P n_P P$ of prime divisors using Algorithm 2.4.*

3. *For each prime divisor P in the support of D, compute $\Gamma(Y, \mathcal{L}^{\otimes i}(-f(P)))$ using Algorithm 2.5, and compute $[k(P) : k(f(P))]$.*
4. *Compute the space $\Gamma(Y, \mathcal{L}_Y^{\otimes i}(-f_*D))$, where*

$$f_*D = \sum_P n_P[k(P) : k(f(P))]f(P),$$

and output the result.

Analysis. The correctness of the algorithm follows from the definition of f_*. Its running time, for fixed k, is polynomial in $\deg \mathcal{L}_X$, i and the maximum of the bit lengths of the coefficients in the input. ♢

The following algorithm, not used in the rest of this paper but included for completeness, computes the push-forward of an effective divisor under a non-constant rational function $X \to \mathbf{P}^1$ in a slightly different setting than before. We only assume X to be given as a projective curve in the sense of §2.1, and we represent effective divisors on $\mathbf{P}^1$ as zero loci of homogeneous polynomials. For simplicity, we only consider divisors of degree at most $\deg \mathcal{L}_X$.

Algorithm 2.8 (Push-forward of an effective divisor by a rational function)**.** *Let X be a projective curve over a field k satisfying the assumptions of §1.1. Let i be a positive integer, let ψ be a non-constant rational function on X given as the quotient of two sections $s, t \in \Gamma(X, \mathcal{L}_X^{\otimes i})$ without common zeroes, and let D be an effective divisor on X of degree $d \leq \deg \mathcal{L}_X$. Given the k-algebra $S_X^{(\max\{4,i\})}$ and the subspace $\Gamma(X, \mathcal{L}_X^{\otimes 2}(-D))$, this algorithm outputs a homogeneous polynomial of degree d defining the closed subscheme ψ_*D of $\mathbf{P}^1_k$. (Such a polynomial is unique up to multiplication by elements of $k^\times$.)*

1. *Compute the space $\Gamma(X, \mathcal{L}_X^{\otimes 4}(-D))$, and use Algorithm 2.4 (with $i = 2$) to decompose D into a linear combination $D = \sum_Q n_Q Q$ of prime divisors.*
2. *For each prime divisor Q occurring in the decomposition of D:*
3. *Compute the base change $X_{k(Q)}$, where $k(Q)$ is the residue field of Q. Compute the primary decomposition of $Q_{k(Q)}$ and pick a rational point Q' in it.*
4. *Compute $\Gamma(X_{k(Q)}, \mathcal{L}_X^{\otimes 2}(-Q'))$, then compute the (one-dimensional) intersection of this space with $k{\cdot}s + k{\cdot}t$, and express some generator of this intersection as $b_Q s - a_Q t$ with $a_Q, b_Q \in k(Q)$. The element $\psi(Q') \in \mathbf{P}^1(k(Q))$ now has homogeneous coordinates $(a_Q : b_Q)$.*
5. *Compute the homogeneous polynomial*

$$f_{\psi_*Q} = \mathrm{N}_{k(Q)/k}(b_Q u - a_Q v) \in k[u, v]$$

defining ψ_*Q.

6. *Output the homogeneous polynomial*

$$f_{\psi_*D} = \prod_Q f_{\psi_*Q}^{n_Q} \in k[u, v]$$

of degree d defining ψ_*D.

Analysis. It is straightforward to check that the algorithm is correct and has running time, for fixed k, polynomial in $\deg \mathcal{L}_X$, i and the maximum of the bit lengths of the coefficients in the input. ♢

2.7. **The norm functor for effective divisors.** Let X be a complete, smooth, geometrically connected curve over a field k, and let E be an effective divisor on X. We view E as a closed subscheme of X, finite over k. In §3.6 below, we will need an explicit description of the norm functor $\mathrm{N}_{E/k}$ (for the canonical morphism $E \to \operatorname{Spec} k$) from §2.5. We view $\mathrm{N}_{E/k}$ as a functor from free $\mathcal{O}_E$-modules of rank 1 to k-vector spaces of dimension 1.

Let $\mathcal{M}$ be a line bundle on X. We abbreviate

$$\Gamma(E, \mathcal{M}) = \Gamma(E, j_E^*\mathcal{M})$$

and

$$\mathrm{N}_{E/k}\mathcal{M} = \mathrm{N}_{E/k}(j_E^*\mathcal{M}),$$

where j_E is the closed immersion of E into X. Suppose we have two line bundles $\mathcal{M}^+$ and $\mathcal{M}^-$, both of degree at least $\deg E + 2g - 1$, together with an isomorphism

$$\mathcal{M} \cong \mathcal{H}om_{\mathcal{O}_X}(\mathcal{M}^-, \mathcal{M}^+).$$

Then we can compute $\Gamma(E, \mathcal{M}^-)$ and $\Gamma(E, \mathcal{M}^+)$ using the short exact sequences

$$0 \longrightarrow \Gamma(X, \mathcal{M}^\pm(-E)) \longrightarrow \Gamma(X, \mathcal{M}^\pm) \longrightarrow \Gamma(E, \mathcal{M}^\pm) \longrightarrow 0,$$

and we can express $\mathrm{N}_{E/k}$ via the isomorphism

$$\mathrm{N}_{E/k}\mathcal{M} \cong \mathrm{Hom}_k\big(\det_k \Gamma(E, \mathcal{M}^-), \det_k \Gamma(E, \mathcal{M}^+)\big) \tag{2.9}$$

deduced from (2.6). We fix k-bases of $\Gamma(E, \mathcal{M}^-)$ and $\Gamma(E, \mathcal{M}^+)$. From the induced trivialisations of $\det_k \Gamma(E, \mathcal{M}^\pm)$ we then obtain a trivialisation of $\mathrm{N}_{E/k}\mathcal{M}$.

We now consider three line bundles $\mathcal{M}$, $\mathcal{N}$ and $\mathcal{P}$ together with the isomorphism

$$\mu\colon \mathcal{M} \otimes_{\mathcal{O}_X} \mathcal{N} \xrightarrow{\sim} \mathcal{P}.$$

By the linearity of the norm functor, μ induces an isomorphism:

$$\mathrm{N}_{E/k}\mathcal{M} \otimes_k \mathrm{N}_{E/k}\mathcal{N} \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{P}. \tag{2.10}$$

As above, we choose isomorphisms

$$\mathcal{M} \cong \mathcal{H}om_{\mathcal{O}_X}(\mathcal{M}^-, \mathcal{M}^+), \quad \mathcal{N} \cong \mathcal{H}om_{\mathcal{O}_X}(\mathcal{N}^-, \mathcal{N}^+), \quad \mathcal{P} \cong \mathcal{H}om_{\mathcal{O}_X}(\mathcal{P}^-, \mathcal{P}^+)$$

on X, where $\mathcal{M}^\pm$, $\mathcal{N}^\pm$ and $\mathcal{P}^\pm$ are line bundles of degree at least $\deg E + 2g + 1$. We fix bases of the six k-vector spaces

$$\Gamma(E, \mathcal{M}^\pm), \quad \Gamma(E, \mathcal{N}^\pm), \quad \Gamma(E, \mathcal{P}^\pm).$$

Then (2.9) gives trivialisations of $\mathrm{N}_{E/k}\mathcal{M}$, $\mathrm{N}_{E/k}\mathcal{N}$ and $\mathrm{N}_{E/k}\mathcal{P}$. Under these trivialisations, the isomorphism (2.10) equals multiplication by some element $\lambda \in k^\times$.

To find an expression for λ, we choose generators $\alpha_{\mathcal{M}}^\pm$ and $\alpha_{\mathcal{N}}^\pm$ of $\Gamma(E, \mathcal{M}^\pm)$ and $\Gamma(E, \mathcal{N}^\pm)$. To these we associate the isomorphisms

$$\alpha_{\mathcal{M}}\colon \Gamma(E, \mathcal{M}^-) \xrightarrow{\sim} \Gamma(E, \mathcal{M}^+) \quad \text{and} \quad \alpha_{\mathcal{N}}\colon \Gamma(E, \mathcal{N}^-) \xrightarrow{\sim} \Gamma(E, \mathcal{N}^+)$$

sending $\alpha_{\mathcal{M}}^-$ to $\alpha_{\mathcal{M}}^+$ and $\alpha_{\mathcal{N}}^-$ to $\alpha_{\mathcal{N}}^+$, respectively. Viewing $\alpha_{\mathcal{M}}$ and $\alpha_{\mathcal{N}}$ as generators of $\Gamma(E, \mathcal{M})$ and $\Gamma(E, \mathcal{N})$ and applying the isomorphism

$$\Gamma(E, \mathcal{M}) \otimes_{\Gamma(E, \mathcal{O}_E)} \Gamma(E, \mathcal{N}) \xrightarrow{\sim} \Gamma(E, \mathcal{P})$$

induced by μ to $\alpha_{\mathcal{M}} \otimes \alpha_{\mathcal{N}}$, we obtain a generator of $\Gamma(E, \mathcal{P})$, which we can in turn identify with an isomorphism:

$$\alpha_{\mathcal{P}}\colon \Gamma(E, \mathcal{P}^-) \xrightarrow{\sim} \Gamma(E, \mathcal{P}^+).$$

We define $\delta_{\mathcal{M}}$ as the determinant of the matrix of $\alpha_{\mathcal{M}}$ with respect to the chosen bases. Under the given trivialisations of $\mathrm{N}_{E/k}\mathcal{M}$, the element $\mathrm{N}_{E/k}^{\mathcal{M}}\alpha_{\mathcal{M}}$ corresponds to $\delta_{\mathcal{M}}$. The same goes for $\mathcal{N}$ and $\mathcal{P}$. On the other hand, the isomorphism (2.10) maps $\mathrm{N}_{E/k}^{\mathcal{M}}\alpha_{\mathcal{M}} \otimes \mathrm{N}_{E/k}^{\mathcal{N}}\alpha_{\mathcal{N}}$ to $\mathrm{N}_{E/k}^{\mathcal{P}}\alpha_{\mathcal{P}}$. We conclude that we can express λ as

$$\lambda = \frac{\delta_{\mathcal{P}}}{\delta_{\mathcal{M}}\delta_{\mathcal{N}}}. \tag{2.11}$$

Algorithm 2.9 (Linearity of the norm functor)**.** *Let X be a projective curve over a field k, and let E, D_1 and D_2 be effective divisors on X such that*

$$\deg E = \deg \mathcal{L}_X, \quad \deg D_1 = i \deg \mathcal{L}_X, \quad \deg D_2 = j \deg \mathcal{L}_X.$$

Given the k-algebra $S_X^{(i+j+4)}$, bases for the k-vector spaces

$$\begin{aligned} \Gamma(X, \mathcal{L}_X^{\otimes 2}), &\quad \Gamma(E, \mathcal{L}_X^{\otimes 2}), \\ \Gamma(X, \mathcal{L}_X^{\otimes(i+2)}(-D_1)), &\quad \Gamma(E, \mathcal{L}_X^{\otimes(i+2)}(-D_1)), \\ \Gamma(X, \mathcal{L}_X^{\otimes(j+2)}(-D_2)), &\quad \Gamma(E, \mathcal{L}_X^{\otimes(j+2)}(-D_2)), \\ \Gamma(X, \mathcal{L}_X^{\otimes(i+j+2)}(-D_1-D_2)), &\quad \Gamma(E, \mathcal{L}_X^{\otimes(i+j+2)}(-D_1-D_2)) \end{aligned}$$

and the matrices of the quotient maps

$$\begin{aligned} \Gamma(X, \mathcal{L}_X^{\otimes 2}) &\longrightarrow \Gamma(E, \mathcal{L}_X^{\otimes 2}), \\ \Gamma(X, \mathcal{L}_X^{\otimes(i+2)}(-D_1)) &\longrightarrow \Gamma(E, \mathcal{L}_X^{\otimes(i+2)}(-D_1)), \\ \Gamma(X, \mathcal{L}_X^{\otimes(j+2)}(-D_2)) &\longrightarrow \Gamma(E, \mathcal{L}_X^{\otimes(j+2)}(-D_2)), \\ \Gamma(X, \mathcal{L}_X^{\otimes(i+j+2)}(-D_1-D_2)) &\longrightarrow \Gamma(E, \mathcal{L}_X^{\otimes(i+j+2)}(-D_1-D_2)) \end{aligned}$$

with respect to the given bases, this algorithm outputs the element $\lambda \in k^\times$ such that the diagram

$$\begin{array}{ccc} k & \xrightarrow[\sim]{t_1 \otimes t_2} & \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes i}(-D_1) \otimes_k \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes j}(-D_2) \\ {\scriptstyle\lambda}\downarrow{\scriptstyle\sim} & & \downarrow{\scriptstyle\sim} \\ k & \xrightarrow[\sim]{t_3} & \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes(i+j)}(-D_1-D_2) \end{array}$$

is commutative, where

$$\begin{gathered} t_1 \colon k \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes i}(-D_1), \quad t_2 \colon k \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes j}(-D_2), \\ t_3 \colon k \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes(i+j)}(-D_1-D_2) \end{gathered}$$

are the trivialisations defined by (2.9) *using the given bases.*

1. *Compute the spaces*

$$\Gamma(E, \mathcal{L}_X^{\otimes(i+4)}(-D_1)) \quad \textit{and} \quad \Gamma(E, \mathcal{L}_X^{\otimes(i+j+4)}(-D_1-D_2))$$

and the multiplication maps

$$\begin{aligned} \Gamma(E, \mathcal{L}_X^{\otimes 2}) \times \Gamma(E, \mathcal{L}_X^{\otimes(i+2)}(-D_1)) &\to \Gamma(E, \mathcal{L}_X^{\otimes(i+4)}(-D_1)), \\ \Gamma(E, \mathcal{L}_X^{\otimes(i+2)}(-D_1)) \times \Gamma(E, \mathcal{L}_X^{\otimes(j+2)}(-D_2)) &\to \Gamma(E, \mathcal{L}_X^{\otimes(i+j+4)}(-D_1-D_2)), \\ \Gamma(E, \mathcal{L}_X^{\otimes 2}) \times \Gamma(E, \mathcal{L}_X^{\otimes(i+j+2)}(-D_1-D_2)) &\to \Gamma(E, \mathcal{L}_X^{\otimes(i+j+4)}(-D_1-D_2)). \end{aligned}$$

2. *Apply the probabilistic method described in* §1.2 *to the bilinear maps just computed to find generators* β_0, β_1 *and* β_2 *of the free* $\Gamma(E,\mathcal{O}_E)$*-modules* $\Gamma(E,\mathcal{L}_X^{\otimes 2})$, $\Gamma(E,\mathcal{L}_X^{\otimes(i+2)}(-D_1))$ *and* $\Gamma(E,\mathcal{L}_X^{\otimes(j+2)}(-D_2))$ *of rank* 1. *(If* k *is small, we may have to extend the base field, but it is easy to see that this is not a problem.)*

3. *Compute the matrix (with respect to the given bases) of the isomorphism* α_1 *defined by the commutative diagram*

$$\begin{array}{ccc} \Gamma(E,\mathcal{L}_X^{\otimes 2}) & \xrightarrow[\sim]{\alpha_1} & \Gamma(E,\mathcal{L}_X^{\otimes(i+2)}(-D_1)) \\ \| & & \sim\downarrow\cdot\beta_0 \\ \Gamma(E,\mathcal{L}_X^{\otimes 2}) & \xrightarrow[\sim]{\cdot\beta_1} & \Gamma(E,\mathcal{L}_X^{\otimes(i+4)}(-D_1)), \end{array}$$

of the isomorphism α_2 *defined by the similar diagram for* $\mathcal{L}_X^{\otimes j}(-D_2)$ *instead of* $\mathcal{L}_X^{\otimes i}(-D_1)$ *and of the isomorphism* α_3 *defined by the commutative diagram*

$$\begin{array}{ccc} \Gamma(E,\mathcal{L}_X^{\otimes 2}) & \xrightarrow[\sim]{\alpha_3} & \Gamma(E,\mathcal{L}_X^{\otimes(i+j+2)}(-D_1-D_2)) \\ \alpha_1\downarrow\sim & & \sim\downarrow\cdot\beta_0 \\ \Gamma(E,\mathcal{L}_X^{\otimes(i+2)}(-D_1)) & \xrightarrow[\sim]{\cdot\beta_2} & \Gamma(E,\mathcal{L}_X^{\otimes(i+j+4)}(-D_1-D_2)). \end{array}$$

4. *Compute the elements* δ_1, δ_2 *and* δ_3 *of* $k^\times$ *as the determinants of the matrices of* α_1, α_2 *and* α_3 *computed in the previous step.*

5. *Output the element* $\dfrac{\delta_3}{\delta_1\delta_2} \in k^\times$.

Analysis. In the notation of the above discussion, we have taken

$$\mathcal{M} = \mathcal{L}_X^{\otimes i}(-D_1), \quad \mathcal{N} = \mathcal{L}_X^{\otimes j}(-D_2), \quad \mathcal{P} = \mathcal{L}_X^{\otimes i+j}(-D_1-D_2),$$
$$\mathcal{M}^- = \mathcal{N}^- = \mathcal{P}^- = \mathcal{L}_X^{\otimes 2},$$
$$\mathcal{M}^+ = \mathcal{L}_X^{\otimes(i+2)}(-D_1), \quad \mathcal{N}^+ = \mathcal{L}_X^{\otimes(j+2)}(-D_2), \quad \mathcal{P}^+ = \mathcal{L}_X^{\otimes(i+j+2)}(-D_1-D_2).$$

Furthermore, β_0 plays the role of $\alpha^-_{\mathcal{M}}$, $\alpha^-_{\mathcal{N}}$ and $\alpha^-_{\mathcal{P}}$, and β_1, β_2 and $\beta_1\beta_2/\beta_0$ play the roles of $\alpha^+_{\mathcal{M}}$, $\alpha^+_{\mathcal{N}}$ and $\alpha^+_{\mathcal{P}}$. This means that α_1, α_2 and α_3 are equal to $\alpha_{\mathcal{M}}$, $\alpha_{\mathcal{N}}$ and $\alpha_{\mathcal{P}}$. It now follows from (2.11) that the output of the algorithm is indeed equal to λ. It is clear that the expected running time, measured in operations in k, is polynomial in $\deg \mathcal{L}_X$, i and j. ♢

2.8. **Computing in the Picard group of a curve.** We now describe how to compute with elements in the Picard group of a curve X, using the operations on divisors described in the first part of this section. We only consider the group $\mathrm{Pic}^0 X$ of isomorphism classes of line bundles of degree 0. This group can be identified canonically with a subgroup of rational points of the Jacobian variety J of X. If X has a rational point, then this subgroup consists of all the rational points of J.

We will only describe Khuri-Makdisi's *medium model* of $\mathrm{Pic}^0 X$ relative to a fixed line bundle $\mathcal{L}$ of degree

$$\deg \mathcal{L} \geq 2g+1,$$

but at the same time

$$\deg \mathcal{L} \leq c(g+1)$$

for some constant $c \geq 1$, as described in [12, §5].

Remark. Khuri-Makdisi starts with a divisor D_0 whose degree satisfies the above inequalities and takes $\mathcal{L} = \mathcal{O}_X(D_0)$. This is of course only a matter of language. Another difference in notation is that Khuri-Makdisi writes $\mathcal{L}_0$ for $\mathcal{L}$ and uses the notation $\mathcal{L}$ for $\mathcal{L}_0^{\otimes 2}$ (in the medium model) or $\mathcal{L}_0^{\otimes 3}$ (in the *large* and *small* models, which we do not describe here).

We represent elements of $\mathrm{Pic}^0 X$ by effective divisors of degree $\deg \mathcal{L}$ as follows: the isomorphism class of a line bundle $\mathcal{M}$ of degree 0 is represented by the divisor of some global section of the line bundle $\mathcal{H}om(\mathcal{M}, \mathcal{L})$ of degree $\deg \mathcal{L}$, i.e., by any effective divisor D such that

$$\mathcal{M} \cong \mathcal{L}(-D).$$

It follows from the inequality $\deg \mathcal{L} \geq 2g$ that we can represent any effective divisor D of degree $\deg \mathcal{L}$ by the subspace $\Gamma(X, \mathcal{L}^{\otimes 2}(-D))$ of codimension $\deg \mathcal{L}$ in $\Gamma(X, \mathcal{L}^{\otimes 2})$.

There are a few basic operations:

- *membership test*: given a subspace W of codimension $\deg \mathcal{L}$ in $\Gamma(X, \mathcal{L}^{\otimes 2})$, decide whether W represents an element of $\mathrm{Pic}^0 X$, i.e., whether W is of the form $\Gamma(X, \mathcal{L}^{\otimes 2}(-D))$ for an effective divisor D of degree $\deg \mathcal{L}$.
- *zero test*: given a subspace W of codimension $\deg \mathcal{L}$ in $\Gamma(X, \mathcal{L}^{\otimes 2})$, decide whether W represents the zero element of $\mathrm{Pic}^0 X$.
- *zero element*: output a subspace of codimension $\deg \mathcal{L}$ in $\Gamma(X, \mathcal{L}^{\otimes 2})$ representing the element $0 \in \mathrm{Pic}^0 X$.
- *addflip*: given two subspaces of $\Gamma(X, \mathcal{L}^{\otimes 2})$ representing elements $x, y \in \mathrm{Pic}^0 X$, compute a subspace of $\Gamma(X, \mathcal{L}^{\otimes 2})$ representing the element $-x-y$.

From the "addflip" operation, one immediately gets negation ($-x = -x - 0$), addition ($x + y = -(-x - y)$) and subtraction ($x - y = -(-x) - y$). Clearly, one can test whether two elements x and y are equal by computing $x - y$ and testing whether the result equals zero.

Remark. With regard to actual implementations of the above algorithms, we note that some of the operations can be implemented in a more efficient way than by composing the basic operations just described. We refer to [13] for details.

By Khuri-Makdisi's results in [13], the above operations can be implemented using randomised algorithms with expected running time of $O(g^{3+\epsilon})$ for any $\epsilon > 0$, measured in operations in the base field. This can be improved to $O(g^\omega)$ by means of fast linear algebra algorithms, where ω is an upper bound for the complexity of matrix multiplication.

Multiplication by a positive integer n can be done efficiently by means of an *addition chain* for n. This is a sequence of positive integers $(a_1, a_2, \ldots, a_m)$ with $a_1 = 1$ and $a_m = n$ such that for each $l > 1$ there exist $i(l)$ and $j(l)$ in $\{1, 2, \ldots, l-1\}$ such that $a_l = a_{i(l)} + a_{j(l)}$. (We consider the indices $i(l)$ and $j(l)$ as given together with the addition chain.) The integer m is called the *length* of the addition chain. Since the "addflip" operation in our set-up takes less time than addition, it is more efficient to use an *anti-addition chain*, which is a sequence of (not necessarily positive) integers $(a_0, a_1, \ldots, a_m)$ such that

$$a_l = \begin{cases} 0 & \text{if } l = 0, \\ 1 & \text{if } l = 1, \\ -a_{i(l)} - a_{j(l)} & \text{if } 2 \leq l \leq m, \end{cases}$$

and $a_m = n$; the $i(l)$ and $j(l)$ are given elements of $\{0, 1, \ldots, l-1\}$ for $2 \leq l \leq m$.

It is well known that for every positive integer n there is an addition chain of length $O(\log n)$, and there are algorithms (such as the *binary method* used in repeated squaring) to find such an addition chain in time $O((\log n)^2)$. We leave it to the reader to give a similar algorithm for finding an anti-addition chain.

For later use, we give versions of the "zero test" and "addflip" algorithms that are identical to Khuri-Makdisi's, except that some extra information computed in the course of the algorithm is part of the output.

Algorithm 2.10 (Zero test)**.** *Let X be a projective curve over a field k, and let x be an element of $\operatorname{Pic}^0 X$. Given the k-algebra $S_X^{(2)}$ and a subspace $\Gamma(\mathcal{L}_X^{\otimes 2}(-D))$ of $\Gamma(\mathcal{L}_X^{\otimes 2})$ representing x, this algorithm outputs* `false` *if $x \neq 0$ (i.e., if the line bundle $\mathcal{L}_X(-D)$ is non-trivial). If $\mathcal{L}_X(-D)$ is trivial, the algorithm outputs a pair $(\texttt{true}, s)$, where s is a global section of $\mathcal{L}_X$ with divisor D.*

1. *Compute the space*
$$\Gamma(\mathcal{L}_X(-D)) = \big\{ s \in \Gamma(\mathcal{L}_X) \;\big|\; s\Gamma(\mathcal{L}_X) \subseteq \Gamma(\mathcal{L}_X^{\otimes 2}(-D)) \big\}.$$
(The truth of this equality follows from Lemma 2.2.)
2. *If $\Gamma(\mathcal{L}_X(-D)) = 0$, output* `false`*. Otherwise, output $(\texttt{true}, s)$, where s is any non-zero element of the one-dimensional k-vector space $\Gamma(\mathcal{L}_X(-D))$.*

Algorithm 2.11 (Addflip)**.** *Let X be a projective curve over a field k, and let x and y be elements of $\operatorname{Pic}^0 X$. Given the k-algebra $S_X^{(5)}$ and subspaces $\Gamma(\mathcal{L}_X^{\otimes 2}(-D))$ and $\Gamma(\mathcal{L}_X^{\otimes 2}(-E))$ of $\Gamma(\mathcal{L}_X^{\otimes 2})$ representing x and y, this algorithm outputs a subspace $\Gamma(\mathcal{L}_X^{\otimes 2}(-F))$ representing $-x-y$, as well as a global section s of $\mathcal{L}_X^{\otimes 3}$ such that*
$$\operatorname{div} s = D + E + F.$$

1. *Compute $\Gamma(\mathcal{L}_X^{\otimes 4}(-D-E))$ as the product of $\Gamma(\mathcal{L}_X^{\otimes 2}(-D))$ and $\Gamma(\mathcal{L}_X^{\otimes 2}(-E))$ (see Lemma 2.1).*
2. *Compute the space*
$$\Gamma(\mathcal{L}_X^{\otimes 3}(-D-E)) = \big\{ s \in \Gamma(\mathcal{L}_X^{\otimes 3}) \;\big|\; s\Gamma(\mathcal{L}_X) \subseteq \Gamma(\mathcal{L}_X^{\otimes 4}(-D-E)) \big\}$$
(see Lemma 2.2).
3. *Choose any non-zero $s \in \Gamma(\mathcal{L}_X^{\otimes 3}(-D-E))$. Let F denote the divisor of s as a global section of $\mathcal{L}_X^{\otimes 3}(-D-E)$.*
4. *Compute the space*
$$\Gamma(\mathcal{L}_X^{\otimes 5}(-D-E-F)) = s\Gamma(\mathcal{L}_X^{\otimes 2}).$$
5. *Compute the space*
$$\Gamma(\mathcal{L}_X^{\otimes 2}(-F)) = \big\{ t \in \Gamma(\mathcal{L}_X^{\otimes 2}) \;\big|\; t\Gamma(\mathcal{L}_X^{\otimes 3}(-D-E)) \subseteq \Gamma(\mathcal{L}_X^{\otimes 5}(-D-E-F)) \big\}$$
(see again Lemma 2.2).
6. *Output the space $\Gamma(\mathcal{L}_X^{\otimes 2}(-F))$ and the section $s \in \Gamma(\mathcal{L}_X^{\otimes 3})$.*

2.9. **Descent of elements of the Picard group.** Let X be a projective curve over a field k in the sense of §2.1, and let O be a k-rational point of X. Let x be an element of $\mathrm{Pic}^0 X$, and let $\mathcal{M}$ be a line bundle representing x. Let $r_x^{\mathcal{L}_X,O}$ be the greatest integer r such that

$$\Gamma(\mathcal{H}om_{\mathcal{O}_X}(\mathcal{M}, \mathcal{L}_X(-rO))) \neq 0.$$

Then $\Gamma(\mathcal{H}om_{\mathcal{O}_X}(\mathcal{M}, \mathcal{L}_X(-r_x^{\mathcal{L}_X,O}O)))$ is one-dimensional, so there exists a unique effective divisor R such that

$$\mathcal{M} \cong \mathcal{L}_X(-R - r_x^{\mathcal{L}_X,O}O).$$

We define the $(\mathcal{L}_X, O)$-*normalised representative* of x as the effective divisor

$$R_x^{\mathcal{L}_X,\mathcal{O}} = R + r_x^{\mathcal{L}_X,O}O$$

of degree $\deg \mathcal{L}_X$; it is a canonically defined divisor (given $\mathcal{L}_X$ and O) with the property that x is represented by $\mathcal{L}_X(-R_x^{\mathcal{L}_X,O})$.

Remark. For any line bundle $\mathcal{N}$ we have the implications $\deg \mathcal{N} \geq g \Longrightarrow \Gamma(\mathcal{N}) \neq 0$ and $\deg \mathcal{N} < 0 \Longrightarrow \Gamma(\mathcal{N}) = 0$. The integer $r_x^{\mathcal{L}_X,O}$ therefore satisfies

$$\deg \mathcal{L}_X - g_X \leq r_x^{\mathcal{L}_X,O} \leq \deg \mathcal{L}_X.$$

Algorithm 2.12 (Normalised representative)**.** *Let X be a projective curve over a field k, and let O be a k-rational point of X. Let x be an element of* $\mathrm{Pic}^0 X$*. Given the k-algebra $S_X^{(4)}$, the spaces $\Gamma(\mathcal{L}_X^{\otimes 2}(-jO))$ for $0 \leq j \leq 2\deg \mathcal{L}_X$ and a subspace of $\Gamma(\mathcal{L}_X^{\otimes 2})$ representing x, this algorithm outputs the integer $r_x^{\mathcal{L}_X,O}$ and the subspace $\Gamma(\mathcal{L}_X^{\otimes 2}(-R_x^{\mathcal{L}_X,O}))$ of $\Gamma(\mathcal{L}_X^{\otimes 2})$.*

1. *Using the negation algorithm, find a subspace $\Gamma(\mathcal{L}_X^{\otimes 2}(-D))$ of $\Gamma(\mathcal{L}_X^{\otimes 2})$ representing $-x$.*
2. *Compute the integers*

$$r = \max\{j \geq 0 \mid \Gamma(\mathcal{L}_X^{\otimes 2}(-D)) \subseteq \Gamma(\mathcal{L}_X^{\otimes 2}(-jO))\}$$

and

$$r' = \max\{j \geq 0 \mid \Gamma(\mathcal{L}_X^{\otimes 2}(-D)) \cap \Gamma(\mathcal{L}_X^{\otimes 2}(-jO)) \neq 0\}.$$

3. *Let s be a non-zero element of $\Gamma(\mathcal{L}_X^{\otimes 2}(-D)) \cap \Gamma(\mathcal{L}_X^{\otimes 2}(-r'O))$. Compute*

$$\Gamma(\mathcal{L}_X^{\otimes 4}(-D - R_x^{\mathcal{L}_X,O})) = s\Gamma(\mathcal{L}_X^{\otimes 2})$$

and

$$\Gamma(\mathcal{L}_X^{\otimes 2}(-R_x^{\mathcal{L}_X,O})) = \left\{t \in \Gamma(\mathcal{L}_X^{\otimes 2}) \;\middle|\; t\Gamma(\mathcal{L}_X^{\otimes 2}(-D)) \subseteq \Gamma(\mathcal{L}_X^{\otimes 4}(-D - R_x^{\mathcal{L}_X,O}))\right\}.$$

4. *Output $r_x^{\mathcal{L}_X,O} = r' - r$ and $\Gamma(\mathcal{L}_X^{\otimes 2}(-R_x^{\mathcal{L}_X,O}))$.*

Analysis. We note that r is the multiplicity with which O occurs in D. This implies that for any $j \geq r$, we have

$$\Gamma(\mathcal{L}_X^{\otimes 2}(-D - (j-r)O)) = \Gamma(\mathcal{L}_X^{\otimes 2}(-D)) \cap \Gamma(\mathcal{L}_X^{\otimes 2}(-jO)).$$

Using the definitions of $r_x^{\mathcal{L}_X,O}$ and $R_x^{\mathcal{L}_X,O}$, it is now straightforward to check that the algorithm is correct and that its running time, measured in operations in k, is polynomial in $\deg \mathcal{L}_X$. ◇

Now let k' be a finite extension of k, and write

$$X' = X \times_{\operatorname{Spec} k} \operatorname{Spec} k'.$$

The natural group homomorphism

$$i\colon \operatorname{Pic}^0 X \to \operatorname{Pic}^0 X'$$

is injective since a line bundle $\mathcal{L}$ of degree 0 on X is trivial if and only if $\Gamma(X, \mathcal{L}) \neq 0$, and this is equivalent to the corresponding condition over k'. We will show how to use normalised representatives to decide whether a given element $x' \in \operatorname{Pic}^0 X'$ lies in the image of i, and if so, to find the unique element $x \in \operatorname{Pic}^0 X$ with $x' = i(x)$.

Algorithm 2.13 (Descent)**.** *Let X be a projective curve over a field k, and let O be a k-rational point of X. Let k' be a finite extension of k, write*

$$X' = X \times_{\operatorname{Spec} k} \operatorname{Spec} k',$$

and let $\mathcal{L}_{X'}$ denote the base extension of the line bundle $\mathcal{L}_X$ to X'. Let x' be an element of $\operatorname{Pic}^0 X'$. Given the k-algebra $S_X^{(4)}$, the spaces $\Gamma(X, \mathcal{L}_X^{\otimes 2}(-rO))$ for $\deg \mathcal{L}_X - g_X \leq r \leq \deg \mathcal{L}_X$ and a subspace of $\Gamma(X', \mathcal{L}_{X'}^{\otimes 2})$ representing x', this algorithm outputs `false` *if x' is not in the image of the canonical map*

$$i\colon \operatorname{Pic}^0 X \to \operatorname{Pic}^0 X'.$$

Otherwise, the algorithm outputs $(\texttt{true}, \Gamma(X, \mathcal{L}_X^{\otimes 2}(-D)))$, where $\Gamma(X, \mathcal{L}_X^{\otimes 2}(-D))$ represents the unique element $x \in \operatorname{Pic}^0 X$ such that $i(x) = x'$.

1. *Compute the $(\mathcal{L}_{X'}, O)$-normalised representative $R_{x'}^{\mathcal{L}_{X'}, O}$ of x'.*
2. *Compute the k-vector space*
$$V = \Gamma(X', \mathcal{L}_{X'}^{\otimes 2}(-R_x)) \cap \Gamma(X, \mathcal{L}_X^{\otimes 2}).$$
3. *If the codimension of V in $\Gamma(X, \mathcal{L}_X^{\otimes 2})$ is equal to $\deg \mathcal{L}_X$, output $(\texttt{true}, V)$; if not, output* `false`.

Analysis. In step 3, we check whether $R_x^{\mathcal{L}_X, O}$ is defined over k or, equivalently, whether x is defined over k. If this is the case, the space V equals $\Gamma(X, \mathcal{L}_X^{\otimes 2}(-R_x))$, where x is the unique element of $\operatorname{Pic}^0 X$ such that $i(x) = x'$. This shows that the algorithm is correct; its running time, measured in operations in k and k', is clearly polynomial in $\deg \mathcal{L}_X$. ♢

2.10. **Computing Picard and Albanese maps.** A finite morphism

$$f\colon X \to Y$$

between complete, smooth, geometrically connected curves over a field k induces two group homomorphisms

$$\operatorname{Pic} f\colon \operatorname{Pic}^0 Y \to \operatorname{Pic}^0 X \quad \text{and} \quad \operatorname{Alb} f\colon \operatorname{Pic}^0 X \to \operatorname{Pic}^0 Y,$$

called the *Picard* and *Albanese* maps, respectively. In terms of line bundles, they can be described as follows. The Picard map sends the class of a line bundle $\mathcal{N}$ on Y to the class of the line bundle $f^*\mathcal{N}$ on X, and the Albanese map sends the class of a line bundle $\mathcal{M}$ on X to the class of the line bundle $\mathrm{N}_f \mathcal{M}$ on Y.

Alternatively, these maps can be described in terms of divisor classes as follows. The group homomorphisms

$$f_*\colon \operatorname{Div}^0 X \to \operatorname{Div}^0 Y \quad \text{and} \quad f^*\colon \operatorname{Div}^0 Y \to \operatorname{Div}^0 X$$

between the groups of divisors of degree 0 on X and Y respect the relation of linear equivalence on both sides. The Picard map sends the class of a divisor E on Y to the class of the divisor f^*E on X, and the Albanese map sends the class of a divisor D on X to the class of the divisor f_*D on Y.

Let us now assume that $f\colon X \to Y$ is a finite morphism of *projective* curves in the sense of §2.5; in particular, we are given an isomorphism $f^*\mathcal{L}_Y \xrightarrow{\sim} \mathcal{L}_X$. Using the following algorithms, we can compute the maps $\operatorname{Pic} f$ and $\operatorname{Alb} f$. The algorithm for the Albanese map actually only reduces the problem to a different one, namely that of computing *traces* in Picard groups with respect to finite extensions of the base field. If A is an Abelian variety over a field k and k' is a finite extension of k, then the trace of an element $y \in A(k')$ is defined by

$$\operatorname{tr}_{k'/k} y = [k' : k]_{\mathrm{i}} \sum_{\sigma} \sigma(y),$$

where σ runs over all k-embeddings of k' into an algebraic closure of k and $[k' : k]_{\mathrm{i}}$ is the inseparable degree of k' over k. Computing traces is a problem that can be solved at least for finite fields, as we will see in §3.4.

Algorithm 2.14 (Picard map)**.** *Let $f\colon X \to Y$ be a finite morphism of projective curves, and let y be an element of $\operatorname{Pic}^0 Y$. Given the k-algebras $S_X^{(4)}$ and $S_Y^{(4)}$, the homomorphism $f^{\#}\colon S_Y^{(4)} \to S_X^{(4)}$ and a subspace $\Gamma(Y, \mathcal{L}_Y^{\otimes 2}(-E))$ of $\Gamma(Y, \mathcal{L}_Y^{\otimes 2})$ representing y, this algorithm outputs a subspace of $\Gamma(X, \mathcal{L}_X^{\otimes 2})$ representing $(\operatorname{Pic} f)(y) \in \operatorname{Pic}^0 X$.*

1. *Compute the subspace $\Gamma(X, \mathcal{L}_X^{\otimes 2}(-D))$ for the divisor $D = f^*E$ using Algorithm* 2.6 *(with $i = j = 2$), and output the result.*

Analysis. Since $(\operatorname{Pic} f)(y)$ is represented by the line bundle $\mathcal{L}_X(-f^*D)$, the correctness of this algorithm follows from that of Algorithm 2.6. Furthermore, the running time of Algorithm 2.6, measured in operations in k, is polynomial in $\deg \mathcal{L}_X$ for fixed i and j; therefore, the running time of this algorithm, measured in operations in k, is polynomial in $\deg \mathcal{L}_X$. ♢

Algorithm 2.15 (Albanese map)**.** *Let $f\colon X \to Y$ be a finite morphism of projective curves over a field k satisfying the assumptions of* §1.1*. Let x be an element of $\operatorname{Pic}^0 X$, and let O be a k-rational point of Y. The field k being fixed, suppose that given a finite extension k' of k and an element $y \in \operatorname{Pic}^0 Y_{k'}$, we can compute $\operatorname{tr}_{k'/k} y$ in time polynomial in $\deg \mathcal{L}_Y$ and $[k' : k]$. Given the k-algebras $S_X^{(6)}$ and $S_Y^{(6)}$, the homomorphism $f^{\#}\colon S_Y^{(6)} \to S_X^{(6)}$, the space $\Gamma(Y, \mathcal{L}_Y^{\otimes 2}(-O))$ and a subspace $\Gamma(X, \mathcal{L}_X^{\otimes 2}(-D))$ of $\Gamma(X, \mathcal{L}_X^{\otimes 2})$ representing x, this algorithm outputs a subspace of $\Gamma(Y, \mathcal{L}_Y^{\otimes 2})$ representing $(\operatorname{Alb} f)(x) \in \operatorname{Pic}^0 Y$.*

1. *Compute $\Gamma(X, \mathcal{L}_X^{\otimes 4}(-D))$ as the product of $\Gamma(X, \mathcal{L}_X^{\otimes 2})$ and $\Gamma(X, \mathcal{L}_X^{\otimes 2}(-D))$.*
2. *Find the decomposition of D as a linear combination $\sum_P n_P P$ of prime divisors using Algorithm* 2.4*.*
3. *For each P occurring in the support of D:*
4. *Compute the base changes $X_{k(P)}$ and $Y_{k(P)}$.*
5. *Decompose the divisor $P_{k(P)}$ on $X_{k(P)}$ as a linear combination of prime divisors using Algorithm* 2.4 *and pick a rational point P' in it.*

6. *Compute the space* $\Gamma(Y_{k(P)}, \mathcal{L}_Y^{\otimes 2}(-f(P') - (\deg \mathcal{L}_Y - 1)O))$*; this represents an element* $y_{P'} \in \operatorname{Pic}^0 Y_{k(P)}$.

7. *Compute the element* $y_P = \operatorname{tr}_{k(P)/k} y_{P'}$ *of* $\operatorname{Pic}^0 Y_{k(P)}$. *Use Algorithm* 2.13 *to get a representation for* y_P *as an element of* $\operatorname{Pic}^0 Y$.

8. *Compute the element* $y = \sum_P n_P y_P$ *of* $\operatorname{Pic}^0 Y$.

9. *Output the element* $y - (\deg f)(\deg \mathcal{L}_Y - 1)y_0$ *of* $\operatorname{Pic}^0 Y$*, where* y_0 *is the element of* $\operatorname{Pic}^0 Y$ *represented by* $\Gamma(Y, \mathcal{L}_Y^{\otimes 2}(-(\deg \mathcal{L}_Y)O))$.

Analysis. The definition of $y_{P'}$ implies that

$$y_{P'} = [\mathcal{L}_Y(-f(P') - (\deg \mathcal{L}_Y - 1)O)],$$

the definition of y_P and the definition of the trace imply that

$$y_P = [\mathcal{L}_Y^{\otimes [k(P):k]}(-f_*P - [k(P) : k](\deg \mathcal{L}_Y - 1)O)]$$

and the definition of y and the fact that $\deg \mathcal{L}_X = (\deg f)(\deg \mathcal{L}_Y)$ imply that

$$\begin{aligned} y &= [\mathcal{L}_Y^{\otimes \deg \mathcal{L}_X}(-f_*D - (\deg \mathcal{L}_X)(\deg \mathcal{L}_Y - 1)O)] \\ &= [\mathcal{L}_Y^{\otimes \deg f}(-f_*D)] + (\deg f)(\deg \mathcal{L}_Y - 1)[\mathcal{L}_Y(-(\deg \mathcal{L}_Y)O)]. \end{aligned}$$

Together with the definition of y_0, this shows that

$$\begin{aligned} y - (\deg f)(\deg \mathcal{L}_Y - 1)y_0 &= [\mathcal{L}_Y^{\otimes \deg f}(-f_*D)] \\ &= \mathrm{N}_f \mathcal{L}_X(-D), \end{aligned}$$

and therefore that the output of the algorithm is indeed $(\operatorname{Alb} f)(x)$. The running time, for fixed k, is polynomial in $\deg \mathcal{L}_X$. ♢

3. Curves over finite fields

In this section we give algorithms for computing with divisors on a curve over a finite field. After some preliminaries, we show how to compute the Frobenius map on divisors and how to choose uniformly random divisors of a given degree. Then we show how to perform various operations in the Picard group of a curve over a finite field, such as choosing random elements, computing the Frey–Rück pairing and finding a basis of the l-torsion for a prime number l. Several results in this section, especially those in §3.7–§3.9, are variants of work of Couveignes [5].

From now on, we allow the finite base field k to vary in our running time estimates. We note (cf. the remark at the end of §1.1) that elements of k can be represented by bit strings whose length is bounded by a linear function in $\log \#k$, that the usual field operations in a finite field k can be done in time polynomial in $\log \#k$, and that factoring a polynomial $f \in k[x]$ can be done in probabilistic polynomial time in $\deg f$ and $\log \#k$.

Let k be a finite field of cardinality q, and let X be a complete, smooth, geometrically connected curve of genus g over k. The *zeta function* of X is the power

series in $\mathbf{Z}[[t]]$ defined by

$$\begin{array}{ccc} \mathrm{Z}_X = \displaystyle\sum_{D \in \mathrm{Eff}\, X} t^{\deg D} & = & \displaystyle\sum_{n=0}^{\infty} (\#\,\mathrm{Eff}^n X) t^n \\ \| & & \| \\ \displaystyle\prod_{P \in \mathrm{PDiv}\, X} \frac{1}{1 - t^{\deg P}} & = & \displaystyle\prod_{d=1}^{\infty} (1 - t^d)^{-\#\,\mathrm{PDiv}^d X}. \end{array}$$

Here $\mathrm{Eff}\, X$ and $\mathrm{PDiv}\, X$ are the sets of effective divisors and prime divisors on X, respectively; a superscript denotes the subset of divisors of the indicated degree. The following properties of the zeta function are well known.

- The power series Z_X can be written as a rational function

$$\mathrm{Z}_X = \frac{L_X}{(1-t)(1-qt)} \text{ with } L_X = 1 + a_1 t + \cdots + a_{2g-1} t^{2g-1} + q^g t^{2g} \in \mathbf{Z}[t]. \tag{3.1}$$

- The factorisation of L_X over the complex numbers has the form

$$L_X = \prod_{i=1}^{2g} (1 - \alpha_i t) \quad \text{with } |\alpha_1| = \cdots = |\alpha_{2g}| = \sqrt{q}. \tag{3.2}$$

- The polynomial L_X satisfies the functional equation

$$q^g t^{2g} L_X(1/qt) = L_X(t). \tag{3.3}$$

From the definition of Z_X and (3.1) it is clear how one can compute the number of effective divisors of a given degree on X starting from the polynomial L_X. We now show how to extract the number of *prime* divisors of a given degree from L_X. Taking logarithmic derivatives in the definition of Z_X and the expression (3.1), we obtain

$$\frac{\mathrm{Z}'_X}{\mathrm{Z}_X} = \frac{1}{t} \sum_{n=1}^{\infty} \left(\sum_{d|n} d \cdot \#\,\mathrm{PDiv}^d X \right) t^n = \frac{L'_X}{L_X} + \frac{1}{1-t} + \frac{q}{1-qt}. \tag{3.4}$$

From L_X we can compute the coefficients of this power series. We can then compute $\#\,\mathrm{PDiv}^d X$ using the Möbius inversion formula. More explicitly, taking logarithmic derivatives in the factorisation (3.2), we obtain *Newton's identity*

$$L'_X / L_X = -\sum_{n=0}^{\infty} s_{n+1} t^n,$$

where the s_n are the power sums

$$s_n = \sum_{i=1}^{2g} \alpha_i^n \in \mathbf{Z} \quad (n \geq 0).$$

Expanding the right-hand side of (3.4) in a power series and comparing coefficients, we get

$$\sum_{d|n} d \,\#\,\mathrm{PDiv}^d X = q^n - s_n + 1,$$

or equivalently, by the Möbius inversion formula,

$$n \,\#\,\mathrm{PDiv}^n X = \sum_{d|n} \mu(n/d)(q^d - s_d + 1),$$

where μ is the usual Möbius function. We note that this simplifies to

$$\#\operatorname{PDiv}^n X = \begin{cases} 1+q-s_1 & \text{if } n=1, \\ \frac{1}{n}\sum_{d|n}\mu(n/d)(q^d-s_d) & \text{if } n \geq 2. \end{cases} \tag{3.5}$$

Let $J = \operatorname{Pic}^0_{X/k}$ denote the Jacobian variety of X. From the fact that the Brauer group of k vanishes it follows that the canonical inclusion

$$\operatorname{Pic}^0 X \to J(k)$$

is an equality. In other words, every rational point of J can be identified with a linear equivalence class of k-rational divisors of degree 0.

From the functional equation (3.3) one can deduce that

$$\#\operatorname{Eff}^n X = \frac{q^{1-g+n}-1}{q-1} L_X(1) \quad \text{for } n \geq 2g-1,$$

which in turn is equivalent to the "class number formula"

$$\#J(k) = \#\operatorname{Pic}^0 X = L_X(1). \tag{3.6}$$

3.1. **The Frobenius map.** Let k be a finite field of cardinality q, and let X be a projective curve over k in the sense of §2.1. We write $d = \deg \mathcal{L}_X$. Let $\operatorname{Sym}^d X$ denote the d-th symmetric power of X over k, and let $\operatorname{Gr}^d \Gamma(X, \mathcal{L}_X^{\otimes 2})$ denote the Grassmann variety of linear subspaces of codimension d in the k-vector space $\Gamma(X, \mathcal{L}_X^{\otimes 2})$. Then we have a commutative diagram:

$$\begin{array}{ccc} \operatorname{Gr}^d \Gamma(X, \mathcal{L}_X^{\otimes 2}) & \longleftarrow & \operatorname{Sym}^d X \\ {\scriptstyle \mathrm{F}_q}\downarrow & & \downarrow{\scriptstyle \mathrm{F}_q} \\ \operatorname{Gr}^d \Gamma(X, \mathcal{L}_X^{\otimes 2}) & \longleftarrow & \operatorname{Sym}^d X \end{array}$$

of varieties over k, where the vertical arrows are the q-power Frobenius morphisms. Now let k' be a finite extension of k, write

$$X' = X \times_{\operatorname{Spec} k} \operatorname{Spec} k',$$

and let D be an effective divisor on X'. The commutativity of the above diagram shows that the divisor $(\mathrm{F}_q)_* D$ on X' can be computed as follows.

Algorithm 3.1 (Frobenius map on divisors)**.** *Let X be a projective curve over a finite field k of q elements. Let k' be a finite extension of k. Let $X' = X \times_{\operatorname{Spec} k} \operatorname{Spec} k'$, and let $\mathcal{L}_{X'}$ be the base extension of the line bundle $\mathcal{L}_X$ to X'. Let i be a positive integer, and let D be an effective divisor on X'. Given the matrix M of the inclusion map*

$$\Gamma(X', \mathcal{L}_{X'}^{\otimes i}(-D)) \longrightarrow \Gamma(X', \mathcal{L}_{X'}^{\otimes i})$$

with respect to any k'-basis of the left-hand side and the k'-basis induced from any k-basis of $\Gamma(X, \mathcal{L}_X^{\otimes i})$ on the right-hand side, this algorithm outputs the analogous matrix for the inclusion map

$$\Gamma(X', \mathcal{L}_{X'}^{\otimes i}(-(\mathrm{F}_q)_* D)) \longrightarrow \Gamma(X', \mathcal{L}_{X'}^{\otimes i}).$$

1. *Apply the Frobenius automorphism of k' over k to the entries of the matrix M, and output the result.*

Analysis. It follows from the discussion preceding the algorithm that the output is indeed equal to $\Gamma(X', \mathcal{L}_{X'}^{\otimes i}(-(\mathrm{F}_q)_* D))$. The algorithm involves $O((\deg \mathcal{L}_X^{\otimes i})^2)$ computations of a q-th power of an element in k', so the running time is polynomial in $\deg \mathcal{L}_X$, i, $\log q$ and $[k' : k]$. ♢

3.2. **Choosing random prime divisors.** Let X be a projective curve (in the sense of §2.1) over a finite field. Our next goal is to generate random effective divisors of given degree on X. We start with an algorithm to generate random prime divisors. For this we do not yet need to know the zeta function of X, although we use its properties in the analysis of the running time of the algorithm.

Algorithm 3.2 (Random prime divisor)**.** *Let X be a projective curve over a finite field k. Let d and i be positive integers such that*

$$d \leq i \deg \mathcal{L}_X - 2g_X.$$

Given d, i and the k-algebra $S_X^{(2i+2)}$, this algorithm outputs a uniformly distributed prime divisor P of degree d on X, represented as the subspace $\Gamma(\mathcal{L}_X^{\otimes i}(-P))$ of $\Gamma(\mathcal{L}_X^{\otimes i})$, provided $\mathrm{PDiv}^d X$ is non-empty. (If $\mathrm{PDiv}^d X = \emptyset$, the algorithm does not terminate.)

1. *Pick a non-zero element $s \in \Gamma(\mathcal{L}_X^{\otimes i})$ uniformly randomly, and let D denote the divisor of s; this is a uniformly random hypersurface section of degree i of X.*
2. *Compute the set $\mathrm{Irr}^d D$ of (reduced) irreducible components of D of degree d over k using Algorithm* 2.4.
3. *With probability $\frac{\#\mathrm{Irr}^d D}{\lfloor (i \deg \mathcal{L}_X)/d \rfloor}$, output a uniformly random element $P \in \mathrm{Irr}^d D$ and stop.*
4. *Go to step* 1.

Analysis. We write $q = \#k$. Let H denote the set of all divisors of non-zero global sections of $\mathcal{L}_X^{\otimes i}$. The Riemann–Roch formula gives

$$\#H = \frac{q^{1-g+i \deg \mathcal{L}} - 1}{q - 1}.$$

When the algorithm finishes, the probability $p(D, P)$ that a specific pair (D, P) has been chosen is

$$\begin{aligned} p(D, P) &= \frac{1}{\#H} \frac{\#\mathrm{Irr}^d D}{\lfloor (i \deg \mathcal{L})/d \rfloor} \frac{1}{\#\mathrm{Irr}^d D} \\ &= \frac{q - 1}{q^{1-g+i \deg \mathcal{L}} - 1} \frac{1}{\lfloor (i \deg \mathcal{L})/d \rfloor}. \end{aligned}$$

For all prime divisors P of degree d, the number of $D \in H$ for which P is in the support of D is equal to

$$\#\{D \mid P \in \operatorname{supp} D\} = \frac{q^{1-g+i \deg \mathcal{L} - d} - 1}{q - 1},$$

so the probability $p(P)$ that a given P is chosen equals

$$\begin{aligned} p(P) &= \#\{D \mid P \in \operatorname{supp} D\} \cdot p(D, P) \\ &= \frac{q^{1-g+i \deg \mathcal{L} - d} - 1}{q^{1-g+i \deg \mathcal{L}} - 1} \frac{1}{\lfloor (i \deg \mathcal{L})/d \rfloor}. \end{aligned}$$

This is independent of P and therefore shows that when the algorithm finishes, the chosen element $P \in \mathrm{PDiv}^d X$ is uniformly distributed. Furthermore, the probability p that the algorithm finishes in a given iteration is

$$\begin{aligned} p &= \#\,\mathrm{PDiv}^d X \cdot \frac{q^{1-g+i\deg\mathcal{L}-d}-1}{q^{1-g+i\deg\mathcal{L}}-1}\frac{1}{\lfloor (i\deg\mathcal{L})/d\rfloor} \\ &= \frac{\#\,\mathrm{PDiv}^d X}{q^d}\frac{q^{1-g+i\deg\mathcal{L}}-q^d}{q^{1-g+i\deg\mathcal{L}}-1}\frac{1}{\lfloor (i\deg\mathcal{L})/d\rfloor} \\ &\geq \frac{\#\,\mathrm{PDiv}^d X}{q^d}(1-q^{-1-g_X})\frac{d}{i\deg\mathcal{L}}. \end{aligned}$$

We claim that the expected running time is polynomial in $\deg\mathcal{L}$, i and $\log q$, under the assumption that $\mathrm{PDiv}^d X \neq \emptyset$. Let $\sigma_0(d)$ denote the number of positive divisors of d. We distinguish two cases:

$$q^{d/2} < 2\sigma_0(d)(2g_X+1) \quad \text{and} \quad q^{d/2} \geq 2\sigma_0(d)(2g_X+1).$$

In the first case, we see that

$$p > (2\sigma_0(d)(2g_X+1))^{-2}(1-q^{-1-g_X})\frac{d}{i\deg\mathcal{L}}.$$

In the second case, we deduce from (3.5) the following estimate for $\#\,\mathrm{PDiv}^d X$:

$$\begin{aligned} |d\#\,\mathrm{PDiv}^d X - q^d| &\leq \sum_{\substack{e\mid d\\ e\neq d}} q^e + \sum_{e\mid d}|s_e| + 1 \\ &\leq (\sigma_0(d)-1)q^{d/2} + \sigma_0(d)\cdot 2g_X q^{d/2} + 1 \\ &< \sigma_0(d)(2g_X+1)q^{d/2} \\ &\leq \frac{1}{2}q^d. \end{aligned}$$

This implies that $\#\,\mathrm{PDiv}^d X > q^d/(2d)$, and hence

$$p > \frac{1-q^{-1-g_X}}{2i\deg\mathcal{L}}.$$

In both cases we see that $1/p$ is bounded by a polynomial in $\deg\mathcal{L}$ and i. We conclude that the expected running time is polynomial in $\deg\mathcal{L}$, i and $\log q$.

3.3. **Choosing random divisors.** As before, let X be a projective curve over a finite field k. From now on we assume that we know the zeta function of X or, equivalently the polynomial L_X. We will give an algorithm for generating uniformly random effective divisors of a given degree on X. These divisors will be built up from prime divisors, so we will need to consider the *decomposition type* of an effective divisor D. This is the sequence of integers $(l_1, l_2, \ldots)$, where l_d is the number of prime divisors of degree d (counted with multiplicities) occurring in D.

One of the ingredients is the concept of *m-smooth* divisors and decomposition types for integers $m \geq 0$. An m-smooth divisor is a linear combination of prime divisors whose degrees are at most m, and an m-smooth decomposition type of degree n is an m-tuple $(l_1, \ldots, l_m)$ such that $\sum_{d=1}^m l_d d = n$. For every m-smooth effective divisor D of degree n, we may view the decomposition type of D as an m-smooth decomposition type, since only its first m coefficients are non-zero.

The algorithm that we will describe takes as input integers $n \geq 0$ and $m \geq 1$, and outputs a uniformly random m-smooth effective divisor of degree n. Clearly, all effective divisors of degree n are n-smooth, so that the algorithm can be used with $m = n$ to produce uniformly random effective divisors of degree n.

The first step is to generate the decomposition type of a uniformly random m-smooth effective divisor of degree n. The method we use for doing this is described by Diem in [6, page 150] and in [7]. The algorithm works by recursion on m. For every $m \geq 1$, we write $\mathrm{Eff}^n_{\leq m} X$ for the set of m-smooth effective divisors D of degree n. Furthermore, for $l \geq 0$ and $m \geq 1$ we write $\mathrm{Eff}^{lm}_{=m} X$ for the set of divisors of degree lm that are linear combinations of prime divisors of degree m. We note that the set $\mathrm{Eff}^n_{\leq m} X$ can be decomposed as

$$\mathrm{Eff}^n_{\leq m} X = \begin{cases} \mathrm{Eff}^n_{=1} X & \text{if } m = 1, \\ \bigsqcup_{l=0}^{\lfloor n/m \rfloor} \mathrm{Eff}^{lm}_{=m} X \times \mathrm{Eff}^{n-lm}_{\leq m-1} X & \text{if } m \geq 2. \end{cases} \tag{3.7}$$

The cardinality of $\mathrm{Eff}^{lm}_{=m} X$ equals the number of ways to choose l elements from the set $\mathrm{PDiv}^m X$ with repeats. For this we have the well-known formula

$$\#\,\mathrm{Eff}^{lm}_{=m} X = \binom{\#\,\mathrm{PDiv}^m X - 1 + l}{l}. \tag{3.8}$$

Furthermore, from the description (3.7) of $\mathrm{Eff}^n_{\leq m} X$ we see that

$$\#\,\mathrm{Eff}^n_{\leq m} X = \begin{cases} \#\,\mathrm{Eff}^n_{=1} X & \text{if } m = 1, \\ \sum_{l=0}^{\lfloor n/m \rfloor} \#\,\mathrm{Eff}^{lm}_{=m} X \cdot \#\,\mathrm{Eff}^{n-lm}_{\leq m-1} X & \text{if } m \geq 2. \end{cases} \tag{3.9}$$

From these relations we can compute $\#\,\mathrm{Eff}^n_{\leq m} X$ recursively, starting from the numbers $\#\,\mathrm{PDiv}^d X$ for $1 \leq d \leq m$. An alternative way to describe these recurrence relations is to use generating functions; see Diem [6, page 149] or [7, Lemma 3.14].

In order to generate decomposition types of uniformly random m-smooth divisors of degree n, we define a probability distribution μ^n_m on the set of m-smooth decomposition types of degree n by defining $\mu^n_m(l_1, \ldots, l_m)$ as the probability that a uniformly randomly chosen effective m-smooth divisor of degree n has decomposition type $(l_1, \ldots, l_m)$. The algorithm now works as follows. We first select an integer $l_m \in \{0, 1, \ldots, \lfloor n/m \rfloor\}$—the number of prime divisors of degree m (counted with multiplicities) occurring in the decomposition—according to the marginal distribution ν^n_m of the m-th coordinate. We then apply the algorithm recursively with $(n - l_m m, m - 1)$ in place of (n, m).

The marginal distribution ν^n_m of the coordinate l_m in an m-tuple $(l_1, \ldots, l_m)$ distributed according to μ^n_m is the following. If $m = 1$, then $l_1 = n$ with probability 1. When $m \geq 2$, the probability that l_m equals a given $l \in \{0, 1, \ldots, \lfloor n/m \rfloor\}$ is

$$\nu^n_m(l) = \frac{\#\,\mathrm{Eff}^{lm}_{=m} X \cdot \#\,\mathrm{Eff}^{n-lm}_{\leq m-1} X}{\#\,\mathrm{Eff}^n_{\leq m} X} \quad (0 \leq l \leq \lfloor n/m \rfloor). \tag{3.10}$$

We compute $\#\,\mathrm{Eff}^n_{\leq m} X$, as well as $\#\,\mathrm{Eff}^{lm}_{=m}$ and $\#\,\mathrm{Eff}^{n-lm}_{\leq m-1} X$ for $0 \leq l \leq \lfloor n/m \rfloor$, using (3.5), (3.8) and (3.9). We then generate a random $l_m \in \{0, 1, \ldots, \lfloor n/m \rfloor\}$,

distributed according to ν_m^n, in the following way. We subdivide the interval

$$I = \{0, 1, \ldots, \#\operatorname{Eff}_{\leq m}^n X - 1\}$$

into $\lfloor n/m \rfloor + 1$ intervals I_l, with $0 \leq l \leq \lfloor n/m \rfloor$ and each I_l having length $\#\operatorname{Eff}_{=m}^{lm} X \cdot \#\operatorname{Eff}_{\leq m-1}^{n-lm} X$, we generate a uniformly random element $x \in I$, and we select the unique l such that $x \in I_l$.

Algorithm 3.3 (Decomposition type of a random divisor)**.** *Given the polynomial L_X for a curve X over a finite field k and integers $n \geq 0$ and $m \geq 1$, this algorithm outputs a random m-smooth decomposition type $(l_1, \ldots, l_m)$ of degree n, distributed according to the distribution μ_m^n.*

1. *If $m = 1$, output the 1-tuple (n) and stop.*
2. *Choose a random element $l_m \in \{0, 1, \ldots, \lfloor n/m \rfloor\}$ according to the distribution ν_m^n from* (3.10)*.*
3. *Call the algorithm recursively with $(n - l_m m, m - 1)$ in place of (n, m) to obtain an $(m-1)$-smooth decomposition type $(l_1, \ldots, l_{m-1})$ of degree $n - l_m m$.*
4. *Output the m-tuple $(l_1, \ldots, l_m)$.*

Analysis. The correctness of the algorithm follows from the above discussion. Its running time is polynomial in g_X, n, m and $\log \#k$. ♢

The preceding algorithm reduces our problem to generating random linear combinations of l prime divisors of a given degree d. In other words, we have to pick a random *multiset* of cardinality l from $\operatorname{PDiv}^d X$. This can be done as follows.

Algorithm 3.4 (Random multiset)**.** *Let S be a finite non-empty set of known cardinality. Suppose we have algorithms to pick uniformly random elements of S and to decide whether two such elements are equal. Given a non-negative integer l, this algorithm outputs a uniformly random multiset of l elements from S.*

1. *Generate a uniformly random subset $\{x_1, \ldots, x_l\}$ of $\{1, 2, \ldots, l + \#S - 1\}$, with $x_1 < x_2 < \cdots < x_l$.*
2. *Define a multiset $(y_1, \ldots, y_l)$ of l elements from $\{0, 1, \ldots, \#S-1\}$ by $y_i = x_i - i$; then $y_1 \leq y_2 \leq \cdots \leq y_l$.*
3. *For each i with $1 \leq i \leq l$, let a_i be the number of elements of $\{0, 1, \ldots, \#S-1\}$ that occur with multiplicity i in $(y_1, \ldots, y_l)$.*
4. *Generate a uniformly random sequence*

$$s_1^1, s_2^1, \ldots, s_{a_1}^1, \qquad s_1^2, s_2^2, \ldots, s_{a_2}^2, \qquad \ldots, \qquad s_1^l, s_2^l, \ldots, s_{a_l}^l$$

   of $a_1 + a_2 + \cdots + a_l$ distinct elements of S.
5. *Output the multiset consisting of the elements s_i^j of S, where s_i^j occurs with multiplicity j.*

Analysis. By construction, $(y_1, \ldots, y_l)$ is a uniformly random multiset of l elements chosen from $\{0, 1, \ldots, \#S - 1\}$, so the "multiplicity vector" $(a_1, \ldots, a_l)$ is the same as that of a uniformly random multiset of l elements from S. The multiset generated in step 5 is uniformly random among the multisets with this "multiplicity vector". This implies that the result is a uniformly random multiset of l elements from S, as required. The running time, measured in bit operations and operations in S (picking random elements and checking equality), is polynomial in l and $\log \#S$. ♢

Combining Algorithms 3.2, 3.3 and 3.4, we obtain the following algorithm to generate a uniformly random effective divisor of a given degree.

Algorithm 3.5 (Random divisor)**.** *Let X be a projective curve over a finite field k. Given positive integers m and i, an integer n satisfying*

$$0 \leq n \leq i \deg \mathcal{L}_X - 2g_X,$$

the graded k-algebra $S_X^{(2i+2)}$ and the polynomial L_X, this algorithm outputs a uniformly random m-smooth effective divisor D of degree n on X, represented as the subspace $\Gamma(\mathcal{L}_X^{\otimes i}(-D))$ of $\Gamma(\mathcal{L}_X^{\otimes i})$.

1. *Generate a random m-smooth decomposition type $(l_1, \ldots, l_m)$ of degree n using Algorithm* 3.3*.*
2. *For $d = 1, \ldots, m$, generate a uniformly random linear combination D_d of l_d prime divisors of degree d on X using Algorithm* 3.4 *(with $S = \mathrm{PDiv}^d X$ and $l = l_d$), where we use Algorithm* 3.2 *to generate random elements of $\mathrm{PDiv}^d X$.*
3. *Compute the subspace $\Gamma(\mathcal{L}_X(-D))$ for the divisor $D = D_1 + \cdots + D_m$ using the addition algorithm described in* §2.2*, and output $\Gamma(\mathcal{L}_X(-D))$.*

Analysis. It follows from the above discussion that the algorithm outputs a uniformly random m-smooth divisor of degree n on X. The running time is polynomial in m, n, i, $\deg \mathcal{L}_X$ and $\log \#k$. ♢

Remark. In practice, the following method for picking a random effective divisor of degree n is faster, but does not give a uniformly distributed output. Let i be a non-negative integer with $i \deg \mathcal{L} - n \geq 2g + 1$. We choose a uniformly random non-zero section $s \in \Gamma(X, \mathcal{L}^{\otimes i})$. If the set of effective divisors D of degree n with $D \leq \operatorname{div} s$ is non-empty, we pick a uniformly random element from it; otherwise we continue with a different section s.

3.4. **The Frobenius endomorphism of the Jacobian.** As before, let k be a finite field of cardinality q, and let X be a complete, smooth and geometrically connected curve over k. Let J be the Jacobian variety of X, and let F_q denote the Frobenius endomorphism of J. This is an isogeny of degree q^g, induced from the Frobenius map on X via Albanese functoriality.

Let k' be a finite extension of k', let $X' = X \times_{\operatorname{Spec} k} \operatorname{Spec} k'$, and let $x \in \mathrm{Pic}^0 X'$. The results of §3.1 imply that if D is an effective divisor of degree $\deg \mathcal{L}_X$ on X' such that $\mathcal{L}_{X'}(-D)$ represents x, we can compute $\mathrm{F}_q(x)$ by applying Algorithm 3.1 to the subspace $\Gamma(X', \mathcal{L}_{X'}^{\otimes 2}(-D))$ of the k'-vector space

$$\Gamma(X', \mathcal{L}_{X'}^{\otimes 2}) \cong k' \otimes_k \Gamma(X, \mathcal{L}_X^{\otimes 2}).$$

Algorithm 3.6 (Frobenius endomorphism of the Jacobian)**.** *Let X be a projective curve over a finite field k of q elements. Let k' be a finite extension of k. Let $X' = X \times_{\operatorname{Spec} k} \operatorname{Spec} k'$, let x be an element of $\mathrm{Pic}^0 X'$, and let D be any effective divisor of degree $\deg \mathcal{L}_X$ on X' representing x. Given the matrix M of the inclusion map*

$$\Gamma(X', \mathcal{L}_{X'}^{\otimes 2}(-D)) \longrightarrow \Gamma(X', \mathcal{L}_{X'}^{\otimes 2})$$

with respect to any k'-basis of the left-hand side and the k'-basis induced from any k-basis of $\Gamma(X, \mathcal{L}_X^{\otimes 2})$ on the right-hand side, this algorithm outputs the analogous matrix for the inclusion map

$$\Gamma(X', \mathcal{L}_{X'}^{\otimes 2}(-D')) \longrightarrow \Gamma(X', \mathcal{L}_{X'}^{\otimes 2}),$$

where D' is an effective divisor of degree $\deg \mathcal{L}_X$ on X' representing $\mathrm{F}_q x$.

1. *Apply the Frobenius automorphism of k' over k to the entries of the matrix M, and output the result.*

Analysis. This is a special case of Algorithm 3.1 and; in fact, we get $D' = (\mathrm{F}_q)_* D$. The running time of this algorithm is polynomial in $\deg \mathcal{L}_X$, $\log q$ and $[k' : k]$. ♢

Let X be a curve over a finite field k of q elements, and suppose that we have a k-rational point of X. In §2.10, the computation of Albanese maps for finite morphisms with target X was reduced to the computation of trace maps

$$\mathrm{tr}_{k'/k} \colon \mathrm{Pic}^0 X' \to \mathrm{Pic}^0 X$$

for finite extensions k'/k. We will now show how to do this in the above situation. For $x \in \mathrm{Pic}^0 X'$, we compute a subspace of $\Gamma(X', \mathcal{L}_{X'}^{\otimes 2})$ representing the element

$$y = \sum_{i=0}^{[k':k]-1} \mathrm{F}_q^i x \in \mathrm{Pic}^0 X'.$$

Now y is in fact the image of the element $\mathrm{tr}_{k'/k}\, x \in \mathrm{Pic}^0 X$ under the inclusion $\mathrm{Pic}^0 X \to \mathrm{Pic}^0 X'$, so we can apply Algorithm 2.13 with the given k-rational point to find a subspace of $\Gamma(X, \mathcal{L}_X^{\otimes 2})$ representing $\mathrm{tr}_{k'/k}\, x$.

3.5. **Picking random elements of the Picard group.** We now consider the problem of picking uniformly random elements in the finite Abelian group $J(k) = \mathrm{Pic}^0 X$. We recall from §2.8 that in the medium model of the Picard group, the class of a line bundle $\mathcal{M}$ of degree 0 is represented by an effective divisor D of degree $\deg \mathcal{L}$ such that $\mathcal{M} \cong \mathcal{L}(-D)$. It follows from the Riemann–Roch theorem and the fact that $\deg \mathcal{L} \geq 2g_X - 1$ that all fibres of the map

$$\begin{aligned} \mathrm{Eff}^{\deg \mathcal{L}} X &\longrightarrow \mathrm{Pic}^0 X \\ D &\longmapsto [\mathcal{L}(-D)] \end{aligned}$$

have cardinality $(q^{1-g+\deg \mathcal{L}} - 1)/(q-1)$. This means that to pick a uniformly random element of $\mathrm{Pic}^0 X$ it suffices to pick a uniformly random divisor of degree $\deg \mathcal{L}$.

Algorithm 3.7 (Random point of the Jacobian)**.** *Let X be a projective curve over a finite field k. Given the k-algebra $S_X^{(6)}$, this algorithm outputs a uniformly random element of $J(k) = \mathrm{Pic}^0 X$, represented by a subspace of codimension $\deg \mathcal{L}_X$ in $\Gamma(\mathcal{L}_X^{\otimes 2})$.*

1. *Using Algorithm 3.5 with $i = 2$ and $m = n = \deg \mathcal{L}_X$, generate a uniformly random divisor D of degree $\deg \mathcal{L}_X$, represented by the subspace $\Gamma(\mathcal{L}_X^{\otimes 2}(-D))$ of $\Gamma(\mathcal{L}_X^{\otimes 2})$, and output the result.*

Analysis. The correctness of the algorithm follows from that of Algorithm 3.5. Its running time is polynomial in $\deg \mathcal{L}_X$ and $\log \#k$. ♢

3.6. **Computing Frey–Rück pairings.** Let n be a positive integer. We assume that k contains a primitive n-th root of unity; this is equivalent to

$$n \mid \#k^\times = q - 1$$

and implies that n is not divisible by the characteristic of k. We write $\mu_n(k)$ for the group of n-th roots of unity in k.

Let X be a complete, smooth, geometrically connected curve over k, and let J be its Jacobian variety. The *Frey–Rück pairing* of order n on $J(k) = \mathrm{Pic}^0 X$ is a bilinear map

$$[\ ,\ ]_n \colon J[n](k) \times J(k)/nJ(k) \longrightarrow \mu_n(k)$$

that is *perfect* in the sense that it induces isomorphisms (of Abelian groups)

$$J[n](k) \xrightarrow{\sim} \mathrm{Hom}(J(k)/nJ(k), \mu_n(k)),$$
$$J(k)/nJ(k) \xrightarrow{\sim} \mathrm{Hom}(J[n](k), \mu_n(k)).$$

It can be defined as follows (see Frey and Rück [10] or Schaefer [18]). Let x and y be elements of $J(k)$ with $nx = 0$. Let D and E be divisors such that x and y are represented by the line bundles $\mathcal{O}_X(D)$ and $O_X(E)$, respectively, and such that the supports of D and E are disjoint. By assumption, there exists a rational function f on X such that $nD = \mathrm{div}(f)$; now $[x, y]_n$ is defined as

$$[x, y]_n = f(E)^{\#k^\times/n}.$$

Here $f(E)$ is defined on $\bar{k}$-valued points (where $\bar{k}$ is an algebraic closure of k) by function evaluation, and extended by linearity to a $\mathrm{Gal}(\bar{k}/k)$-equivariant homomorphism from the group of divisors on $X_{\bar{k}}$ to $\bar{k}^\times$.

Let us now give a slightly different interpretation of $f(E)$ that brings us in the right situation to compute $[x, y]_n$. We consider an arbitrary non-zero rational function f and an arbitrary divisor E such that the divisors

$$D = \mathrm{div}(f)$$

and E have disjoint supports. Since $f(E)$ is by definition linear in E, it suffices to consider the case where E is an effective divisor. As in §2.7, if $\mathcal{M}$ is a line bundle on X, we abbreviate

$$\mathrm{N}_{E/k}\mathcal{M} = \mathrm{N}_{E/k}(j_E^*\mathcal{M}),$$

where j_E is the closed immersion of E into X. Since D and E have disjoint supports, we have a canonical trivialisation

$$t_D \colon k \cong \mathrm{N}_{E/k}\mathcal{O}_X \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{O}_X(D).$$

On the other hand, multiplication by f induces an isomorphism

$$\mathrm{N}_{E/k}f \colon \mathrm{N}_{E/k}\mathcal{O}_X(D) \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{O}_X \cong k.$$

of one-dimensional k-vector spaces. We claim that the composed isomorphism

$$k \xrightarrow[\sim]{t_D} \mathrm{N}_{E/k}\mathcal{O}_X(D) \xrightarrow[\sim]{\mathrm{N}_{E/k}f} k \tag{3.11}$$

is multiplication by $f(E)$. This is true in the case where E is a single point, since then $\mathrm{N}_{E/k}$ is (canonically isomorphic to) the identity functor. We deduce the general case from this by extending the base field to an algebraic closure of k and using the fact that both $f(E)$ and the norm functor are linear in E. For the latter claim, we refer to Deligne [19, exposé XVII, no. 6.3.27].

Lemma 3.8. *Let x and y be elements of $J(k)$ with $nx = 0$, let $\mathcal{M}$ be a line bundle representing x, and let E^+ and E^- be effective divisors such that $\mathcal{O}_X(E^+ - E^-)$ represents y. (In particular, $\mathcal{M}$ has degree 0, and E^+ and E^- have the same degree.) For any pair of trivialisations*

$$t^\pm \colon k \xrightarrow{\sim} \mathrm{N}_{E^\pm/k}\mathcal{M}$$

of k-vector spaces and any trivialisation

$$s\colon \mathcal{O}_X \xrightarrow{\sim} \mathcal{M}^{\otimes n}$$

of line bundles on X, the isomorphism

$$k \xrightarrow[\sim]{(t^+)^n} \mathrm{N}_{E^+/k}\mathcal{M}^{\otimes n} \xrightarrow[\sim]{\mathrm{N}_{E^+/k}s^{-1}} k \xrightarrow[\sim]{\mathrm{N}_{E^-/k}s} \mathrm{N}_{E^-/k}\mathcal{M}^{\otimes n} \xrightarrow[\sim]{(t^-)^{-n}} k$$

is multiplication by an element of $k^\times$ whose $(\#k^\times/n)$-th power equals $[x,y]_n$.

(We have used the isomorphisms $\mathrm{N}_{E^\pm/k}(\mathcal{M}^{\otimes n}) \cong (\mathrm{N}_{E^\pm/k}\mathcal{M})^{\otimes n}$ expressing the linearity of $\mathrm{N}_{E/k}$, and denoted both sides of the isomorphism by $\mathrm{N}_{E^\pm/k}\mathcal{M}^{\otimes n}$.)

Proof. We fix a non-zero rational section h of $\mathcal{M}$ such that the divisor

$$D = \operatorname{div} h$$

is disjoint from $E^\pm$. Then we have canonical trivialisations

$$t_D^\pm\colon k \xrightarrow{\sim} \mathrm{N}_{E^\pm/k}\mathcal{O}_X(D)$$

as above. Composing these with the isomorphism

$$\mathrm{N}_{E^\pm/k}h\colon \mathrm{N}_{E^\pm/k}\mathcal{O}_X(D) \xrightarrow{\sim} \mathrm{N}_{E^\pm/k}\mathcal{M}$$

induced by multiplication by h gives trivialisations

$$t_h^\pm = \mathrm{N}_{E^\pm/k}h \circ t_D\colon k \xrightarrow{\sim} \mathrm{N}_{E^\pm/k}\mathcal{M}.$$

Now consider any isomorphism

$$s\colon \mathcal{O}_X \xrightarrow{\sim} \mathcal{M}^{\otimes n}$$

of line bundles on X, and define

$$f = s^{-1} \circ h^n\colon \mathcal{O}_X(nD) \xrightarrow{\sim} \mathcal{O}_X;$$

then f can be viewed as a rational function with divisor nD. We now have two commutative diagrams, one for each choice of the sign:

$$\begin{array}{ccccc}
k & \xrightarrow[\sim]{(t_D^\pm)^n} & \mathrm{N}_{E^\pm/k}\mathcal{O}_X(nD) & \xrightarrow[\sim]{\mathrm{N}_{E^\pm/k}f} & k \\
\| & & \sim\downarrow{\scriptstyle \mathrm{N}_{E^\pm/k}h^n} & & \| \\
k & \xrightarrow[\sim]{(t_h^\pm)^n} & \mathrm{N}_{E^\pm/k}\mathcal{M}^{\otimes n} & \xrightarrow[\sim]{\mathrm{N}_{E^\pm/k}s^{-1}} & k.
\end{array}$$

As noted above, the top row is multiplication by $f(E^\pm)$; by the commutativity of the diagram, the same holds for the bottom row. Finally, we note that replacing $t_h^\pm$ by *any* pair of trivialisations

$$t^\pm\colon k \xrightarrow{\sim} \mathrm{N}_{E^\pm/k}\mathcal{M}$$

changes the isomorphism in the bottom row of the above diagram by some n-th power in $k^\times$. This implies that the isomorphism

$$k \xrightarrow[\sim]{(t^\pm)^n} \mathrm{N}_{E^\pm/k}\mathcal{M}^{\otimes n} \xrightarrow[\sim]{\mathrm{N}_{E^\pm/k}s^{-1}} k$$

equals multiplication by an element of $k^\times$ whose $(\#k^\times/n)$-th power is $f(E^\pm)^{\#k^\times/n}$. The lemma follows from this by the definition of $[x,y]_n$. □

Algorithm 3.9 (Frey–Rück pairing)**.** *Let X be a projective curve over a finite field k, let n be an integer dividing $\#k^\times$, and let x and y be elements of $J(k)$ with $nx = 0$. Given the k-algebra $S_X^{(7)}$ and subspaces $\Gamma(\mathcal{L}_X^{\otimes 2}(-D))$ and $\Gamma(\mathcal{L}_X^{\otimes 2}(-E^-))$ of $\Gamma(\mathcal{L}_X^{\otimes 2})$ representing x and y, this algorithm outputs the element $[x,y]_n \in \mu_n(k)$.*

1. *Find an anti-addition chain $(a_0, a_1, \ldots, a_m)$ for n as described in §2.8. In particular, for $l = 2, 3, \ldots, m$ we get $i(l)$ and $j(l)$ in $\{0, 1, \ldots, l-1\}$ such that*

$$a_l = -a_{i(l)} - a_{j(l)}.$$

2. *Choose any non-zero global section u of $\mathcal{L}_X$, and let D_0 denote its divisor. Compute the space*

$$\Gamma(\mathcal{L}_X^{\otimes 2}(-D_0)) = u\Gamma(\mathcal{L}_X).$$

Write $D_1 = D$.

3. *Using Algorithm* 2.11*, find effective divisors D_2, D_3, …, D_m of degree $\deg \mathcal{L}_X$, where each D_l is represented as the space $\Gamma(\mathcal{L}_X^{\otimes 2}(-D_l))$, and non-zero global sections s_2, s_3, …, s_m of $\mathcal{L}_X^{\otimes 3}$ such that $\mathcal{L}_X^{\otimes 3}(-D_{i(l)} - D_{j(l)} - D_l)$ is trivial and*

$$\operatorname{div}(s_l) = D_{i(l)} + D_{j(l)} + D_l.$$

4. *Using Algorithm* 2.10*, verify that $\mathcal{L}_X(-D_m)$ is trivial and find a non-zero global section v of $\mathcal{L}_X(-D_m)$.*

5. *Put $E^+ = D_0$. For $E \in \{E^+, E^-\}$:*

6. *Compute the k-vector spaces*

$$\Gamma(E, \mathcal{L}_X^{\otimes 2}),$$
$$\Gamma(E, \mathcal{L}_X^{\otimes 3}(-D_l)) \text{ for } 1 \le l \le m,$$
$$\Gamma(E, \mathcal{L}_X^{\otimes 4}(-D_{i(l)} - D_{j(l)})) \text{ for } 2 \le l \le m$$

as quotients of the corresponding spaces of global sections on X. Fix a k-basis of $\Gamma(E, \mathcal{L}_X^{\otimes 3}(-D_0))$ by multiplying the chosen basis of $\Gamma(E, \mathcal{L}_X^{\otimes 2})$ by u. For $2 \le l \le m$, fix a k-basis of $\Gamma(E, \mathcal{L}_X^{\otimes 5}(-D_l - D_{i(l)} - D_{j(l)}))$ by multiplying the chosen basis of $\Gamma(E, \mathcal{L}_X^{\otimes 2})$ by s_l.

Notation: *For $0 \le l \le m$, define a trivialisation*

$$t_l(E)\colon k \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{L}_X(-D_l)$$

by (2.9) *using the fixed bases of $\Gamma(E, \mathcal{L}_X^{\otimes 2})$ and $\Gamma(E, \mathcal{L}_X^{\otimes 3}(-D_l))$. For $2 \le l \le m$, define a trivialisation*

$$t'_l(E)\colon k \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes 2}(-D_{i(l)} - D_{j(l)})$$

by (2.9) *using the fixed bases of $\Gamma(E, \mathcal{L}_X^{\otimes 2})$ and $\Gamma(E, \mathcal{L}_X^{\otimes 4}(-D_{i(l)} - D_{j(l)}))$, and define a trivialisation*

$$t''_l(E)\colon k \xrightarrow{\sim} \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes 3}(-D_l - D_{i(l)} - D_{j(l)})$$

by (2.9) *using the fixed bases of $\Gamma(E, \mathcal{L}_X^{\otimes 2})$ and $\Gamma(E, \mathcal{L}_X^{\otimes 5}(-D_l - D_{i(l)} - D_{j(l)}))$.*

7. *Put $\gamma_0(E) = \gamma_1(E) = 1$.*

8. *For $l = 2, 3, \ldots, m$:*

9. *Using Algorithm* 2.9, *compute the elements* $\lambda_l^{(1)}(E)$ *and* $\lambda_l^{(2)}(E)$ *of* $k^\times$ *such that the diagrams*

$$\begin{array}{ccc} k & \xrightarrow[\sim]{t_{i(l)}(E)\otimes t_{j(l)}(E)} & \mathrm{N}_{E/k}\mathcal{L}_X(-D_{i(l)})\otimes \mathrm{N}_{E/k}\mathcal{L}_X(-D_{j(l)}) \\ {\scriptstyle \lambda_l^{(1)}(E)}\downarrow\sim & & \downarrow\sim \\ k & \xrightarrow[\sim]{t_l'(E)} & \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes 2}(-D_{i(l)}-D_{j(l)}) \end{array}$$

and

$$\begin{array}{ccc} k & \xrightarrow[\sim]{t_l(E)\otimes t_l'(E)} & \mathrm{N}_{E/k}\mathcal{L}_X(-D_l)\otimes \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes 2}(-D_{i(l)}-D_{j(l)}) \\ {\scriptstyle \lambda_l^{(2)}(E)}\downarrow\sim & & \downarrow\sim \\ k & \xrightarrow[\sim]{t_l''(E)} & \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes 3}(-D_l-D_{i(l)}-D_{j(l)}) \end{array}$$

are commutative. Define $\lambda_l(E)=\lambda_l^{(1)}(E)\lambda_l^{(2)}(E)$.

10. *Put* $\gamma_l(E)=\dfrac{\lambda_l(E)}{\gamma_{i(l)}(E)\gamma_{j(l)}(E)}$.

11. *Compute* $\delta(E)\in k^\times$ *as the determinant of the matrix of the isomorphism*

$$v\colon \Gamma(E,\mathcal{L}_X^2)\xrightarrow{\sim}\Gamma(E,\mathcal{L}_X^3(-D_m))$$

with respect to the fixed bases.

12. *Output the element* $\left(\dfrac{\gamma_m(E^-)\delta(E^-)}{\gamma_m(E^+)\delta(E^+)}\right)^{\#k^\times/n}\in k^\times$.

Analysis. We recursively define rational sections h_1, h_2, …, h_m of $\mathcal{L}_X^{\otimes(a_l-1)}$ by

$$h_l=\begin{cases} u^{-1} & \text{for } l=0,\\ 1 & \text{for } l=1,\\ (h_{i(l)}h_{j(l)}s_l)^{-1} & \text{for } l=2,3,\dots,m.\end{cases}$$

Then it follows immediately that each h_l has divisor a_lD-D_l. The rational section

$$s=h_mv$$

of $\mathcal{L}_X^{\otimes n}$ has divisor nD, so multiplication by s induces an isomorphism

$$s\colon \mathcal{O}_X\xrightarrow{\sim}\mathcal{L}_X(-D)^{\otimes n}.$$

Let $E\in\{E^+,E^-\}$. For $2\le l\le m$, the element $\lambda_l(E)\in k^\times$ has the property that the diagram

$$\begin{array}{ccc} k & \xrightarrow[\sim]{t_l(E)\otimes t_{i(l)}(E)\otimes t_{j(l)}(E)} & \mathrm{N}_{E/k}\mathcal{L}_X(-D_l)\otimes \mathrm{N}_{E/k}\mathcal{L}_X(-D_{i(l)})\otimes \mathrm{N}_{E/k}\mathcal{L}_X(-D_{j(l)}) \\ {\scriptstyle \lambda_l(E)}\downarrow\sim & & \downarrow\sim \\ k & \xrightarrow[\sim]{t_l''(E)} & \mathrm{N}_{E/k}\mathcal{L}_X^{\otimes 3}(-D_l-D_{i(l)}-D_{j(l)}) \end{array}$$

is commutative. We note that $t_l''(E)$ is multiplication by $\mathrm{N}_{E/k}^{\mathcal{L}_X^{\otimes 3}(-D_l-D_{i(l)}-D_{j(l)})}s_l$. Using this, one proves by induction on l that the diagram

$$\begin{array}{ccc} k & \overset{t_l(E)}{\underset{\sim}{\longrightarrow}} & \mathrm{N}_{E/k}\mathcal{L}_X(-D_l) \\ {\scriptstyle\gamma_l(E)}\downarrow\sim & & \sim\downarrow{\scriptstyle\mathrm{N}_{E/k}h_l} \\ k & \overset{t_1(E)^{a_l}}{\underset{\sim}{\longrightarrow}} & \mathrm{N}_{E/k}\mathcal{L}_X(-D)^{\otimes a_l} \end{array}$$

is commutative. It follows from this and the definition of s that the diagram

$$\begin{array}{ccccc} k & \overset{\mathrm{N}_{E/k}v}{\underset{\sim}{\longrightarrow}} & \mathrm{N}_{E/k}\mathcal{L}_X(-D_m) & \overset{t_m(E)^{-1}}{\underset{\sim}{\longrightarrow}} & k \\ {\scriptstyle\mathrm{id}}\downarrow\sim & & {\scriptstyle\mathrm{N}_{E/k}h_m}\downarrow\sim & & \sim\downarrow{\scriptstyle\gamma_m(E)} \\ k & \overset{\mathrm{N}_{E/k}s}{\underset{\sim}{\longrightarrow}} & \mathrm{N}_{E/k}\mathcal{L}_X(-D)^{\otimes n} & \overset{t_1(E)^{-n}}{\underset{\sim}{\longrightarrow}} & k \end{array}$$

is commutative. The composed isomorphism in the top row is multiplication by $\delta(E)$, and hence the composed isomorphism in the bottom row is multiplication by $\gamma_m(E)\delta(E)$. The correctness of this algorithm now follows from Lemma 3.8. The running time is polynomial in $\deg \mathcal{L}_X$, $\log n$ and $\log \#k$. ♢

3.7. **Finding relations between torsion points.** Let X be a projective curve over a finite field k, represented as in §2.1, let J be its Jacobian, and let l be a prime number different from the characteristic of k. We will show how to find all the $\mathbf{F}_l$-linear relations between given elements of $J[l](k)$. In particular, given a basis $(b_1, \ldots, b_n)$ for a subspace V of $J[l](k)$ and another point $x \in J[l](k)$, this allows us to check whether $x \in V$, and if so, express x as a linear combination of $(b_1, \ldots, b_n)$.

Let k' be an extension of k containing a primitive l-th root of unity. It is well known that the problem just described can be reduced, via the Frey–Rück pairing, to the discrete logarithm problem in the group $\mu_l(k')$. Algorithm 3.11 below makes this precise. We begin with a bound on the number of elements needed to generate a finite-dimensional vector space over a finite field with high probability.

Lemma 3.10. *Let $\mathbf{F}$ be a finite field, and let V be an $\mathbf{F}$-vector space of finite dimension d. Let α be a real number with $0 < \alpha < 1$, and write*

$$m = \begin{cases} 0 & \text{if } d = 0, \\ d - 1 + \left\lceil \dfrac{\log \frac{1}{1-\alpha^{1/d}}}{\log \#\mathbf{F}} \right\rceil & \text{if } d > 0. \end{cases}$$

If $v_1, \ldots, v_m$ are uniformly random elements of V, the probability that V is generated by $v_1, \ldots, v_m$ is at least α.

Proof. Fix a basis of V. The matrix of the linear map

$$\begin{aligned} \mathbf{F}^m &\longrightarrow V \\ (c_1, \ldots, c_m) &\mapsto \sum_{i=1}^{m} c_i v_i \end{aligned}$$

is a uniformly random $d \times m$-matrix over $\mathbf{F}$. The probability that it has rank d is the probability that its rows (which are uniformly random elements of $\mathbf{F}^m$) are linearly independent. This occurs with probability

$$\begin{aligned} p &= \frac{(\#\mathbf{F}^m - 1)(\#\mathbf{F}^m - \#\mathbf{F}) \cdots (\#\mathbf{F}^m - \#\mathbf{F}^{d-1})}{\#\mathbf{F}^{dm}} \\ &\geq \frac{(\#\mathbf{F}^m - \#\mathbf{F}^{d-1})^d}{\#\mathbf{F}^{dm}} \\ &= \left(1 - (\#\mathbf{F})^{-(m-d+1)}\right)^d. \end{aligned}$$

The choice of m implies that $p \geq \alpha$. □

Remark. The integer m defined in Lemma 3.10 can be approximated independently of α by $d - 1 + \frac{\log d}{\log \#\mathbf{F}}$, in the sense that for any fixed α the difference is bounded for $d \geq 1$.

Algorithm 3.11 (Relations between torsion points)**.** *Let X be a projective curve over a finite field k, let J be its Jacobian, and let l be a prime number different from the characteristic of k. Let $x_1, \ldots, x_n$ be elements of $J[l](k)$. Given the k-algebra $S_X^{(7)}$, the polynomial L_X, and subspaces $\Gamma(\mathcal{L}_X^{\otimes 2}(-D_i))$ of $\Gamma(\mathcal{L}_X^{\otimes 2})$ representing x_i for $1 \leq i \leq n$, this algorithm outputs an $\mathbf{F}_l$-basis for the kernel of the map*

$$\begin{aligned} \Sigma\colon \mathbf{F}_l^n &\longrightarrow J[l](k) \\ (c_1, \ldots, c_n) &\longmapsto \sum_{i=1}^{n} c_i x_i. \end{aligned}$$

The algorithm depends on a parameter $\alpha \in (0, 1)$.

1. *Generate a minimal extension k' of k such that k' contains a primitive l-th root of unity ζ. Let*
$$\lambda\colon \mu_l(k') \xrightarrow{\sim} \mathbf{F}_l$$
denote the corresponding discrete logarithm, i.e., the unique isomorphism of one-dimensional $\mathbf{F}_l$-vector spaces sending ζ to 1.
2. *Define an integer $m \geq 0$ by*
$$m = \begin{cases} 0 & \text{if } n = 0, \\ n - 1 + \left\lceil \dfrac{\log \frac{1}{1-\alpha^{1/n}}}{\log l} \right\rceil & \text{if } n > 0. \end{cases}$$
3. *Choose m uniformly random elements $y_1, \ldots, y_m$ in $J(k')$ as described in §3.5; their images in $J(k')/lJ(k')$ are again uniformly distributed.*
4. *Compute the $m \times n$-matrix*
$$M = (\lambda([y_i, x_j]_l)) \quad (1 \leq i \leq m,\ 1 \leq j \leq n)$$
with coefficients in $\mathbf{F}_l$, where the pairing $[\ ,\ ]_l$ is evaluated using Algorithm 3.9 (with k' as the base field) and where the isomorphism λ is evaluated using some algorithm for computing discrete logarithms in $\mu_l(k')$.
5. *Compute an $\mathbf{F}_l$-basis $(b_1, \ldots, b_r)$ for the kernel of M.*
6. *If $\Sigma(b_1) = \cdots = \Sigma(b_r) = 0$, output $(b_1, \ldots, b_r)$ and stop.*
7. *Go to step 3.*

Analysis. We write V for the image of Σ and V' for the quotient of $J(k')/lJ(k')$ by the annihilator of V under the pairing $[\ ,\ ]_l$. Then we have an induced isomorphism

$$V \xrightarrow{\sim} \mathrm{Hom}_{\mathbf{F}_l}(V', \mu_l(k')).$$

We also consider the map

$$\begin{aligned} \Sigma' \colon \mathbf{F}_l^m &\longrightarrow V' \\ (c_1, \ldots, c_m) &\longmapsto \sum_{i=1}^{m} c_i y_i. \end{aligned}$$

There is a commutative diagram

$$\begin{array}{ccc} \mathbf{F}_l^n & \longrightarrow & \mathrm{Hom}_{\mathbf{F}_l}(\mathbf{F}_l^m, \mu_l(k')) \\ {\scriptstyle\Sigma}\downarrow & & \uparrow{\scriptstyle f \mapsto f \circ \Sigma'} \\ V & \xrightarrow{\sim} & \mathrm{Hom}_{\mathbf{F}_l}(V', \mu_l(k')). \end{array}$$

We identify $\mu_l(k')$ with $\mathbf{F}_l$ using the isomorphism λ and equip $\mathrm{Hom}_{\mathbf{F}_l}(\mathbf{F}_l^m, \mu_l(k'))$ with the dual basis of the standard basis of $\mathbf{F}_l^m$. Then the top arrow in the diagram is given by the matrix M defined in step 4. This means that we have an inclusion

$$\ker \Sigma \subseteq \ker M.$$

In step 6 we check whether this inclusion is an equality. By the surjectivity of Σ, this is the case if and only if the rightmost map in the diagram is injective, i.e., if and only if Σ' is surjective. Since $\dim_{\mathbf{F}_l} V \leq n$, this happens with probability at least α by Lemma 3.10; hence steps 3–7 are executed at most $1/\alpha$ times on average. This implies that for fixed α, the running time is polynomial in g_X, l, n and $\log \#k$. ♢

Remarks. (1) If we know an upper bound for the dimension of the $\mathbf{F}_l$-vector space generated by the x_i, then we can use this upper bound instead of n in the expression for m in step 2.
(2) It matters little what algorithm we use for computing the discrete logarithm in $\mu_l(k')$, since the generating k' already makes running time of Algorithm 3.11 polynomial in l. For example, we can simply tabulate the function λ.

3.8. **The Kummer map on a divisible group.** Let k be a finite field of q elements, and let l be a prime number. Let $\mathbf{G}$ be an étale l-divisible group over k. (The étaleness is automatic if l is invertible in k.) We denote by $\mathrm{F}_q \colon \mathbf{G} \to \mathbf{G}$ the (q-power) Frobenius endomorphism of $\mathbf{G}$; this is an automorphism because of the assumption that $\mathbf{G}$ is étale.

For any non-negative integer n such that all the points of $\mathbf{G}[l^n]$ are k-rational, the *Kummer map* of order l^n on $\mathbf{G}$ over k is the homomorphism

$$\begin{aligned} K_{l^n}^{\mathbf{G}/k} \colon \mathbf{G}(k)/l^n\mathbf{G}(k) &\longrightarrow \mathbf{G}[l^n](k) \\ x &\longmapsto \mathrm{F}_q(y) - y, \end{aligned}$$

where y is any point of $\mathbf{G}$ over an algebraic closure of k such that the image of $l^n y$ in $\mathbf{G}(k)/l^n\mathbf{G}(k)$ equals x.

Let $\chi \in \mathbf{Z}_l[t]$ be the characteristic polynomial of the Frobenius automorphism of $\mathbf{G}$ on the Tate module of $\mathbf{G}$. Then the element $t \bmod \chi$ of $\mathbf{Z}_l[t]/(\chi)$ is invertible. Let n be any non-negative integer, and let a be a positive integer such that

$$t^a = 1 \quad \text{in } (\mathbf{Z}_l[t]/(l^n, \chi))^\times.$$

Then $t^a - 1$ is divisible by l^n in $\mathbf{Z}_l[t]/(\chi)$, and we let h_a be the unique element of $\mathbf{Z}_l[t]/(\chi)$ such that

$$t^a - 1 = l^n h_a \in \mathbf{Z}_l[t]/(\chi).$$

By the Cayley–Hamilton theorem, $\mathbf{Z}_l[t]/(\chi)$ acts on $\mathbf{G}$ with t acting as F_q. The above identity therefore implies that

$$\mathrm{F}_q^a - 1 = l^n h_a(\mathrm{F}_q) \quad \text{on } \mathbf{G}.$$

Let k_a be an extension of k with

$$[k_a : k] = a.$$

Then $\mathbf{G}[l^n]$ is defined over k_a, and we can express the Kummer map over k_a in terms of the Frobenius endomorphism over k as

$$\begin{aligned} K_{l^n}^{\mathbf{G}/k_a} \colon \mathbf{G}(k_a)/l^n\mathbf{G}(k_a) &\longrightarrow \mathbf{G}[l^n](k_a) \\ x &\longmapsto h_a(\mathrm{F}_q)(x). \end{aligned}$$

In §3.9 we are going to apply this to a certain l-divisible subgroup of the l-power torsion of the Jacobian of a projective curve over k.

3.9. **Computing the l-torsion in the Picard group.** Let X be a projective curve over k, represented as in §2.1, and let J be its Jacobian. Let F_q denote the Frobenius endomorphism of J over k, and let $\chi \in \mathbf{Z}[t]$ be the characteristic polynomial of F_q.

Let l be a prime number different from the characteristic of k. We are going to apply the results of §3.8 to a certain l-divisible subgroup $\mathbf{G}$ of the group $J[l^\infty]$ of l-power torsion points of J. This $\mathbf{G}$ is defined as follows. Let $\bar{f} = (t-1)^b$ be the largest power of $t-1$ dividing $\chi \bmod l$, so that $\chi \bmod l$ has the factorisation

$$(\chi \bmod l) = \bar{f} \cdot \bar{f}^\perp$$

in coprime monic polynomials in $\mathbf{F}_l[t]$. Hensel's lemma implies that this factorisation can be lifted uniquely to a factorisation

$$\chi = f \cdot f^\perp,$$

where f and $f^\perp$ are coprime monic polynomials in $\mathbf{Z}_l[t]$. The Chinese remainder theorem gives a decomposition

$$\mathbf{Z}_l[t]/(\chi) \xrightarrow{\sim} \mathbf{Z}_l[t]/(f) \times \mathbf{Z}_l[t]/(f^\perp), \tag{3.12}$$

which in turn induces a decomposition

$$J[l^\infty] \cong \mathbf{G} \times \mathbf{G}^\perp$$

of l-divisible groups. We note that $\mathbf{G}$ is of rank b and that f is the characteristic polynomial of F_q on $\mathbf{G}$. Let a be a positive integer such that

$$t^a = 1 \quad \text{in } (\mathbf{F}_l[t]/\bar{f})^\times, \tag{3.13}$$

let h_a be the unique element of $\mathbf{Z}_l[t]/(f)$ such that

$$t^a - 1 = l h_a \in \mathbf{Z}_l[t]/(f), \tag{3.14}$$

and let k_a be an extension of degree a of k. All the points of $\mathbf{G}[l]$ are k_a-rational, and the b-dimensional $\mathbf{F}_l$-vector space $\mathbf{G}[l](k_a)$ is the generalised eigenspace corresponding to the eigenvalue 1 of F_q inside the $\mathbf{F}_l$-vector space of points of $J[l]$ over an algebraic closure of k_a. In particular, we have the identity

$$J[l](k) = \{x \in \mathbf{G}[l](k_a) \mid \mathrm{F}_q(x) = x\}.$$

As explained in §3.8, the map

$$\begin{aligned}\mathbf{G}(k_a)/l\mathbf{G}(k_a) &\longrightarrow \mathbf{G}[l](k_a)\\ x &\longmapsto h_a(\mathrm{F}_q)(x)\end{aligned}$$

is well defined and equal to the Kummer map

$$K_l^{\mathbf{G}/k_a}\colon \mathbf{G}(k_a)/l\mathbf{G}(k_a) \longrightarrow \mathbf{G}[l](k_a)$$

of order l. It is in fact an isomorphism; this follows for example from the finiteness of $\mathbf{G}(k_a)$.

Let us now explain how to use the above results to generate uniformly random elements of the $\mathbf{F}_l$-vector space $\mathbf{G}[l](k_a)$. We factor $\#J(k_a)$ as

$$\#J(k_a) = l^{c_a} m_a \quad \text{with } c_a \geq 0,\ m_a \geq 1 \text{ and } l \nmid m_a.$$

Let e be the idempotent in $\mathbf{Z}_l[t]/(\chi)$ corresponding to the element $(1,0)$ on the right-hand side of (3.12). Composing the maps

$$J(k_a) \xrightarrow{m_a} J[l^\infty](k_a) \xrightarrow{e(\mathrm{F}_q)} \mathbf{G}(k_a) \longrightarrow \mathbf{G}(k_a)/l\mathbf{G}(k_a) \xrightarrow{h_a(\mathrm{F}_q)} \mathbf{G}[l](k_a) \tag{3.15}$$

we get a surjective group homomorphism from $J(k_a)$ to $\mathbf{G}[l](k_a)$. We can use this map to convert uniformly random elements of $J(k_a)$ into uniformly random elements of $\mathbf{G}[l](k_a)$, provided we know e and h_a to sufficient l-adic precision. It is clear that to compute the Kummer map we only need to know the image of h_a in $\mathbf{Z}_l[t]/(f,l) = \mathbf{F}_l[t]/((t-1)^b)$. Since $\mathbf{G}(k_a)$ can be identified with a subgroup of $\#J(k_a)$, it is annihilated by l^{c_a}, and we have

$$J[l^\infty](k_a) = J[l^{c_a}](k_a) \quad \text{and} \quad \mathbf{G}(k_a) = \mathbf{G}[l^{c_a}](k_a).$$

This implies that it suffices to know e to precision $O(l^{c_a})$.

Let us check that there is a reasonably small a for which (3.13) holds. For any non-negative integer γ the identity

$$t^{l^\gamma} - 1 = (t-1)^{l^\gamma}$$

holds in $\mathbf{F}_l[t]$, and the right-hand side maps to zero in $\mathbf{F}_l[t]/(t-1)^b$ if and only if $l^\gamma \geq b$. Since l is a prime number, we conclude that the order of t in $\mathbf{F}_l[t]/((t-1)^b)$ equals l^γ, where γ is the least non-negative integer such that $l^\gamma \geq b$.

Algorithm 3.12 (Basis for the l-torsion of the Picard group)**.** *Let X be a projective curve over a finite field k with q elements, let J be its Jacobian, and let l be a prime number different from the characteristic of k. Given the k-algebra $S_X^{(7)}$ and the characteristic polynomial χ of the Frobenius endomorphism of J over k, this algorithm outputs an $\mathbf{F}_l$-basis for $J[l](k) = (\operatorname{Pic} X)[l]$. The algorithm depends on a parameter $\alpha \in (0,1)$.*

1. *Compute the greatest integer b such that $(t-1)^b$ divides $\chi \bmod l$ in $\mathbf{F}_l[t]$. Compute the non-negative integer r defined by*

$$r = \begin{cases} 0 & \text{if } b = 0,\\ b - 1 + \left\lceil \dfrac{\log \frac{1}{1-\alpha^{1/b}}}{\log l} \right\rceil & \text{if } b \geq 1. \end{cases}$$

2. *Define $a = l^\gamma$, where γ is the least non-negative integer such that $l^\gamma \geq b$. Generate a finite extension k_a of degree a of k. Factor $\#J(k_a)$ as*

$$\#J(k_a) = l^{c_a} m_a \quad \text{with } l \nmid m_a.$$

3. *Lift the factorisation*

$$(\chi \bmod l) = \bar{f} \cdot \bar{f}^{\perp} \quad \text{in } \mathbf{F}_l[t],$$

where $\bar{f} = (t-1)^b$*, to a factorisation*

$$(\chi \bmod l^{c_a}) = f \cdot f^{\perp}$$

into coprime monic polynomials in $(\mathbf{Z}/l^{c_a}\mathbf{Z})[t]$*. Compute the image of the idempotent* e *in* $(\mathbf{Z}/l^{c_a}\mathbf{Z})[t]/(\chi)$ *using the extended Euclidean algorithm, and compute the image of* h_a *in* $\mathbf{F}_l[t]/((t-1)^b)$ *using the definition* (3.14) *of* h_a*.*

4. *Generate* r *uniformly random elements of* $J(k_a)$ *using Algorithm* 3.7*, and map them to elements* $x_1, \ldots, x_r \in \mathbf{G}[l](k_a)$ *using the homomorphism* (3.15)*.*

5. *Using Algorithm* 3.11*, compute a basis for the kernel of the* $\mathbf{F}_l$*-linear map*

$$\Sigma\colon \mathbf{F}_l^r \longrightarrow \mathbf{G}[l](k_a)$$
$$(c_1, \ldots, c_r) \longmapsto \sum_{i=1}^{r} c_i x_i.$$

If the dimension of this kernel is greater than $r - b$*, go to step* 4*.*

6. *Use the* $\mathbf{F}_l$*-linear relations between* $x_1, \ldots, x_r$ *computed in the previous step to find a subsequence* $(y_1, \ldots, y_b)$ *of* $(x_1, \ldots, x_r)$ *that is an* $\mathbf{F}_l$*-basis of* $\mathbf{G}[l](k_a)$*.*

7. *Let* M *be the matrix with respect to the basis* $(y_1, \ldots, y_b)$ *of the* $\mathbf{F}_l$*-linear automorphism of* $\mathbf{G}[l](k_a)$ *induced by* F_q*. Compute* M *by computing* $\mathrm{F}_q(y_i)$ *for* $i = 1, \ldots, b$ *using Algorithm* 3.1 *and then applying Algorithm* 3.11 *to express the* $\mathrm{F}_q(y_i)$ *as linear combinations of the* y_i*.*

8. *Compute a basis for the kernel of* $M - I$*, where* I *is the* $b \times b$ *identity matrix. Map the basis elements to elements* $z_1, \ldots, z_t$ *of* $\mathbf{G}[l](k_a)$ *using the injective homomorphism*

$$\mathbf{F}_l^b \longrightarrow \mathbf{G}[l](k_a)$$
$$(a_1, \ldots, a_b) \longmapsto \sum_{i=1}^{b} a_i y_i.$$

Output $(z_1, \ldots, z_t)$*.*

Analysis. As noted earlier, the definition of a implies that a equals the order of t in $(\mathbf{F}_l[t]/(t-1)^b)^{\times}$; furthermore, $J[l](k)$ equals the kernel of $\mathrm{F}_q - \mathrm{id}$ on $\mathbf{G}[l](k_a)$. The elements $x_1, \ldots, x_r$ of $\mathbf{G}[l](k_a)$ are uniformly random by the fact that (3.15) is a homomorphism. By Lemma 3.10, they generate the b-dimensional $\mathbf{F}_l$-vector space $\mathbf{G}[l](k_a)$ with probability at least α. The definition of a also implies that

$$a \leq \max\{1, 2g_X l - 1\},$$

while the "class number formula" (3.6) gives the upper bound

$$c_a \leq \frac{\log \#J(k_a)}{\log l} \leq \frac{2g_X \log\bigl(1 + q^{a/2}\bigr)}{\log l}.$$

This shows that c_a is bounded by a polynomial in g_X, $\log q$ and l. For fixed α we therefore reach step 6 in expected polynomial time in $\deg \mathcal{L}_X$, $\log q$ and l. In steps 6–8 we compute a basis for the kernel of $\mathrm{F}_q - \mathrm{id}$, which is $J[l](k)$. We conclude that

the algorithm is correct and, for fixed α, runs in probabilistic polynomial time in $\deg \mathcal{L}_X$, $\log q$ and l. ♢

Remark. The elements $z_j \in J[l](k_a)$ output by the preceding algorithm are in fact defined over k. In general, I do not know how to generate k-vector spaces (instead of k_a-vector spaces) representing them. However, if we know a k-rational point on X, then we can use Algorithm 2.13 to accomplish this.

Acknowledgements

I would like to thank Johan Bosman, Claus Diem, Bas Edixhoven, Robin de Jong, Kamal Khuri-Makdisi and Hendrik Lenstra for useful comments, conversations and correspondence on topics related to this paper. I also thank the referee for a number of helpful remarks.

References

[1] L. M. Adleman and H. W. Lenstra, Jr., Finding irreducible polynomials over finite fields. In: *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing (Berkeley, CA, 1986)*, 350–355. Association for Computing Machinery, New York, 1986.

[2] J. G. Bosman, *Explicit computations with modular Galois representations*. Proefschrift (Ph.D. thesis), Universiteit Leiden, 2008.

[3] P. J. Bruin, *Modular curves, Arakelov theory, algorithmic applications*. Proefschrift (Ph.D. thesis), Universiteit Leiden, 2010.

[4] G. Castelnuovo, Sui multipli di una serie lineare di gruppi di punti appartenente ad una curva algebrica. *Rendiconti del Circolo Matematico di Palermo* **7** (1893), 89–110. (= *Memorie scelte*, 95–113. Zanichelli, Bologna, 1937.)

[5] J.-M. Couveignes, Linearizing torsion classes in the Picard group of algebraic curves over finite fields. *Journal of Algebra* **321** (2009), 2085–2118. MR2501511 (2010e:14019)

[6] C. Diem, *On arithmetic and the discrete logarithm problem in class groups of curves*. Habilitationsschrift, Universität Leipzig, 2008.

[7] C. Diem, On the discrete logarithm problem in class groups of curves. *Mathematics of Computation* **80** (2011), 443–475. MR2728990 (2011j:11242)

[8] W. Eberly and M. Giesbrecht, Efficient decomposition of associative algebras over finite fields. *Journal of Symbolic Computation* **29** (2000), 441–458. MR1751390 (2001a:16079)

[9] S. J. Edixhoven and J.-M. Couveignes (with R. S. de Jong, F. Merkl and J. G. Bosman), *Computational Aspects of Modular Forms and Galois Representations*. Annals of Mathematics Studies **176**, Princeton University Press, 2011.

[10] G. Frey and H.-G. Rück, A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Mathematics of Computation* **62** (1994), 865–874. MR1218343 (94h:11056)

[11] R. Hartshorne, *Algebraic Geometry*. Graduate Texts in Mathematics **52**, Springer-Verlag, New York, 1977. MR0463157 (57:3116)

[12] K. Khuri-Makdisi, Linear algebra algorithms for divisors on an algebraic curve. *Mathematics of Computation* **73** (2004), no. 245, 333–357. MR2034126 (2005a:14081)

[13] K. Khuri-Makdisi, Asymptotically fast group operations on Jacobians of general curves. *Mathematics of Computation* **76** (2007), no. 260, 2213–2239. MR2336292 (2009a:14072)

[14] R. Lazarsfeld, A sampling of vector bundle techniques in the study of linear series. In: M. Cornalba, X. Gomez-Mont and A. Verjovsky (editors), *Lectures on Riemann Surfaces (Trieste, 1987)*, 500–559. World Scientific Publishing, Teaneck, NJ, 1989. MR1082360 (92f:14006)

[15] A. Mattuck, Symmetric products and Jacobians. *American Journal of Mathematics* **83** (1961), no. 1, 189–206. MR0142553 (26:122)

[16] D. Mumford, Varieties defined by quadratic equations. With an appendix by G. Kempf. In: *Questions on Algebraic Varieties* (Centro Internationale Matematico Estivo, 3° ciclo, Varenna, 1969), 29–100. Edizioni Cremonese, Roma, 1970. MR0282975 (44:209)

[17] M. O. Rabin, Probabilistic algorithms in finite fields. *SIAM Journal on Computing* **9** (1980), no. 2, 273–280. MR568814 (81g:12002)

[18] E. F. Schaefer, A new proof for the non-degeneracy of the Frey–Rück pairing and a connection to isogenies over the base field. In: T. Shaska (editor), *Computational Aspects of Algebraic Curves* (Conference held at the University of Idaho, 2005), 1–12. Lecture Notes Series in Computing **13**, World Scientific Publishing, Hackensack, NJ, 2005. MR2181869 (2006i:14019)

[19] *Théorie des topos et cohomologie étale des schémas* (SGA 4). Tome 3 (exposés IX à XIX). Séminaire de Géométrie Algébrique du Bois-Marie 1963–1964, dirigé par M. Artin, A. Grothendieck et J.-L. Verdier, avec la collaboration de P. Deligne et B. Saint-Donat. Lecture Notes in Mathematics **305**, Springer-Verlag, Berlin/Heidelberg/New York, 1973. MR0354654 (50:7132)

[20] W. A. Stein, *Modular Forms, a Computational Approach*. With an appendix by P. E. Gunnells. Graduate Studies in Mathematics **79**, American Mathematical Society, Providence, RI, 2007. MR2289048 (2008d:11037)

Institut für Mathematik, Universität Zürich, Winterthurerstrasse 190, CH-8057 Zürich

E-mail address: peter.bruin@math.uzh.ch