J. Cryptology (2006) 19: 1–25
DOI: 10.1007/s00145-004-0231-y

Journal of
CRYPTOLOGY

An Extension of Kedlaya's Algorithm to Hyperelliptic Curves in Characteristic 2*

Jan Denef
Department of Mathematics, University of Leuven,
Celestijnenlaan 200B, B-3001 Leuven-Heverlee, Belgium
jan.denef@wis.kuleuven.ac.be

Frederik Vercauteren
Department of Electrical Engineering, University of Leuven,
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium
frederik.vercauteren@esat.kuleuven.ac.be
and
Computer Science Department, University of Bristol,
Woodland Road, Bristol BS8 1UB, England
frederik@cs.bris.ac.uk

Communicated by Johannes Buchmann

Received 19 November 2002 and revised 6 May 2004
Online publication 23 February 2005

Abstract. We present an algorithm to compute the zeta function of an arbitrary hyperelliptic curve over a finite field $\mathbb{F}_q$ of characteristic 2, thereby extending the algorithm of Kedlaya for odd characteristic. Given a genus g hyperelliptic curve defined over $\mathbb{F}_{2^n}$, the average-case time complexity is $O(g^{4+\varepsilon}n^{3+\varepsilon})$ and the average-case space complexity is $O(g^3n^3)$, whereas the worst-case time and space complexities are $O(g^{5+\varepsilon}n^{3+\varepsilon})$ and $O(g^4n^3)$, respectively.

Key words. Hyperelliptic curves, Cryptography, Kedlaya's algorithm, Monsky–Washnitzer cohomology.

1. Introduction

Since elliptic curve cryptosystems were introduced by Koblitz [20] and Miller [29], various other systems based on the discrete logarithm problem in the Jacobian of curves have been proposed, such as hyperelliptic curves [21], superelliptic curves [14] and $\mathcal{C}_{ab}$ curves [2]. One of the main initialisation steps of these cryptosystems is to generate a suitable curve defined over a given finite field. To ensure the security of the system, the

* The second author was supported in part by the Fund for Scientific Research - Flanders (Belgium), Grant G.0186.02.

curve must be chosen such that the group order of the Jacobian is divisible by a large prime.

Currently, there exist several approaches for computing the number of points on the Jacobian of random curves. The first method is l-adic in nature: the number of points is computed modulo sufficient small primes l by working in l-torsion subgroups of the Jacobian and the final result is determined using the Chinese remainder theorem. This approach was first described by Schoof [38] for elliptic curves and leads to a polynomial time algorithm in all characteristics. A detailed description of Schoof's algorithm and the improvements by Atkin [3] and Elkies [9] can be found in [4] and [25]. Pila [35] and later Adleman and Huang [1] extended Schoof's algorithm to higher genus curves. Currently, only the genus 2 version of this algorithm is practical [16], [17].

The second approach is p-adic in nature and is especially efficient for algebraic varieties over finite fields of small characteristic. These p-adic algorithms come in two flavours. The first strategy computes a p-adic approximation of the Serre–Tate canonical lift and the action of Frobenius on this lift. This approach was first described by Satoh [36] for elliptic curves. An overview of the many variants and further optimisations of Satoh's algorithm can be found in [41]. Mestre [27] presented a "dual" algorithm using the Arithmetic–Geometric mean and sketched how it could be extended to ordinary hyperelliptic curves [28]. Results by Lercier and Lubicz [26] show that this algorithm is very efficient as long as the genus is small; this is due to the exponential dependence on the genus.

The second strategy computes the action of Frobenius on p-adic cohomology groups. Kedlaya [19] described such an algorithm for hyperelliptic curves over finite fields of small odd characteristic, using the theory of Monsky–Washnitzer cohomology. The running time of the algorithm is $O(g^{4+\varepsilon}n^{3+\varepsilon})$ for a hyperelliptic curve of genus g over $\mathbb{F}_{p^n}$. The algorithm readily generalises to superelliptic curves as shown by Gaudry and Gürel [15]. A related approach by Lauder and Wan [22] is based on Dwork's proof of the rationality of the zeta function and results in a polynomial time algorithm to compute the zeta function of an arbitrary algebraic variety over a finite field. Despite its polynomial time complexity, a first implementation indicates that cryptographical sizes are out of reach. Note that Wan [42] already suggested the use of p-adic methods, including the methods of Dwork and Monsky, several years ago. Using Dwork cohomology, Lauder and Wan [23] specialised their original algorithm to curves which are Artin–Schreier covers of the affine line minus one point, leading to an $O(g^{5+\varepsilon}n^{3+\varepsilon})$ time algorithm. In [7] we described an extension of Kedlaya's algorithm to the same class of curves in characteristic 2 with the same time complexity. More recently, Lauder and Wan [24] extended their work to a larger class of Artin–Schreier covers (that does not however include all hyperelliptic curves in characteristic 2).

In this paper we extend Kedlaya's algorithm to arbitrary hyperelliptic curves defined over a finite field of characteristic 2. Given a genus g hyperelliptic curve defined over $\mathbb{F}_{2^n}$, the average-case time complexity is $O(g^{4+\varepsilon}n^{3+\varepsilon})$ and the average-case space complexity is $O(g^3n^3)$, whereas the worst-case time and space complexities are $O(g^{5+\varepsilon}n^{3+\varepsilon})$ and $O(g^4n^3)$, respectively. Note that for the curves treated in [24], Lauder and Wan obtained a worst-case time complexity of $O(g^{6+\varepsilon}n^{3+\varepsilon})$. An implementation in the C programming language shows that cryptographical sizes are now feasible for any genus g. This paper is the theoretical version of [40]: it provides a detailed description of the underlying mathematics, presents all missing proofs and corrects the complexity analysis.

The remainder of the paper is organised as follows: Section 2 reviews the basics of Monsky–Washnitzer cohomology and Section 3 shows how to extend Kedlaya's algorithm to characteristic 2. Section 4 contains a ready to implement description of the resulting algorithm and a detailed complexity analysis. Finally, Section 5 presents running times and memory usages of an implementation in the C programming language.

2. Monsky–Washnitzer Cohomology

In this section we briefly recall the definition of Monsky–Washnitzer cohomology as introduced by Monsky and Washnitzer [34], [31], [32]; more details can be found in the lectures by Monsky [33] and the survey by van der Put [39].

Let $\overline{X}$ be a smooth affine variety over a finite field $k := \mathbb{F}_q$ with $q = p^n$ elements. Denote the coordinate ring of $\overline{X}$ by $\overline{A}$. Let R be the ring of Witt vectors of $\mathbb{F}_q$, i.e. the degree n unramified extension of the p-adic integers $\mathbb{Z}_p$ with residue field $\mathbb{F}_q$ and let K be the fraction field of R. Elkik [10] showed that there always exists a smooth finitely generated R-algebra A such that $A \otimes_R \mathbb{F}_q \cong \overline{A}$. In general A does not allow a lift of the Frobenius endomorphism $\overline{F}$ on $\overline{A}$; Monsky and Washnitzer solve this problem by constructing a subalgebra $A^\dagger$ of the p-adic completion of A, whose elements satisfy growth conditions. The *dagger ring* or *weak completion* $A^\dagger$ is defined as follows: write $A := R[x_1, \ldots, x_n]/(f_1, \ldots, f_m)$, then

$$A^\dagger := R\langle x_1, \ldots, x_n\rangle^\dagger/(f_1, \ldots, f_m),$$

where $R\langle x_1, \ldots, x_n\rangle^\dagger$ consists of power series

$$\left\{\sum a_\alpha x^\alpha \in R[[x_1, \ldots, x_n]] \mid \exists C, \rho \in \mathbb{R}, C > 0, 0 < \rho < 1, \forall \alpha : |a_\alpha| \leq C\rho^{|\alpha|}\right\},$$

with $\alpha := (\alpha_1, \ldots, \alpha_n)$, $x^\alpha := x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ and $|\alpha| := \sum_{i=0}^n \alpha_i$.

Let $\overline{B}/k$ and B/R be smooth and finitely generated with $B \otimes_R \mathbb{F}_q \cong \overline{B}$ and let $B^\dagger$ be the dagger ring of B. Given a morphism of k-algebra's $\overline{G} : \overline{A} \to \overline{B}$, there always exists an R-morphism $G : A^\dagger \to B^\dagger$ lifting $\overline{G}$. This last property implies that we can lift the q-power Frobenius from $\overline{A}$ to $A^\dagger$.

For $A^\dagger$ we can define the universal module $D^1(A^\dagger)$ of differentials

$$D^1(A^\dagger) := (A^\dagger\, dx_1 + \cdots + A^\dagger\, dx_n)/\left(\sum_{i=1}^m A^\dagger \left(\frac{\partial f_i}{\partial x_1}\, dx_1 + \cdots + \frac{\partial f_i}{\partial x_n}\, dx_n\right)\right).$$

Let $D^i(A^\dagger) := \bigwedge^i D^1(A^\dagger)$ be the ith exterior product of $D^1(A^\dagger)$ and denote with $d_i : D^i(A^\dagger) \to D^{i+1}(A^\dagger)$ the exterior differentiation. Since $d_{i+1} \circ d_i = 0$ we get the de Rham complex $D(A^\dagger)$

$$0 \longrightarrow D^0(A^\dagger) \xrightarrow{d_0} D^1(A^\dagger) \xrightarrow{d_1} D^2(A^\dagger) \xrightarrow{d_2} D^3(A^\dagger) \cdots.$$

The ith cohomology group of $D(A^\dagger)$ is defined as $H^i(\overline{A}/R) := \operatorname{Ker} d_i/\operatorname{Im} d_{i-1}$ and $H^i(\overline{A}/K) := H^i(\overline{A}/R) \otimes_R K$ finally defines the ith Monsky–Washnitzer cohomology group. One can prove that for smooth, finitely generated k-algebra's $\overline{A}$

the map $\overline{A} \mapsto H^\bullet(\overline{A}/K)$ is well defined and functorial, which justifies the notation. Replacing $A^\dagger$ with A in the above construction of the ith Monsky–Washnitzer cohomology group $H^i(\overline{A}/K)$ gives rise to the ith algebraic de Rham cohomology group $H^i_{\mathrm{DR}}(A/K)$. Unlike the Monsky–Washnitzer cohomology, the algebraic de Rham cohomology essentially depends on the algebra A and in general $H^i(\overline{A}/K)$ will not be isomorphic to $H^i_{\mathrm{DR}}(A/K)$.

Let F be a lift of the q-power Frobenius endomorphism of $\overline{A}$ to $A^\dagger$, then F induces an endomorphism F_* on the cohomology groups $H^i(\overline{A}/K)$. The main theorem of Monsky–Washnitzer cohomology is that these groups satisfy a Lefschetz fixed point formula.

Theorem 1 (Lefschetz Fixed Point Formula). *Let $\overline{X}/\mathbb{F}_q$ be a smooth affine variety of dimension d, then the number of $\mathbb{F}_q$-rational points on $\overline{X}$ equals*

$$\sum_{i=0}^{d} (-1)^i \operatorname{Tr}(q^d F_*^{-1} | H^i(\overline{A}/K)).$$

3. Cohomology of Hyperelliptic Curves

3.1. *Overview of Kedlaya's Construction*

Let $\mathbb{F}_q$ be a finite field with $q = p^n$ elements and fix an algebraic closure $\overline{\mathbb{F}}_q$. Throughout this section we assume that p is a small odd prime. Let $\overline{Q}(x)$ be a monic polynomial of degree $2g+1$ over $\mathbb{F}_q$ without repeated roots and let $\overline{C}$ be the affine hyperelliptic curve defined by the equation $y^2 = \overline{Q}(x)$. Kedlaya does not work with the curve $\overline{C}$ itself, but with the affine curve $\overline{C}'$ which is obtained from $\overline{C}$ by removing the locus of $y = 0$, i.e. the points $(\overline{\xi}_i, 0) \in \overline{\mathbb{F}}_q \times \overline{\mathbb{F}}_q$ where $\overline{\xi}_i$ is a zero of $\overline{Q}(x)$. The coordinate ring $\overline{A}$ of $\overline{C}'$ is clearly given by $\mathbb{F}_q[x, y, y^{-1}]/(y^2 - \overline{Q}(x))$.

Let K be a degree n unramified extension of $\mathbb{Q}_p$, with valuation ring R, such that $R/pR = \mathbb{F}_q$. Take any monic lift $Q(x) \in R[x]$ of $\overline{Q}(x)$ and let C be the smooth affine hyperelliptic curve defined by $y^2 = Q(x)$. Let C' be the curve obtained from C by removing the locus of $y = 0$. Then the coordinate ring of C' is $A = R[x, y, y^{-1}]/(y^2 - Q(x))$. Let $A^\dagger$ denote the weak completion of A. Since $\overline{F} = \overline{\sigma}^n$, with $\overline{\sigma}$ the p-power Frobenius, it is sufficient to lift $\overline{\sigma}$ to an endomorphism σ of $A^\dagger$. It is natural to define σ as the Frobenius substitution on R and to extend it to $A^\dagger$ by mapping x to $x^\sigma := x^p$ and y to y^σ with

$$y^\sigma := y^p \left(1 + \frac{Q(x)^\sigma - Q(x)^p}{Q(x)^p}\right)^{1/2} = y^p \sum_{i=0}^{\infty} \binom{\frac{1}{2}}{i} \frac{(Q(x)^\sigma - Q(x)^p)^i}{y^{2pi}}.$$

An easy calculation shows that $\operatorname{ord}_p\binom{1/2}{i} \geq 0$ which implies that y^σ is an element of $A^\dagger$ since p divides $Q(x)^\sigma - Q(x)^p$. Note that it is essential that y^{-1} is an element of $A^\dagger$, which explains why we compute with C' instead of C.

Since C' has dimension one, the only non-trivial Monsky–Washnitzer cohomology groups are $H^0(\overline{A}/K)$ and $H^1(\overline{A}/K)$. Finding a basis for $H^0(\overline{A}/K)$ is easy since by definition $H^0(\overline{A}/K) := \operatorname{Ker} d_0$, with d_0 the derivation from $A^\dagger$ into $D^1 A^\dagger$, which implies that $H^0(\overline{A}/K)$ is a one-dimensional K-vector space. The case $H^1(\overline{A}/K)$ is more difficult

and proceeds in two steps. Kedlaya first constructs a basis for the algebraic de Rham cohomology of A and devises reduction formulae to express any differential form on this basis. Then he proves that these formulae lead to a convergent process when applied to the de Rham cohomology of $A^\dagger$, i.e. $H^1(\overline{A}/K)$ and concludes that the basis for the algebraic de Rham cohomology also is a basis for $H^1(\overline{A}/K)$.

The de Rham cohomology of A splits into eigenspaces under the hyperelliptic involution: a positive eigenspace generated by $x^i/y^2\,dx$ for $i = 0, \ldots, 2g$ and a negative eigenspace generated by $x^i/y\,dx$ for $i = 0, \ldots, 2g-1$. Using the equation of the curve, any differential form can be written as $\sum_{k=-B_U}^{B_L} \sum_{i=0}^{2g} a_{i,k} x^i/y^k\,dx$ with $a_{i,k} \in K$ and $B_U, B_L \in \mathbb{N}$. A differential of the form $P(x)/y^s\,dx$ with $P(x) \in K[x]$ and $s \in \mathbb{N}$ can be reduced as follows. Since $Q(x)$ has no repeated roots, we can always write the polynomial $P(x) = U(x)Q(x) + V(x)Q'(x)$. Using the fact that $d(V(x)/y^{s-2})$ is exact, one obtains

$$\frac{P(x)}{y^s}\,dx \equiv \left(U(x) + \frac{2V'(x)}{(s-2)}\right)\frac{dx}{y^{s-2}},$$

where $\equiv$ means equality modulo exact differentials. This congruence can be used to reduce a differential form involving negative powers of y to the case $s = 1$ and $s = 2$. A differential $P(x)/y\,dx$ with $\deg P = m \geq 2g$ can be reduced by repeatedly subtracting suitable multiples of the exact differential $d(x^{i-2g}y)$ for $i = m, \ldots, 2g$. Finally, it is clear that the differential $P(x)/y^2\,dx$ is congruent to $(P(x) \bmod Q(x))/y^2\,dx$ modulo exact differentials. A differential of the form $P(x)y^s\,dx$ with $P(x) \in K[x]$ and $s \in \mathbb{N}$ is exact if s is even and equal to $P(x)Q(x)^{\lceil s/2 \rceil}/y\,dx$ if s is odd and thus can be reduced using the above reduction formula.

Kedlaya then proves two lemmata which bound the denominators introduced during the above reduction process. The result is as follows: let $A(x) \in R[x]$ be a polynomial of degree $\leq 2g$, then for $k \in \mathbb{N}$ the reduction of $A(x)y^{2k+1}\,dx$ becomes integral upon multiplication by $p^{\lfloor \log_p((2g+1)(k+1)-2) \rfloor}$ and the reduction of $A(x)/y^{2k+1}\,dx$ becomes integral upon multiplication by $p^{\lfloor \log_p(2k+1) \rfloor}$. This implies that the reduction process converges for elements of $D^1(A^\dagger)$.

The final step in the algorithm consists of computing the action induced by σ on a basis of $H^1(\overline{A}/K)$. Using the Lefschetz fixed point theorem, Kedlaya shows that it is sufficient to compute the matrix M through which σ acts on the anti-invariant part $H^1(\overline{A}/K)^-$ of $H^1(\overline{A}/K)$. Therefore we only need to compute $(x^i/y\,dx)^\sigma = px^{p(i+1)-1}/y^\sigma\,dx$ for $i = 0, \ldots, 2g-1$. Using the aforementioned reduction process we express $(x^i/y\,dx)^\sigma$ on the basis of $H^1(\overline{A}/K)^-$ and compute the matrix M. The characteristic polynomial of Frobenius can then be recovered from the coefficients of the characteristic polynomial of the matrix $MM^\sigma \cdots M^{\sigma^{n-1}}$ through which the Frobenius $F = \sigma^n$ acts on $H^1(\overline{A}/K)^-$.

3.2. *Cohomology of Hyperelliptic Curves over* $\mathbb{F}_{2^n}$

Let $\mathbb{F}_q$ be a finite field with $q = 2^n$ elements and fix an algebraic closure $\overline{\mathbb{F}}_q$. Consider the smooth affine hyperelliptic curve $\overline{C}$ of genus g defined by the equation

$$\overline{C} : y^2 + \overline{h}(x)y = \overline{f}(x),$$

with $\overline{h}(x), \overline{f}(x) \in \mathbb{F}_q[x]$, $\overline{f}(x)$ monic of degree $2g+1$ and $\deg \overline{h} \leq g$. Write $\overline{h}(x)$ as $\overline{c} \cdot \prod_{i=0}^{s}(x - \overline{\theta}_i)^{m_i}$ with $\overline{\theta}_i \in \overline{\mathbb{F}}_q$, $\overline{c} \in \mathbb{F}_q \backslash \{0\}$ the leading coefficient of $\overline{h}(x)$ and define

$\overline{H}(x) = \prod_{i=0}^{s}(x - \overline{\theta}_i) \in \mathbb{F}_q[x]$. If $\overline{h}(x)$ is a constant, we set $\overline{H}(x) = 1$. Without loss of generality we can assume that $\overline{H}(x) \mid \overline{f}(x)$. Indeed, the isomorphism defined by $x \mapsto x$ and $y \mapsto y + \sum_{i=0}^{s} \overline{b}_i x^i$ transforms the curve in

$$y^2 + \overline{h}(x)y = \overline{f}(x) - \sum_{i=0}^{s} \overline{b}_i^2 x^{2i} - \overline{h}(x) \sum_{i=0}^{s} \overline{b}_i x^i.$$

The polynomial $\overline{H}(x)$ will divide the right-hand side of the above equation if and only if $\overline{f}(\overline{\theta}_j) = \sum_{i=0}^{s} \overline{b}_i^2 \cdot \overline{\theta}_j^{2i}$ for $j = 0, \ldots, s$. This is a system of linear equations in the indeterminates $\overline{b}_i^2$ and its determinant is a Vandermonde determinant. Since the $\overline{\theta}_j$ are the zeros of a polynomial defined over $\mathbb{F}_q$, the system of equations is invariant under the q-power Frobenius automorphism $\overline{F}$ and it follows that the $\overline{b}_i^2$ and therefore the $\overline{b}_i$ are elements of $\mathbb{F}_q$. We conclude that we can always assume that $\overline{H}(x) \mid \overline{f}(x)$.

Let $\overline{\pi} : \overline{C}(\overline{\mathbb{F}}_q) \to \mathbb{A}^1(\overline{\mathbb{F}}_q)$ be the projection on the x-axis. It is clear that $\overline{\pi}$ ramifies at the points $(\overline{\theta}_i, 0) \in \overline{\mathbb{F}}_q \times \overline{\mathbb{F}}_q$ for $i = 0, \ldots, s$ where $\overline{H}(\overline{\theta}_i) = 0$. Note that the ordinate of these points is zero, since we assumed that $\overline{H}(x) \mid \overline{f}(x)$. Let $\overline{C}'$ be the curve obtained from $\overline{C}$ by removing the ramification points $(\overline{\theta}_i, 0)$ for $i = 0, \ldots, s$. Then the coordinate ring $\overline{A}$ of $\overline{C}'$ is

$$\mathbb{F}_q[x, y, \overline{H}(x)^{-1}]/(y^2 + \overline{h}(x)y - \overline{f}(x)).$$

Let K be a degree n unramified extension of $\mathbb{Q}_2$ with valuation ring R and residue field $R/2R = \mathbb{F}_q$. Write $\overline{h}(x) = \overline{c} \cdot \prod_{i=0}^{r} \overline{P}_i(x)^{t_i}$, where the $\overline{P}_i(x)$ are monic and irreducible over $\mathbb{F}_q$. Let $D = \max_i t_i$, then $\overline{h}(x)$ divides $\overline{H}(x)^D$, since we have the identity $\overline{H}(x) = \prod_{i=0}^{r} \overline{P}_i(x)$. Lift $\overline{P}_i(x)$ for $i = 0, \ldots, r$ to any monic polynomial $P_i(x) \in R[x]$. Define $H(x) = \prod_{i=0}^{r} P_i(x)$ and $h(x) = c \cdot \prod_{i=0}^{r} P_i(x)^{t_i}$, with c any lift of $\overline{c}$ to R. Since $\overline{H}(x)$ divides $\overline{f}(x)$ we can define $\overline{Q}_{\overline{f}}(x) = \overline{f}(x)/\overline{H}(x)$. Let $Q_f(x) \in R[x]$ be any monic lift of $\overline{Q}_{\overline{f}}(x)$ and finally set $f(x) = H(x)Q_f(x)$. The result is that we have now constructed a lift C of the curve $\overline{C}$ to R defined by the equation

$$C : y^2 + h(x)y = f(x).$$

Note that due to the careful construction of C we have the following properties: $H(x) | h(x)$, $H(x) | f(x)$ and $h(x) | H(x)^D$. Let K^{ur} be the maximal unramified extension of K with valuation ring R^{ur}. For $k = 0, \ldots, s$, let θ_k be the zeros of $H(x)$ and note that these are units in R^{ur}. Furthermore, let $\pi : C(\overline{K}) \to \mathbb{A}^1(\overline{K})$ be the projection on the x-axis, then the $(\theta_k, 0)$ are ramification points of π.

Consider the curve C' obtained from C by deleting the ramification points $(\theta_k, 0)$ for $k = 0, \ldots, s$, then the coordinate ring A of C' is

$$R[x, y, H(x)^{-1}]/(y^2 + h(x)y - f(x))$$

and there exists an involution ι on A which sends x to x and y to $-y - h(x)$. Let $A^\dagger$ denote the weak completion of A. Using the equation of the curve, we can represent any element of $A^\dagger$ as a series $\sum_{i=-\infty}^{+\infty}(U_i(x) + V_i(x)y)S(x)^i$, with the degree of $U_i(x)$ and

$V_i(x)$ smaller than the degree of $S(x)$, where $S(x) = H(x)$ if $\deg H > 0$ and $S(x) = x$ if $H(x) = 1$. The growth condition on the dagger ring implies that there exist real numbers δ and $\epsilon > 0$ such that $\mathrm{ord}_2(U_i(x)) \geq \epsilon \cdot |i| + \delta$ and $\mathrm{ord}_2(V_i(x)) \geq \epsilon \cdot |i+1| + \delta$, where $\mathrm{ord}_2(P(x))$ is defined as $\min_j \mathrm{ord}_2(p_j)$ for $P(x) = \sum p_j x^j \in K[x]$.

Lift the 2-power Frobenius $\overline{\sigma}$ on $\mathbb{F}_q$ to the Frobenius substitution σ on R. We extend σ to an endomorphism of $A^\dagger$ by mapping x to x^2 and y to y^σ, with

$$(y^\sigma)^2 + h(x)^\sigma\, y^\sigma - f(x)^\sigma = 0 \quad \text{and} \quad y^\sigma \equiv y^2 \bmod 2.$$

Using Newton iteration we can compute the solution to the above equations as an element of the 2-adic completion A^∞ as

$$W_{k+1} \equiv W_k - \frac{W_k^2 + h(x)^\sigma\, W_k - f(x)^\sigma}{2W_k + h(x)^\sigma} \bmod 2^{k+1}. \tag{1}$$

The only remaining difficulty in the above Newton iteration is that we have to invert $2W_k + h(x)^\sigma$ in the ring A^∞. Since $h(x) \mid H(x)^D$, it makes sense to define $Q_H(x) := H(x)^D/h(x)$ and we clearly have $1/h(x) = Q_H(x)/H(x)^D$. We can now compute the inverse of $2W_k + h(x)^\sigma$ as

$$\frac{Q_H(x)^2}{H(x)^{2D} \cdot \left(1 + (Q_H(x)^2(2W_k + h(x)^\sigma - h(x)^2))/H(x)^{2D}\right)}. \tag{2}$$

Note that $h(x)^\sigma \equiv h(x)^2 \bmod 2$, which implies that the denominator in the above formula is invertible in A^∞. Contrary to the odd characteristic case it is not immediately clear that the solution $W := \lim_{k\to+\infty} W_k$ is an element of $A^\dagger$. A theorem by Bosch [5] guarantees the existence of such a solution, but does not provide bounds on the rate of convergence. Since these bounds are needed in the complexity analysis, we prove the following lemma.

Lemma 1. *For $k \geq 1$, let $W_k = \sum_{i=-L_k}^{A_k} U_i(x)S(x)^i + \sum_{i=-L_k}^{B_k} V_i(x)S(x)^i y \in A$, with $S(x) = H(x)$ if $\deg H > 0$, $S(x) = x$ if $H(x) = 1$ and $\deg U_i < \deg S$, $\deg V_i < \deg S$ satisfy*

$$W_k^2 + h(x)^\sigma\, W_k - f(x)^\sigma \equiv 0 \bmod 2^k \quad \text{and} \quad W_k \equiv y^2 \bmod 2$$

with $U_{A_k} \neq 0$, $V_{B_k} \neq 0$, $U_{-L_k} \neq 0$ or $V_{-L_k} \neq 0$ and such that $U_i = 0$ or $\mathrm{ord}_2(U_i(x)) < k$ for $-L_k \leq i \leq A_k$ and $V_i = 0$ or $\mathrm{ord}_2(V_i(x)) < k$ for $-L_k \leq i \leq B_k$. Then A_k, B_k and L_k can be bounded for $k \geq 2$ as

$$\begin{aligned} A_k &\leq 2(k-1)(d_S^f - 2d_S^h) + 2d_S^h, \\ B_k &\leq 2(k-2)(d_S^f - 2d_S^h) + (d_S^f - d_S^h), \\ L_k &\leq 4(k-1)D - 2D, \end{aligned} \tag{3}$$

with $d_S^f := \deg f/\deg S$ and $d_S^h := \deg h/\deg S$.

Proof. An easy calculation shows that $W_1 \equiv f(x) - h(x)y \bmod 2$, thus $A_1 \leq d_S^f$, $B_1 \leq d_S^h$, $L_1 \leq 0$ and that

$$W_2 \equiv \frac{f(x)^\sigma - f(x)^2 - h(x)^\sigma f(x)}{h(x)^2} + y\frac{h(x)^\sigma + 2f(x)}{h(x)} \bmod 4,$$

which implies that W_2 satisfies the lemma. The Newton iteration (1) can be rewritten as

$$h(x)^2\, W_{k+1} \equiv -W_k^2 + (h(x)^2 - h(x)^\sigma)\, W_k + f(x)^\sigma \bmod 2^{k+1}.$$

Let $\alpha_k(x) := \sum_{i=-L_k}^{A_k} U_i(x)S(x)^i$ and $\beta_k(x) := \sum_{l=-L_k}^{B_k} V_i(x)S(x)^i$ such that $W_k = \alpha_k(x) + \beta_k(x)y$. Note that $W_k \equiv W_{k-1} \bmod 2^{k-1}$, so we can define

$$\Delta_{\alpha,k}(x) := \frac{\alpha_k(x) - \alpha_{k-1}(x)}{2^{k-1}} \quad \text{and} \quad \Delta_{\beta,k}(x) := \frac{\beta_k(x) - \beta_{k-1}(x)}{2^{k-1}},$$

for $k \geq 1$ and $\Delta_{\alpha,0}(x) := \Delta_{\beta,0}(x) := 0$. It is clear that W_k can be written as

$$W_k = \Delta_{\alpha,1} + 2\Delta_{\alpha,2} + \cdots + 2^{k-1}\Delta_{\alpha,k} + y\left(\Delta_{\beta,1} + 2\Delta_{\beta,2} + \cdots + 2^{k-1}\Delta_{\beta,k}\right).$$

Plugging this into the Newton iteration gives the following equation

$$\begin{aligned} h(x)^2\, W_{k+1} \\ \equiv & - \sum_{\substack{1\leq i<j \\ i+j-1<k+1}} 2^{i+j-1}\left(\Delta_{\alpha,i}\Delta_{\alpha,j} + (f(x) - h(x)y)\,\Delta_{\beta,i}\Delta_{\beta,j}\right) \\ & - y \sum_{i+j-1<k+1} 2^{i+j-1}\Delta_{\alpha,i}\Delta_{\beta,j} \\ & - \sum_{2(i-1)<k+1} 2^{2(i-1)}\left(\Delta_{\alpha,i}^2 + (f(x) - h(x)y)\Delta_{\beta,i}^2\right) \\ & + (h(x)^2 - h(x)^\sigma) \sum_{i<k+1} 2^{i-1}(\Delta_{\alpha,i} + \Delta_{\beta,i}y) + f(x)^\sigma \bmod 2^{k+1}. \end{aligned}$$

By definition $Q_H(x)h(x) = H(x)^D$, which implies $1/h(x)^2 = Q_H(x)^2/H(x)^{2D}$ and $\deg Q_H = D\deg H - \deg h$. Since $\deg \Delta_{\alpha,i} \leq A_i$ and $\deg \Delta_{\beta,i} \leq B_i$, we conclude that A_{k+1} is less than or equal to

$$\begin{aligned} \max\Bigg(& \max_{i+j<k+2}(A_i + A_j,\, B_i + B_j + d_S^f),\ \max_{2i<k+3}(2A_i,\, 2B_i + d_S^f), \\ & \max_{i<k+1} A_i + 2d_S^h,\, 2d_S^f\Bigg) - 2d_S^h. \end{aligned}$$

Using the bounds given in (3) for A_i and B_i and the bounds $A_1 \leq d_S^f$, $B_1 \leq d_S^h$ and $L_1 \leq 0$, we see that A_{k+1} also satisfies the bounds (3). Similar reasoning can be used to prove that B_{k+1} and L_{k+1} also satisfy the given bounds. □

Lemma 1 implies that the q-power Frobenius $\overline{F}$ can be lifted to an endomorphism F on the dagger ring $A^\dagger$, since we can simply take $F := \sigma^n$. If we are to compute the action of F on the first Monsky–Washnitzer cohomology group $H^1(\overline{A}/K)$, we need to determine a basis for $H^1(\overline{A}/K)$. Following Kedlaya, we proceed in two steps: we first determine a basis for the algebraic de Rham cohomology group $H^1_{\mathrm{DR}}(A/K)$ and then show that this is also a basis for $H^1(\overline{A}/K)$.

Analogous to the odd characteristic case, the algebraic de Rham cohomology $H^1_{\mathrm{DR}}(A/K)$ of A splits into eigenspaces under the hyperelliptic involution. The positive eigenspace $H^1_{\mathrm{DR}}(A/K)^+$ is generated by $x^i/H(x)\,dx$ for $i = 0, \ldots, s$ and the negative eigenspace $H^1_{\mathrm{DR}}(A/K)^-$ is generated by $x^i y\,dx$ for $i = 0, \ldots, 2g-1$. Note that the positive eigenspace corresponds to the deleted ramification points $(\theta_k, 0)$ for $k = 0, \ldots, s$. Every element of $H^1_{\mathrm{DR}}(A/K)$ can be written as a linear combination of differentials of the form $x^k H(x)^m y^l\,dx$, $x^k H(x)^m y^l\,dy$ with $k, l \in \mathbb{N}$ and $m \in \mathbb{Z}$. Using the equation of the curve, we can reduce to the case $l = 0$ or 1. Since $d(x^k H(x)^m y)$ and $d(x^k H(x)^m y^2)$ are exact, we conclude that $H^1_{\mathrm{DR}}(A/K)$ is generated by differentials of the form $x^k H(x)^m\,dx$ and $x^k H(x)^m y\,dx$ with $k \in \mathbb{N}$ and $m \in \mathbb{Z}$.

It is clear that $x^k H(x)^m\,dx$ is exact for $k \in \mathbb{N}$ and $m \geq 0$. If $\deg H > 0$ and $m < 0$ we can assume that $0 \leq k < \deg H$ and since $H(x)$ is square-free we can write x^k as $A(x)H(x) + B(x)H'(x)$, which leads to

$$x^k H(x)^m\,dx = A(x)H(x)^{m+1}\,dx + B(x)H'(x)H(x)^m\,dx.$$

Since $d(B(x)H(x)^{m+1})$ is exact we can reduce the above differential further for $m < -1$ by using the relation

$$B(x)H'(x)H(x)^m\,dx \equiv -\frac{B'(x)H(x)^{m+1}}{m+1}\,dx,$$

where $\equiv$ means equality modulo exact differentials. As a result we can now reduce any form $x^k H(x)^m\,dx$ to a linear combination of the differentials $x^i/H(x)\,dx$ for $i = 0, \ldots, s$.

For $m > 0$ we can reduce the differential form $x^k H(x)^m y\,dx$ for $k \in \mathbb{N}$ if we know how to reduce the form $x^i y\,dx$ for $i \in \mathbb{N}$. Rewriting the equation of the curve as $(2y+h(x))^2 = 4f(x)+h(x)^2$ and differentiating both sides leads to $(2y+h(x))\,d(2y+h(x)) = (2f'(x) + h(x)h'(x))\,dx$. Furthermore, for all $j \geq 1$, we have the following relations:

$$\begin{aligned} x^j(2f'(x) + h(x)h'(x))(2y + h(x))\,dx &= x^j(2y + h(x))^2\,d(2y + h(x)) \\ &\equiv -\tfrac{1}{3}(2y + h(x))^3\,dx^j \\ &= -\frac{j}{3}x^{j-1}(4f(x) + h(x)^2)(2y + h(x))\,dx. \end{aligned}$$

Since $P(x)h(x)\,dx$ is exact for any polynomial $P(x) \in K[x]$, we finally obtain that

$$\left[x^j(2f'(x) + h(x)h'(x)) + \frac{j}{3}x^{j-1}(4f(x) + h(x)^2)\right] y\,dx \equiv 0.$$

The polynomial between brackets has degree $2g + j$ and its leading coefficient is $2(2g + 1) + 4j/3 \neq 0$. Note that the formula is also valid for $j = 0$. This means that we can reduce $x^i y\,dx$ for any $i \geq 2g$ by subtracting a suitable multiple of the above differential for $j = i - 2g$.

For $m < 0$ we need an extra trick to reduce the form $x^k H(x)^m y\,dx$ with $k \in \mathbb{N}$. Recall that $Q_f(x) = f(x)/H(x)$ and since the curve is non-singular, we conclude

that $\gcd(Q_f(x), H(x)) = 1$. Furthermore, $H(x)$ has no repeated roots which implies $\gcd(H(x), Q_f(x)H'(x)) = 1$. Let $i = -m > 0$, then we can partially reduce $x^k y/H(x)^i\,dx$ by writing x^k as $A(x)H(x) + B(x)Q_f(x)H'(x)$, which leads to

$$\frac{x^k}{H(x)^i}\,y\,dx = \frac{A(x)}{H(x)^{i-1}}\,y\,dx + \frac{B(x)Q_f(x)H'(x)}{H(x)^i}\,y\,dx.$$

The latter differential form can be reduced using the following congruence:

$$\begin{aligned}\frac{B(x)}{H(x)^i}&(2f'(x) + h(x)h'(x))(2y + h(x))\,dx\\ &= \frac{B(x)}{H(x)^i}(2y + h(x))^2\,d(2y + h(x))\\ &\equiv -\tfrac{1}{3}(2y + h(x))^3 d\left(\frac{B(x)}{H(x)^i}\right).\end{aligned}$$

Substituting the expressions $h(x) = Q_h(x)H(x)$, $f(x) = Q_f(x)H(x)$ and $(2y + h(x))^2 = 4f(x) + h(x)^2$, we get

$$\begin{aligned}&\frac{B(x)Q_f(x)H'(x)}{H(x)^i}y\,dx\\ &\quad\equiv \frac{B(iH'Q_h^2 - 6Q_f' - 3Q_hh') - B'(4Q_f + Q_hh)}{(6-4i)H^{i-1}}\,y\,dx + \frac{I}{H}\,dx,\end{aligned}$$

where $I(x)/H(x)\,dx$ is a suitable invariant differential. As a result we can write any form $x^kH(x)^m y\,dx$ for $k \in \mathbb{N}$ and $m \in \mathbb{Z}$ as a linear combination of the differentials $x^i y\,dx$ for $i = 0, \ldots, 2g-1$ and $x^i/H(x)\,dx$ for $i = 0, \ldots, s$.

To show that the Monsky–Washnitzer cohomology $H^1(\overline{A}/K)$ is generated by the same differential forms as the algebraic de Rham cohomology, we need to bound the denominators introduced during the reduction process.

Lemma 2. *Let $A := R[x, y]/(y^2 + h(x)y - f(x))$ and suppose that*

$$x^r y\,dx = \sum_{i=0}^{2g-1} a_i x^i y\,dx + dS, \tag{4}$$

with $r \in \mathbb{N}$, $a_i \in K$ and $S \in A \otimes K$. Then $2^m a_i \in R$, $2^{m'}S - \beta \in A$, where $m = 3 + \lfloor \log_2(r + g + 1) \rfloor$, $m' = 1 + m + \lfloor \log_2(2g + \deg h) \rfloor$ and β is some suitable element in K.

Proof. The proof has two distinct parts. The first part is similar to Kedlaya's argument in Lemma 3 of [19], and is based on a local analysis around the point at infinity of the curve C. Put $t = x^g/y$, then one easily verifies that

$$x = t^{-2}\left(1 + \sum_{j=1}^{\infty}\alpha_j t^j\right) \quad \text{and} \quad y = t^{-2g-1}\left(1 + \sum_{j=1}^{\infty}\beta_j t^j\right), \tag{5}$$

with $\alpha_j, \beta_j \in R$. To see this, put $z = 1/x$, rewrite the equation of the curve C as $z + tz^{g+1}h(1/z) - t^2z^{2g+1}f(1/z) = 0$ and write z as a power series in t using Newton iteration. The relation (4) can be rewritten as

$$2^{m-1}x^r(2y + h(x))\,dx = \sum_{i=0}^{2g-1} 2^{m-1}a_i x^i(2y + h(x))\,dx + dT,$$

with $T \in A \otimes K$. Considering the involution ι of A which sends x to x and $2y + h(x)$ to $-(2y + h(x))$, we see that we can write $T = \sum_{i=0}^{N} A_i x^i(2y + h(x))$, with N big enough and $A_i \in K$. This yields

$$\begin{aligned} &2^{m-1}x^r(2y + h(x))\,dx - \sum_{i=0}^{2g-1} 2^{m-1}a_i x^i(2y + h(x))\,dx \\ &\qquad = d\left(\sum_{i=0}^{N} A_i x^i(2y + h(x))\right). \end{aligned} \tag{6}$$

In the above equation we express x and y in terms of t using equalities (5). Since $x^i y = t^{-2i-2g-1} + \cdots$, we get $x^i(2y + h(x))\,dx = (-4t^{-2i-2g-4} + \cdots)\,dt$, which yields

$$\begin{aligned} &2^{m-1}\sum_{j=-\max(2r+2g+4,6g+2)} \gamma_j t^j\,dt \\ &\qquad = d\left(\sum_{i=0}^{N} 2A_i(t^{-2i-2g-1} + \cdots) + A_i(c\,t^{-2i-2\deg h} + \cdots)\right), \end{aligned}$$

with $\gamma_j \in K$ for all j and $\gamma_j \in R$ when $j < -2(2g-1) - 2g - 4 = -6g - 2$ and c is the leading coefficient of $h(x)$. Note that c is a unit in R. Integrating with respect to t and dividing by 2 gives

$$\sum_{j\geq -\max(2r+2g+3,6g+1)} \gamma'_j t^j = \sum_{i=0}^{N} A_i(t^{-2i-2g-1} + \cdots) + \sum_{i=0}^{N} \frac{A_i}{2}(c\,t^{-2i-2\deg h} + \cdots), \tag{7}$$

with $\gamma'_j \in K$ for all j and $\gamma'_j \in R$ when $j < -6g - 1$. Indeed, the integration process introduces denominators which become integral after multiplication with $2^{\lfloor \log(2r+2g+2)\rfloor} = 2^{m-2}$ if $r \geq 2g-1$. A first consequence of (7) is that $A_i = 0$ for all $i > \max(r+1, 2g)$. We claim that (7) implies that $A_i \in R$ for all $i > 2g$. Suppose the claim is false. Then let i_0 be the largest integer with $i_0 > 2g$ and $A_{i_0} \notin R$. Note that $-2i_0 - 2g - 1 < -6g - 1$, since $i_0 > 2g$. Hence the monomials in the left-hand side of (7) with degree $\leq -2i_0 - 2g - 1$ have coefficients in R. Moreover, the monomials of degree $< -2i_0 - 2g - 1$, in the first sum in the right-hand side of (7) also have coefficients in R, but this is false for the monomial of degree $-2i_0 - 2g - 1$. Hence the second sum in the right-hand side of (7) contains a monomial of degree $-2i_0 - 2g - 1$ whose coefficient is not in R. That means that there is a maximal i_1 with $A_{i_1}/2 \notin R$ and $-2i_1 - 2\deg h \leq -2i_0 - 2g - 1$. Because of parity we have that $-2i_1 - 2\deg h < -2i_0 - 2g - 1$. Since c is a unit, the right-hand side of (7) contains a monomial of degree $-2i_1 - 2\deg h < -2i_0 - 2g - 1$

whose coefficient is not in R. However, this contradicts what we said about the left-hand side. This finishes the claim that $A_i \in R$ for all $i > 2g$.

We now turn to the second part of the proof. Note that $(2y + h(x))^2 = v(x)$ with $v(x) := 4f(x) + h(x)^2$. Moreover, $d(2y + h(x)) = (w(x)/(2y + h(x)))\, dx$, where $w(x) := 2f'(x) + h(x)h'(x)$. We use these formulae to deduce from (6) a relation which does not involve y. For this purpose we multiply (6) with $(2y + h(x))/dx = w(x)/d(2y + h(x))$ obtaining

$$2^{m-1}x^r v(x) - \sum_{i=0}^{2g-1} 2^{m-1}a_i x^i v(x) = \sum_{i=0}^{N} A_i i x^{i-1} v(x) + \sum_{i=0}^{N} A_i x^i w(x).$$

We rewrite this in the form

$$\left(\sum_{i=0}^{2g-1} 2^{m-1}a_i x^i\right) v(x) + \left(\sum_{i=0}^{2g} A_i i x^{i-1}\right) v(x) + \left(\sum_{i=0}^{2g} A_i x^i\right) w(x) = F(x), \qquad (8)$$

where

$$F(x) := 2^{m-1}x^r v(x) - \sum_{i=2g+1}^{N} A_i i x^{i-1} v(x) - \sum_{i=2g+1}^{N} A_i x^i w(x) \qquad (9)$$

is a polynomial over R, since $A_i \in R$ for all $i > 2g$. From (8) and (9) it follows that $\sum_{i=0}^{2g} A_i \theta_k^i$ has valuation ≥ 0 for each root θ_k of $H(x)$, because $v(\theta_k) = 0$ and $w(\theta_k) \neq 0$. To eliminate the disturbing factor 2 in the definition of $w(x)$, we consider $q(x) := h'(x)H(x)/h(x) \in R[x]$ and $u(x) := \frac{1}{2}(w(x) - q(x)v(x)/H(x)) = f'(x) - 2q(x)f(x)/H(x)$. Note that $u(x) \in R[x]$, $\deg q = \max(0, \deg H - 1)$, $\deg u = 2g$ and that the leading coefficient of $u(x)$ is a unit in R. Rewrite (8) in such a way that $w(x)$ gets replaced by $u(x)$:

$$\left(\sum_{i=0}^{2g-1} 2^{m-1}a_i x^i + \sum_{i=0}^{2g} A_i i x^{i-1} + \frac{q(x)}{H(x)} \sum_{i=0}^{2g} A_i x^i\right) v(x) + \left(\sum_{i=0}^{2g} 2A_i x^i\right) u(x) = F(x).$$

Write $q(x)\sum_{i=0}^{2g} A_i x^i = H(x)\sum_{i=0}^{2g-1} B_i x^i + \mathrm{Rem}(x)$, with $\mathrm{Rem}(x) \in K[x]$ of degree $< \deg H$. Since $\sum_{i=0}^{2g} A_i \theta_k^i$ has valuation ≥ 0 for each root θ_k of $H(x)$, the same holds for $\mathrm{Rem}(\theta_k)$. Thus $\mathrm{Rem}(x) \in R[x]$ since the discriminant of $H(x)$ is a unit in R. Hence

$$\left(\sum_{i=0}^{2g-1} \left(2^{m-1}a_i + (i+1)A_{i+1} + B_i\right) x^i\right) v(x) + \left(\sum_{i=0}^{2g} 2A_i x^i\right) u(x) = F(x) - \frac{\mathrm{Rem}(x)v(x)}{H(x)}. \qquad (10)$$

We consider (10) as a system of $4g + 1$ linear equations in the unknowns $2^{m-1}a_i + (i + 1)A_{i+1} + B_i$ for $i = 0, \ldots, 2g - 1$ and $2A_i$ for $i = 0, \ldots, 2g$. The determinant of this

system is the resultant $\mathrm{Res}(v(x), u(x))$ of $v(x)$ and $u(x)$ because $\deg v = 2g+1$ and $\deg u = 2g$. This resultant is a unit in R because the valuation of $v(\xi)$ is zero for each root ξ of $u(x)$, since the resultant of $f'(x)$ and $h(x)$ is a unit. We conclude that the solutions of the linear system are elements of R, thus $2A_i \in R$ and $2^{m-1}a_i + (i+1)A_{i+1} + B_i \in R$. From the definition of the B_i it follows that $2B_i \in R$ since $2A_i \in R$ and $\mathrm{Rem}(x) \in R[x]$. Hence $2^m a_i \in R$, which concludes the proof of Lemma 2. □

Remark. Lemma 2 remains valid when we replace $\sum_{i=0}^{2g-1}$ by $\sum_{i=\kappa}^{2g-1+\kappa}$ whenever $r \geq \kappa \in \mathbb{N}$. The proof is the same, except that we also have to show that $A_i = 0$ for all $i < \kappa$. This follows from (6) using a local analysis at a point on the curve with $x = \theta_k$. Such a local analysis is given in the proof of Lemma 3 below.

Lemma 3. *Let* $A := R[x, y, H(x)^{-1}]/(y^2 + h(x)y - f(x))$ *with* $\deg h > 0$ *and suppose that*

$$\frac{B(x)}{H(x)^r} y\, dx = \sum_{i=0}^{2g-1} a_i x^i y\, dx + \sum_{i=0}^{s} \frac{b_i x^i}{H(x)}\, dx + dS, \tag{11}$$

where $r \in \mathbb{N}$, $B(x) \in R[x]$ *of degree* $< \deg H$, $a_i, b_i \in K$ *and* $S \in A \otimes K$. *Then* $2^m a_i \in R$, $2^{m'} b_i \in R$, $2^{m'} S - \beta \in A$, *with* $m = 3 + \lfloor \log_2(r+1) \rfloor$, $m' = 1 + m + \lfloor \log_2(2g + \deg h) \rfloor$ *and* β *is some suitable element in* K.

Proof. The proof again consists of two distinct parts. The first part is similar to Kedlaya's argument in Lemma 2 of [19] and is based on a local analysis around the ramification points $(\theta_k, 0)$ for $k = 0, \ldots, s$. In the completion of the local ring of the curve at $(\theta_k, 0)$ we can write

$$x - \theta_k = \gamma_{k,2} y^2 + \sum_{j \geq 3} \gamma_{k,j} y^j,$$

with $\gamma_{k,j} \in R^{\mathrm{ur}}$ and $\gamma_{k,2}$ a unit in R^{ur}. Indeed, to see this write $h(x)$ and $f(x)$ as a Taylor expansion around θ_k and use the equation of the curve and the condition $f'(\theta_k) \not\equiv 0 \bmod 2$, to express $x - \theta_k$ as a power series in y using Newton iteration.

Applying the involution ι to (11), we see that this relation implies

$$\begin{aligned} 2^{m-1} B(x) H(x)^{-r} (2y + h(x))\, dx - \sum_{i=0}^{2g-1} 2^{m-1} a_i x^i (2y + h(x))\, dx \\ = d\left(\sum_{i=-N}^{M} B_i(x) H(x)^i (2y + h(x)) \right), \end{aligned} \tag{12}$$

with N and M large enough integers. Expressing $x - \theta_k$ in terms of y, we get $B_i(x)H(x)^i = u_{k,i} B_i(\theta_k) y^{2i} + \cdots$ with $u_{k,i}$ a unit in R^{ur}. Substituting this in (12) and dividing by 2 we obtain

$$\begin{aligned} 2^{m-2} \sum_{j \geq -2r+2} \gamma'_{k,j} y^j\, dy \\ = d\left(\sum_{i=-N}^{M} u_{k,i} B_i(\theta_k) y^{2i+1} + \frac{u_{k,i} B_i(\theta_k)}{2} \frac{\gamma_{k,2}^{m_k} h^{(m_k)}(\theta_k)}{m_k!} y^{2i+2m_k} + \cdots \right) \end{aligned}$$

with $\gamma'_{k,j} \in K^{\mathrm{ur}}$ for all j and $\gamma'_{k,j} \in R^{\mathrm{ur}}$ when $j \leq 1$. Integrating the left-hand side of this equation with respect to y yields a series whose terms of degree ≤ 2 have coefficients in R^{ur}. The leading term of the right-hand side is $u_{k,-N} B_{-N}(\theta_k) y^{-2N+1}$, which implies that $B_{-N}(\theta_k)$ is integral for $k = 0, \ldots, s$. Since the discriminant of $H(x)$ is a unit in R we conclude that $B_{-N}(x)$ has integral coefficients. Bringing the integral terms to the left-hand side and repeating the same argument, shows that $B_i(x) \in R[x]$ for $i = -N, \ldots, 0$. This terminates the first part of the proof.

The second part of the proof proceeds along the same lines as in Lemma 2. Rewrite the sum $\sum_{i=1}^{M} B_i(x) H(x)^i (2y + h(x))$ as $\sum_{i=0}^{M'} A_i x^i (2y + h(x))$ with $M' \in \mathbb{N}$ and $A_i \in K$. Using the same formulae as in Lemma 2 we deduce from (12) a relation which does not involve y by multiplying both sides with $(2y + h(x))/dx = w(x)/d(2y + h(x))$, which leads to

$$\begin{aligned} 2^{m-1} \frac{B(x)}{H(x)^r} v(x) - \sum_{i=0}^{2g-1} 2^{m-1} a_i x^i v(x) \\ = \sum_{i=-N}^{0} B_i(x) H(x)^i w(x) + \sum_{i=0}^{M'} A_i x^i w(x) \\ + \sum_{i=-N}^{0} \left(B_i(x) i H(x)^{i-1} H'(x) + B_i'(x) H(x)^i \right) v(x) + \sum_{i=0}^{M'} A_i i x^{i-1} v(x). \end{aligned}$$

Comparing the valuation at infinity of both sides shows that $A_i = 0$ for $i > 2g$. We can therefore rewrite the above equation in the form

$$\left(\sum_{i=0}^{2g-1} 2^{m-1} a_i x^i \right) v(x) + \left(\sum_{i=0}^{2g} A_i i x^{i-1} \right) v(x) + \left(\sum_{i=0}^{2g} A_i x^i \right) w(x) = F(x), \quad (13)$$

where

$$\begin{aligned} F(x) := 2^{m-1} \frac{B(x)}{H(x)^r} v(x) - \sum_{i=-N}^{0} B_i(x) H(x)^i w(x) \\ - \sum_{i=-N}^{0} (B_i(x) i H(x)^{i-1} H'(x) + B_i'(x) H(x)^i) v(x) \end{aligned}$$

is a polynomial over R since the $B_i(x) \in R[x]$ for $i = -N, \ldots, 0$ and the left-hand side of (13) is a polynomial. From the definition of the A_i it follows that $H(x)$ divides $\sum_{i=0}^{2g} A_i x^i$. It is now easy to see that the rest of the proof is exactly the same as in the proof of Lemma 2 with $\mathrm{Rem}(x) = 0$, hence $2^m a_i \in R$ and this concludes the proof of Lemma 3. □

Remark. Lemma 3 remains valid when we replace the term $\sum_{i=0}^{2g-1} a_i x^i y \, dx$ in (11) by $\sum_{i=-\kappa}^{\Delta-\kappa} C_i(x) H(x)^i y \, dx$, with $\Delta = \lfloor (2g-1)/\deg H \rfloor$ and $C_i(x) \in K[x]$ of degree $< \deg H$ whenever $r \geq \kappa \in \mathbb{N}$. The proof is exactly the same.

Remark. If $r = 0$, then in the above proof the $B_i(x)$ are zero for all $i \leq 0$, and for $0 \leq i \leq 2g-1$ the a_i are completely determined by (13) as we saw by considering resultants. This shows that the $x^i y\,dx$ for $i = 0, \ldots, 2g-1$ and the $(x^i/H(x))\,dx$ for $i = 0, \ldots, s$ are linearly independent in $H^1_{\mathrm{DR}}(A/K)$.

An immediate consequence of Lemmata 2 and 3 is that the basis for $H^1_{\mathrm{DR}}(A/K)$ also generates $H^1(\overline{A}/K)$: reducing a differential $\sum_{k,l} a_{k,l} x^k S(x)^l y\,dx \in D^1(A^\dagger)$ with $k, l \in \mathbb{Z}$ and $0 \leq k < \deg S$ introduces denominators whose valuation grows logarithmically in $|l|$, whereas the valuation of $a_{k,l}$ grows linearly in $|l|$. Combining this with the above remark, we conclude that the basis for $H^1_{\mathrm{DR}}(A/K)$ is also a basis for $H^1(\overline{A}/K)$.

Under the action of the hyperelliptic involution, the Monsky–Washnitzer cohomology $H^1(\overline{A}/K)$ decomposes as the direct sum of the ι-invariant part $H^1(\overline{A}/K)^+$ and the ι-anti-invariant part $H^1(\overline{A}/K)^-$. Let r_k be the number of ramification points $(\overline{\theta}, 0)$ defined over $\mathbb{F}_{q^k}$, then the Lefschetz fixed point formula applied to C' gives

$$\begin{aligned}
\#C(\mathbb{F}_{q^k}) - r_k &= \#C'(\mathbb{F}_{q^k}) \\
&= \mathrm{Tr}(q^k F_*^{-k} | H^0(\overline{A}/K)) - \mathrm{Tr}(q^k F_*^{-k} | H^1(\overline{A}/K)) \\
&= q^k - \mathrm{Tr}(q^k F_*^{-k} | H^1(\overline{A}/K)^+) - \mathrm{Tr}(q^k F_*^{-k} | H^1(\overline{A}/K)^-) \\
&= q^k - r_k - \mathrm{Tr}(q^k F_*^{-k} | H^1(\overline{A}/K)^-).
\end{aligned}$$

Let $\widetilde{C}$ be the unique smooth projective curve birational to $\overline{C}$, then

$$\#\widetilde{C}(\mathbb{F}_{q^k}) = q^k + 1 - \mathrm{Tr}\left(q^k F_*^{-k} | H^1(\overline{A}/K)^-\right) = q^k + 1 - \sum_{i=1}^{2g} \alpha_i^k,$$

with α_i the eigenvalues of qF_*^{-1} on $H^1(\overline{A}/K)^-$. The Weil conjectures imply that there exist $2g$ algebraic integers $\beta_1, \ldots, \beta_{2g}$ with $\beta_i \beta_{g+i} = q$ for $i = 1, \ldots, g$ and $|\beta_i| = \sqrt{q}$ for $i = 1, \ldots, 2g$, such that for all $k > 0$ we have $\#\widetilde{C}(\mathbb{F}_{q^k}) = q^k + 1 - \sum_{i=1}^{2g} \beta_i^k$. Comparing both expressions, we see that we can relabel the α_i such that $\alpha_i = \beta_i$ for $i = 1, \ldots, 2g$. Since then $\alpha_i \alpha_{g+i} = q$, the α_i are also the eigenvalues of F_* on $H^1(\overline{A}/K)^-$. Let $\chi(t)$ be the characteristic polynomial of F_* on $H^1(\overline{A}/K)^-$, then we can finally recover the zeta function $Z(\widetilde{C}/\mathbb{F}_q; t)$ as

$$Z(\widetilde{C}/\mathbb{F}_q; t) = \frac{t^{2g}\chi(1/t)}{(1-t)(1-qt)}.$$

4. Algorithm and Complexity

Using the formulae of the previous section, we describe an algorithm to compute the characteristic polynomial of Frobenius $\chi(t)$ and the zeta function of a smooth projective hyperelliptic curve $\widetilde{C}$ of genus g over $\mathbb{F}_q$ with $q = 2^n$.

4.1. *Precision of Computation*

We have shown that $\chi(t) = t^{2g} + a_1 t^{2g-1} + \cdots + a_{2g}$ can be computed as the characteristic polynomial of F_* on $H^1(\overline{A}/K)^-$. The Weil conjectures imply that $q^{g-i} a_i = a_{2g-i}$, so it

suffices to compute $a_1, \ldots, a_g$. Furthermore, for $i = 1, \ldots, g$ the a_i can be bounded by

$$|a_i| \leq \binom{2g}{i} q^{i/2} \leq \binom{2g}{g} q^{g/2} \leq 2^{2g} q^{g/2}.$$

Therefore it suffices to compute the action of F_* on a basis of $H^1(\overline{A}/K)^-$ modulo 2^B with

$$B \geq \left\lceil \log_2 \left(2 \binom{2g}{g} q^{g/2} \right) \right\rceil.$$

However, it is not sufficient to compute y^σ modulo 2^B since we need to take into account the loss of precision introduced during the reduction process of the differential forms. Let $y^\sigma \equiv \alpha_N + \beta_N y \bmod 2^N$ and write $\beta_N = \sum_{i=-L_N}^{B_N} V_i(x) S(x)^i$, then Lemma 1 implies that $L_N \leq 4(N-1)D - 2D$ and $B_N \leq 2(N-2)(d_S^f - 2d_S^h) + (d_S^f - d_S^h)$, with $d_S^f := \deg f / \deg S$ and $d_S^h := \deg h / \deg S$. Since we have to reduce the forms $x^{2i+1} y^\sigma \, dx$ for $i = 0, \ldots, 2g-1$, the loss of precision will be determined by the reduction of $x^{4g-1} V_{B_N}(x) S(x)^{B_N} y \, dx$ and $x V_{-L_N} S(x)^{-L_N} y \, dx$. The highest power of x appearing in the former differential form is less than $2N(\deg f - 2 \deg h) + 6g$ and by Lemma 2 the loss of precision is bounded by $c_{N,1} := 3 + \lfloor \log_2(2N(\deg f - 2 \deg h) + 7g + 1) \rfloor$. Similarly, Lemma 3 implies that the loss of precision introduced during the reduction of the latter differential form is bounded by $c_{N,2} := 3 + \lfloor \log_2(4ND - 6D + 1) \rfloor$. As a result, we conclude that it is sufficient to compute modulo 2^N with

$$N > B + \max(c_{N,1}, c_{N,2}). \tag{14}$$

4.2. *Detailed Algorithm*

The function `Hyperelliptic_Zeta_Function` given in Algorithm 1 computes the zeta function of a smooth projective hyperelliptic curve $\widetilde{C}$ defined over $\mathbb{F}_q$ with $q = 2^n$. Step 1 determines the minimal precision N satisfying inequality (14).

In step 2 we call the subroutine `Lift_Curve`, which first constructs an isomorphic curve such that $\overline{H}(x) \mid \overline{h}(x)$ and $\overline{H}(x) \mid \overline{f}(x)$ and lifts the curve using the construction described in Section 3.2. The result of this function is a hyperelliptic curve $C : y^2 + h(x)y = f(x)$ over R, a polynomial $H(x)$ and an integer D such that $H(x) \mid h(x)$, $H(x) \mid f(x)$ and $h(x) \mid H(x)^D$. Since this function is rather straightforward, we have omitted the pseudo-code.

In step 3 we compute $y^\sigma \bmod 2^N$ using the function `Lift_Frobenius_y` given in Algorithm 2. The parameters α_N, β_N are Laurent polynomials in S with coefficients polynomials over $R \bmod 2^N$ of degree $< \deg S$. This function implements the Newton iteration (1), but has quadratic, instead of linear, convergence. Note that Algorithm 2 is in fact a double Newton iteration: $\alpha + \beta y$ converges to y^σ, whereas $\gamma + \delta y$ is an approximation of the inverse of the denominator (2) in the Newton iteration.

Once we have determined an approximation of y^σ, we compute the action of σ_* on the basis of $H^1(\overline{A}/K)^-$ as $2x^{2i+1} y^\sigma \, dx$ for $i = 0, \ldots, 2g-1$. In step 4 we reduce these forms using the function `Reduce_MW_Cohomology` given in Algorithm 3, which is based on the reduction formulae devised in Section 3.2. Given a differential $Gy \, dx$ with G a Laurent polynomial in S, this function computes a polynomial $\Lambda \in K[x]$, with

Algorithm 1 (Hyperelliptic_Zeta_Function).

IN: *Hyperelliptic curve* $\overline{C}$ *over* $\mathbb{F}_q$ *given by equation* $y^2 + \overline{h}(x)y = \overline{f}(x)$.
OUT: *The zeta function* $Z(\widetilde{C}/\mathbb{F}_q; t)$.
1. $B = \left\lceil \log_2 \left(2\binom{2g}{g} q^{g/2} \right) \right\rceil$; $N > B + \max(c_{N,1}, c_{N,2})$;
2. $(h(x), f(x), H(x), D) =$ `Lift_Curve(`$\overline{h}(x), \overline{f}(x)$`)`;
3. $\alpha_N, \beta_N =$ `Lift_Frobenius_y(`h, f, H, D, N`)`;
4. `For` $i = 0$ `To` $2g - 1$ `Do`
   4.1. $R_i(x) =$ `Reduce_MW_Cohomology(`$2x^{2i+1}\beta_N, h, f, H, B$`)`;
   4.2. `For` $j = 0$ `To` $2g - 1$ `Do` $M[j][i] =$ `Coeff(`R_i, j`)`;
5. $M_F = MM^{\sigma} \cdots M^{\sigma^{n-1}} \bmod 2^B$;
6. $\chi(T) =$ `Characteristic_Pol(`M_F`)` $\bmod 2^B$;
7. `For` $i = 0$ `To` g `Do`
   7.1. `If Coeff(`$\chi, 2g - i$`)` $> \binom{2g}{i} q^{i/2}$ `Then Coeff(`$\chi, 2g - i$`)` $-= 2^B$;
   7.2. `Coeff(`χ, i`)` $= q^{g-i}$ `Coeff(`$\chi, 2g - i$`)`;
8. `Return` $Z(\widetilde{C}/\mathbb{F}_q; t) = \dfrac{t^{2g}\chi(1/t)}{(1-t)(1-qt)}$.

deg $\Lambda < 2g$ such that for a given precision B we have the following equivalence modulo exact and invariant forms $\Lambda y\,dx \sim Gy\,dx \bmod 2^B$, where $\bmod 2^B$ means modulo $2^B(Ry\,dx + \cdots + Rx^{2g-1}y\,dx)$. In step 2.3 we use the function `XGCD` which takes as input two polynomials $A(x), B(x) \in K[x]$ and returns polynomials $C(x), L_A(x), L_B(x)$ such that $C(x) = \gcd(A(x), B(x))$ and $C(x) = L_A(x)A(x) + L_B(x)B(x)$. Note that the

Algorithm 2 (Lift_Frobenius_y).

IN: *Curve* $C : y^2 + h(x)y = f(x)$ *over* R, *polynomial* $H(x) \in R[x]$ *with* $H|h$ *and* $H|f$, $D \in \mathbb{N}$ *such that* $h|H^D$ *and precision* N.
OUT: *Laurent polynomials* α_N, β_N *in* S *with* $S = H$ *if* $\deg H > 0$, $S = x$ *if* $H = 1$ *satisfying* $y^{\sigma} \equiv \alpha_N + \beta_N y \bmod 2^N$.
1. $B = \lceil \log_2 N \rceil + 1$; $T = N$; $Q_S := S^D \operatorname{div} h$;
2. `For` $i = B$ `Down To` 1 `Do` $P[i] = T$; $T = \lceil T/2 \rceil$;
3. $\alpha \equiv f \bmod 2$; $\beta \equiv -h \bmod 2$; $\gamma = 1$; $\delta = 0$;
4. `For` $i = 2$ `To` B `Do`
   4.1. $T_A \equiv \left((\alpha + h^{\sigma}) \cdot \alpha + \beta^2 \cdot f - f^{\sigma}\right) \cdot Q_S^2 \cdot S^{-2D} \bmod 2^{P[i]}$;
   4.2. $T_B \equiv (2\alpha - h \cdot \beta + h^{\sigma}) \cdot \beta \cdot Q_S^2 \cdot S^{-2D} \bmod 2^{P[i]}$;
   4.3. $D_A \equiv 1 + (h^{\sigma} - h^2 + 2\alpha) \cdot Q_S^2 \cdot S^{-2D} \bmod 2^{P[i-1]}$;
   4.4. $D_B \equiv 2\beta \cdot Q_S^2 \cdot S^{-2D} \bmod 2^{P[i-1]}$;
   4.5. $V_A \equiv D_A \cdot \gamma + D_B \cdot \delta \cdot f - 1 \bmod 2^{P[i-1]}$;
   4.6. $V_B \equiv D_A \cdot \delta + D_B \cdot (\gamma - \delta \cdot h) \bmod 2^{P[i-1]}$;
   4.7. $\gamma \equiv \gamma - (V_A \cdot \gamma + V_B \cdot \delta \cdot f) \bmod 2^{P[i-1]}$;
   4.8. $\delta \equiv \delta - (V_A \cdot \delta + V_B \cdot (\gamma - \delta \cdot h)) \bmod 2^{P[i-1]}$;
   4.9. $\alpha \equiv \alpha - (T_A \cdot \gamma + T_B \cdot \delta \cdot f) \bmod 2^{P[i]}$;
   4.10. $\beta \equiv \beta - (T_A \cdot \delta + T_B \cdot (\gamma - \delta \cdot h)) \bmod 2^{P[i]}$;
5. `Return` $\alpha_N = \alpha$, $\beta_N = \beta$.

Algorithm 3 (Reduce_MW_Cohomology).

IN: *Polynomials* $h(x)$, $f(x)$, $H(x) \in R[x]$ *with* $H|h$ *and* $H|f$, H *monic*, *Laurent polynomial* $G = \sum T_i(x)S(x)^i$ *with* $S = H$ *if* $\deg H > 0$ *and* $S = x$ *if* $H = 1$, $T_i(x) \in R[x]$ *with* $\deg T_i < \deg S$, *precisions* B, N.
OUT: $\Lambda \in K[x]$, *with* $\deg \Lambda < 2g$ *such that* $\Lambda y\,dx \sim Gy\,dx \bmod 2^B$.
1. $Q_f = f \text{ div } S$; $Q_h = h \text{ div } S$; $P = 0$; $V = 0$; $v_G =$ `Valuation(G)`;
2. `For` $i = v_G$ `To` -1
   2.1. $V \equiv P +$ `Coeff(G,i)` $\bmod 2^N$;
   2.2. $P \equiv V \text{ div } S \bmod 2^N$; $V \equiv V - P \cdot S \bmod 2^N$;
   2.3. $(1, L_A, L_B) =$ `XGCD(S,` $Q_f \cdot S'$`)`;
   2.4. $L_A = V \cdot L_A \bmod 2^N$; $L_B = V \cdot L_B \bmod 2^N$;
   2.5 $P \equiv P + L_A + \dfrac{L_B \cdot (-iQ_h^2 \cdot S' - 3(2Q_f' + Q_h \cdot h')) - L_B' \cdot (4Q_f + Q_h h)}{6+4i} \bmod 2^N$;
3. $d_G =$ `Degree(G)`; $d_T = (d_G + 1) \cdot$ `Degree(S)`; $T = 0$;
4. `For` $i = d_G$ `Down To` 0 `Do` $T = T \cdot S +$ `Coeff(G,` i`)` $\bmod 2^N$; $T = T + P$;
5. `For` $i = d_T$ `Down To` $2g$
   5.1. $P \equiv x^{i-2g}(2f' + h \cdot h') + \dfrac{i-2g}{3} x^{i-2g-1}(4f + h^2) \bmod 2^N$;
   5.2 $T \equiv T -$ (`Coeff(T,` i`)` $\cdot P$)/ `Coeff(P,` i`)` $\bmod 2^N$;
6. `Return` $\Lambda \equiv T \bmod 2^B$.

remarks after Lemmata 2 and 3 imply that the result of Algorithm 3 is correct modulo 2^B since we computed modulo 2^N and N satisfies $N - \max(c_{N,1}, c_{N,2}) > B$. The result of step 4 of Algorithm 1 is an approximation modulo 2^B of the matrix M through which σ_* acts on $H^1(\overline{A}/K)^-$. In step 5 we compute its norm M_F as $MM^\sigma \cdots M^{\sigma^{n-1}}$. Note that since M is not necessarily defined over R, we could lose up to cn bits of precision, where 2^c is the largest denominator appearing in M. By Lemmata 1 and 2, c is bounded by $O(\log g)$ independently of n. In theory we would therefore have to replace the bound B in Algorithm 1 by $B + cn$, which does not change the complexity of the algorithm.

In practise however it turns out that the largest denominator appearing in M_F is almost always the same as the largest denominator appearing in M and therefore it is not necessary to increase B. This phenomenon can be heuristically explained as follows: since the eigenvalues of $F_* = \sigma_*^n$ on $H^1(\overline{A}/K)^-$ have non-negative 2-adic valuation there is an R-submodule of $H^1(\overline{A}/K)^-$ which is stable under the action of σ_*. For this R-submodule we can take for instance the canonical image of the crystalline cohomology of C over R. Note that the R-submodule generated by $x^i y\,dx$ for $i = 0, \ldots, 2g-1$ is not canonical and in general not stable under σ_*. Let A_0 be the matrix that expresses $x^i y\,dx$ for $i = 0, \ldots, 2g-1$ in terms of a basis of such a stable R-submodule and let A be A_0 times a power of 2 such that A is a matrix over R which is not zero modulo 2. Then $M = A^{-1}UA^\sigma$ where U is the matrix of σ_* with respect to the new basis. Note that U is a matrix over R and that the norm of M equals $A^{-1}UU^\sigma \cdots U^{\sigma^{n-1}}A$. Thus the loss of precision is no more than $2d$ bits where d is the 2-adic valuation of $\det(A)$. If U and A are generic enough then $|c-d|$ is small. Furthermore, the bound (14) turns out to be slightly larger than what is needed and compensates for the loss of $2d$ bits.

In steps 6 and 7 we recover the characteristic polynomial of Frobenius from the first g coefficients of the characteristic polynomial of M_F. Finally, we return the zeta function of the smooth projective hyperelliptic curve $\widetilde{C}$ birational to $\overline{C}$ in step 8.

4.3. *Complexity*

In this section we analyse the space and time requirements of Algorithm 1 for a genus g hyperelliptic curve over $\mathbb{F}_{2^n}$ assuming fast arithmetic, i.e. using the Schönhage–Strassen algorithm [37] that computes the product of two m-bit integers in time $O(m^{1+\varepsilon})$ for any constant $\varepsilon \in \mathbb{R}_{>0}$. Before proceeding through the individual steps of the algorithm, we analyse the complexity of the basic operations in Algorithm 1 and the asymptotic behaviour of the bounds given in Lemma 1.

For a fixed precision N, let R_N denote the degree n unramified extension of $\mathbb{Z}_2/2^N\mathbb{Z}_2$. Elements of R_N are represented as polynomials over $\mathbb{Z}/2^N\mathbb{Z}$ modulo a sparse irreducible polynomial $P(t)$ of degree n. Since each element of this ring requires $O(nN)$ space, we can perform the basic operations, i.e. multiplication and division, in time $O(n^{1+\varepsilon}N^{1+\varepsilon})$.

Computing the Frobenius substitution σ on R_N can be accomplished in time $O(n^{2+\varepsilon}N^{1+\varepsilon})$ as follows. Since t is a root of $P(t)$, t^σ will also be a root of $P(t)$ and $t^\sigma \equiv t^2 \bmod 2$. Therefore, t^σ can be computed using the Newton iteration $T_{k+1} = T_k - P(T_k)/P'(T_k)$ initialised with t^2. Since the Newton iteration converges quadratically and we compute with the minimal precision in each step, the total complexity will be determined by the last step which takes $O(n)$ multiplications in R_N. Precomputing $t^\sigma \bmod 2^N$ can thus be accomplished in time $O(n^{2+\varepsilon}N^{1+\varepsilon})$. After this precomputation, we can compute the Frobenius substitution of any element $E(t)$ as $E(t^\sigma)$, which needs $O(n)$ multiplications in R_N and thus takes $O(n^{2+\varepsilon}N^{1+\varepsilon})$ time.

Lemma 1 bounds the maximum bit-size of the Laurent series we compute with and therefore determines the complexity of Algorithm 1. Since these bounds depend on the degree and splitting type of $h(x)$, we make a distinction between average-case and worst-case complexity. To this end we introduce three parameters which allow us to analyse both cases simultaneously.

- Let the asymptotic behaviour of $\deg f - 2\deg h$ be $O(g^\lambda)$. Since the degree of $f(x)$ is $2g+1$ and $h(x)$ is a random polynomial of degree $\leq g$, we conclude that $\lambda = 0$ on average and $\lambda = 1$ in the worst case.
- Let the asymptotic behaviour of $\deg H$ be $O(g^\mu)$. With very high probability a random polynomial of degree $\leq g$ has $O(g)$ different roots, which implies that $\mu = 1$ on average and $\mu = 0$ in the extreme case.
- Let the asymptotic behaviour of D be $O(g^\nu)$, then $\nu = 0$ on average and $\nu = 1$ in the worst case, since with very high probability a random polynomial only has roots with multiplicity $O(1)$.

In step 1 of Algorithm 1 we determine the minimal precision N satisfying inequality (14), which implies that N is $O(gn)$. The function `Lift_Frobenius_y` in step 3 is a Newton lifting. Since the precision doubles in every iteration, we see that its complexity is determined by the last iteration, which consists of $O(1)$ multiplications of Laurent polynomials in S with coefficients polynomials over R_N of degree less than $\deg S$. Lemma 1 implies that the bit-size of these objects is $O((g^\lambda + g^{\mu+\nu})nN^2)$. Since

the cost of the other operations in `Lift_Frobenius_y`, e.g. computing the Frobenius substitution of $O(g)$ elements of R_N, is less than the $O(1)$ multiplications, the overall time complexity of step 3 is $O((g^\lambda + g^{\mu+\nu})^{1+\varepsilon} n^{1+\varepsilon} N^{2+\varepsilon})$.

In step 4 of Algorithm 1 we reduce the $2g$ differential forms $2x^{2i+1}\beta_N y\ dx$ for $i = 0, \ldots, 2g-1$ using the function `Reduce_MW_Cohomology` given in Algorithm 3. In step 2 the dominant operations are $O(1)$ multiplications of polynomials over R_N of degree $O(g)$ and the extended GCD computation of two such polynomials. The former operation clearly takes time $O(g^{1+\varepsilon} n^{1+\varepsilon} N^{1+\varepsilon})$ and using Moenck's algorithm [30] the latter operation can also be performed in time $O(g^{1+\varepsilon} n^{1+\varepsilon} N^{1+\varepsilon})$. Lemma 1 implies that these operations have to be repeated $O(g^\nu N)$ times, so the time complexity of step 2 is $O(g^{1+\nu+\varepsilon} n^{1+\varepsilon} N^{2+\varepsilon})$. Write β_N as $\sum_{i=-L_N}^{B_N} V_i(x) S(x)^i$, then step 4 essentially is Horner's rule to compute $\sum_{i=0}^{B_N} V_i(x) S(x)^i$. Note that in practise we perform this step only once for all of the $2g$ reductions and use a binary tree algorithm which is asymptotically faster than Horner's method. The complexity of step 4 then becomes $O(g^{\lambda+\varepsilon} n^{1+\varepsilon} N^{2+\varepsilon})$. Lemma 1 implies that substeps 5.1 and 5.2 have to be executed $O(g^\lambda N)$ times and since each iteration consists of $O(g)$ multiplications and $O(1)$ divisions in R_N, the time complexity of step 5 is $O(g^{1+\lambda} n^{1+\varepsilon} N^{2+\varepsilon})$. Since we have to reduce $O(g)$ differential forms, the overall time complexity of step 4 of Algorithm 1 is $O((g^{2+\lambda} + g^{2+\nu+\varepsilon}) n^{1+\varepsilon} N^{2+\varepsilon})$.

In step 5 we need to determine the norm of a $2g \times 2g$ matrix M over K as $M M^\sigma \cdots M^{\sigma^{n-1}}$. This can be accomplished by computing $M_{i+1} = M_i M_i^{\sigma^{2^i}}$ for $i = 0, \ldots, \lfloor \log_2 n \rfloor$ with $M_0 = M$ and combining these to recover the norm of M. This process takes $O(\log n)$ multiplications of $2g \times 2g$ matrices at a cost of $O(g^3 n^{1+\varepsilon} N^{1+\varepsilon})$ time and $O(g^2 \log n)$ applications of powers of σ which takes $O(g^2 n^{2+\varepsilon} N^{1+\varepsilon})$ time if we precompute $t^{\sigma^{2^i}}$ for $i = 0, \ldots, \lfloor \log_2 n \rfloor$. The overall time complexity of step 5 thus becomes $O((n+g) g^2 n^{1+\varepsilon} N^{1+\varepsilon})$.

Finally, we need to compute the characteristic polynomial of a $2g \times 2g$ matrix over K, which can be done using the classical algorithm based on the Hessenberg form [6, Section 2.2.4]. The complexity of this algorithm is $O(g^3)$ ring operations or $O(g^3 n^{1+\varepsilon} N^{1+\varepsilon})$ time.

Since (14) implies that N is $O(gn)$, we have proved the following theorem.

Theorem 2. *The zeta function of a hyperelliptic curve of genus g defined over $\mathbb{F}_{2^n}$ can be computed in $O((g^\lambda + g^\nu) g^{4+\varepsilon} n^{3+\varepsilon})$ time and $O((g^\lambda + g^{\mu+\nu}) g^2 n^3)$ space, where λ, μ and ν are defined as in the beginning of Section* 4.3. *This implies the following complexities*:

- *Average case*: $O(g^{4+\varepsilon} n^{3+\varepsilon})$ *time and* $O(g^3 n^3)$ *space*.
- *Worst case*: $O(g^{5+\varepsilon} n^{3+\varepsilon})$ *time and* $O(g^4 n^3)$ *space*.

5. Implementation and Numerical Results

In this section we present running times of an implementation of Algorithm 1 in the `C` programming language and give some examples of Jacobians of hyperelliptic curves with almost prime group order.

Table 1. Running time and memory usage for genus 2, 3 and 4 hyperelliptic curves over $\mathbb{F}_{2^n}$

Size of Jacobian	Genus 2 curves		Genus 3 curves		Genus 4 curves	
gn	Time (s)	Mem (MB)	Time (s)	Mem (MB)	Time (s)	Mem (MB)
120	22	4.5	28	5.4	26	5.2
144	35	5.7	46	7.3	43	7.2
168	60	8.6	78	11	76	11
192	89	13	112	14	109	13
216	143	16	171	17	157	16

The basic operations on integers modulo 2^N for $N \leq 256$ were written in assembly language. Elements of R_N are represented as polynomials over $\mathbb{Z}/2^N\mathbb{Z}$ modulo a degree n irreducible polynomial, which we chose to be either a trinomial or a pentanomial. For multiplication of elements in R_N, polynomials over R_N and Laurent series over $R_N[x]$ we used Karatsuba's trick [18], which allows to multiply two m-bit integers in time $O(m^{\log_2 3})$. Redoing the complexity analysis then results in an average-case time complexity of $O(g^{5.17}n^{4.75})$ bit-operations.

5.1. *Running Times and Memory Usage*

Table 1 contains running times and memory usages of Algorithm 1 for genus 2, 3 and 4 hyperelliptic curves over various finite fields of characteristic 2. These results were obtained on an AMD XP 1700+ processor running Linux Redhat 7.1. Note that the field degrees are chosen such that gn, and therefore the size of the group order of the Jacobian, is constant across each row.

Although of no importance to cryptography, it is worth mentioning that Algorithm 1 is also practical for large genus hyperelliptic curves, e.g. the zeta function of a genus 350 hyperelliptic curve over $\mathbb{F}_2$ can be computed in 83 hours. For more information, we refer the interested reader to Section 4.4.4 of [41].

5.2. *Hyperelliptic Curve Examples*

In this subsection we give three examples of Jacobians of hyperelliptic curves with almost prime group order. The correctness of these results is easily proved by multiplying a random divisor with the given group order and verifying that the result is principal, i.e. is the zero element in the Jacobian $J_{\widetilde{C}}(\mathbb{F}_q)$.

It is clear that the given curves are non-supersingular, since the coefficient a_g of $\chi(T)$ is odd [12]. Furthermore, all curves are immune to Weil descent [13] and multiplicative reduction [11].

Let $\alpha = \sum_{i=0}^{n-1} \alpha_i t^i \in \mathbb{F}_{2^n}$, then α is represented by the integer $\sum_{i=0}^{n-1} \alpha_i 2^i$ written in hexadecimal notation.

Genus 2 *Hyperelliptic Curve over* $\mathbb{F}_{2^{83}}$

Let $\mathbb{F}_{2^{83}}$ be defined as $\mathbb{F}_2[t]/\overline{P}(t)$ with $\overline{P}(t) = t^{83} + t^7 + t^4 + t^2 + 1$ and consider the random hyperelliptic curve C_2 of genus 2 defined by

$$y^2 + \left(\sum_{i=0}^{2} h_i x^i\right) y = x^5 + \sum_{i=0}^{4} f_i x^i,$$

where

$$\begin{aligned} h_0 &= \texttt{4D168CAB78F1F7EB78D54} & h_1 &= \texttt{3B167A2F520486B2A8A60} \\ h_2 &= \texttt{507FC6D8D98A1411D1F24} \\ f_0 &= \texttt{6ABF379716E615F0997AF} & f_1 &= \texttt{1D13C5C10A58A238681F3} \\ f_2 &= \texttt{3ACC287DAA28D01EDDB58} & f_3 &= \texttt{74BF8FFD1A04B1E8B845B} \\ f_4 &= \texttt{10046A0ED36CF3B146071} \end{aligned}$$

The group order of the Jacobian $J_{\widetilde{C}_2}$ of C_2 over $\mathbb{F}_{2^{83}}$ is

$$\#J_{\widetilde{C}_2} = 2 \cdot 467680523946120545534688076793656194973179161188 93,$$

where the last factor is prime. The coefficients a_1 and a_2 of the characteristic polynomial of Frobenius $\chi(T) = T^4 + a_1T^3 + a_2T^2 + a_3T + a_4$ are given by

$$a_1 = 4789617893650 \quad \text{and} \quad a_2 = 12304549269471460402134471.$$

Genus 3 *Hyperelliptic Curve over* $\mathbb{F}_{2^{59}}$

Let $\mathbb{F}_{2^{59}}$ be defined as $\mathbb{F}_2[t]/\overline{P}(t)$ with $\overline{P}(t) = t^{59} + t^7 + t^4 + t^2 + 1$ and consider the random hyperelliptic curve C_3 of genus 3 defined by

$$y^2 + \left(\sum_{i=0}^{3} h_i x^i\right) y = x^7 + \sum_{i=0}^{6} f_i x^i,$$

where

$$\begin{aligned} h_0 &= \texttt{44EC0A3F607D5FE} & h_1 &= \texttt{183AFFC60B6C97A} \\ h_2 &= \texttt{5E8C286F052173E} & h_3 &= \texttt{39BFF4C327D0FCC} \\ f_0 &= \texttt{2CE03A6BD01418F} & f_1 &= \texttt{15160EE501EA31D} \\ f_2 &= \texttt{2DDF3B805A56673} & f_3 &= \texttt{72EAAC2B03D6F33} \\ f_4 &= \texttt{30BF8CAF4CF398A} & f_5 &= \texttt{288F45CEB700047} \\ f_6 &= \texttt{692BDF3913214F7} \end{aligned}$$

The group order of the Jacobian $J_{\widetilde{C}_3}$ of C_3 over $\mathbb{F}_{2^{59}}$ is

$$\#J_{\widetilde{C}_3} = 2 \cdot 95780971232851005943503002779523943538413536699032693,$$

where the last factor is prime. The coefficients a_1, a_2 and a_3 of the characteristic polynomial of Frobenius $\chi(T) = T^6 + a_1T^5 + a_2T^4 + a_3T^3 + a_4T^2 + a_5T + a_6$ are given by

$$\begin{aligned} a_1 &= -428922942, \\ a_2 &= 394510910624097420, \\ a_3 &= -30791687405615177802034 4677. \end{aligned}$$

Genus 4 *Hyperelliptic Curve over* $\mathbb{F}_{2^{47}}$

Let $\mathbb{F}_{2^{47}}$ be defined as $\mathbb{F}_2[t]/\overline{P}(t)$ with $\overline{P}(t) = t^{47} + t^5 + 1$ and consider the random hyperelliptic curve C_4 of genus 4 defined by

$$y^2 + \left(\sum_{i=0}^{4} h_i x^i\right) y = x^9 + \sum_{i=0}^{8} f_i x^i,$$

where

$$\begin{array}{lll} h_0 = \texttt{45EDAA69BB7B} & h_1 = \texttt{29185CC987F2} & h_2 = \texttt{5B56AF467634} \\ h_3 = \texttt{063A420D7308} & h_4 = \texttt{3AD67360D2FB} & \\ f_0 = \texttt{116A64DA4E4A} & f_1 = \texttt{1267C8BFEDF4} & f_2 = \texttt{5DED53867285} \\ f_3 = \texttt{3E2486D3500B} & f_4 = \texttt{66718C5D41BD} & f_5 = \texttt{5FBD515320F1} \\ f_6 = \texttt{4B960757EC52} & f_7 = \texttt{67B0202BA7D5} & f_8 = \texttt{545283F149A8} \end{array}$$

The group order of the Jacobian $J_{\widetilde{C}_4}$ of C_4 over $\mathbb{F}_{2^{43}}$ is

$$\#J_{\widetilde{C}_4} = 2 \cdot 196159429641733316151830117421270924231809135724223902787,$$

where the last factor is prime. The coefficients a_1, a_2, a_3 and a_4 of the characteristic polynomial of Frobenius $\chi(T) = T^8 + a_1 T^7 + a_2 T^6 + a_3 T^5 + a_4 T^4 + a_5 T^3 + a_6 T^2 + a_7 T + a_8$ are given by

$$\begin{aligned} a_1 &= 294806, \\ a_2 &= -5127513198846, \\ a_3 &= 236526738819576049756, \\ a_4 &= 31534922966327446198018115985. \end{aligned}$$

6. Conclusion

In this paper we have presented an extension of Kedlaya's algorithm to compute the zeta function of an arbitrary hyperelliptic curve C over a finite field of characteristic 2. The main difference with Kedlaya's algorithm is that the hyperelliptic curve can no longer be lifted arbitrarily; instead, a very specific lift is needed to ensure that the algebraic de Rham cohomology and the Monsky–Washnitzer cohomology are isomorphic. For a genus g hyperelliptic curve defined over $\mathbb{F}_{2^n}$, the average-case time complexity is $O(g^{4+\varepsilon} n^{3+\varepsilon})$ and the average-case space complexity is $O(g^3 n^3)$, whereas the worst-case time and space complexities are $O(g^{5+\varepsilon} n^{3+\varepsilon})$ and $O(g^4 n^3)$, respectively. An implementation in the C programming language shows that cryptographical sizes are now feasible for any genus g, e.g. computing the order of a 160-bit Jacobian of a hyperelliptic curve of genus 2, 3 or 4 takes about 75 seconds. Due to the generality of the cohomological approach, it seems likely that Kedlaya's algorithm can be extended to arbitrary curves. For a first step in this direction, we refer to [8] which presents an algorithm to compute the zeta function of any non-singular C_{ab} curve over a finite field of small characteristic.

References

[1] L. M. Adleman and M.-D. Huang. Counting rational points on curves and abelian varieties over finite fields. In H. Cohen, editor, *Algorithmic Number Theory. 2nd International Symposium. ANTS-II*, pages 1–16. Volume 1122 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 1996.

[2] S. Arita. Algorithms for computations in Jacobians of C_{ab} curves and their application to discrete-log-based public key cryptosystems. In *Proceedings of Conference on the Mathematics of Public Key Cryptography*, pages 165–175, 1999.

[3] A.O.L. Atkin. The number of points on an elliptic curve modulo a prime. Series of e-mails to the NMBRTHRY mailing list, 1992.

[4] I.F. Blake, G. Seroussi, and N.P. Smart. *Elliptic Curves in Cryptography*. London Mathematical Society Lecture Note Series, 265. Cambridge University Press, Cambridge, 1999.

[5] S. Bosch. A rigid analytic version of M. Artin's theorem on analytic equations. *Math. Ann.*, 255:395–404, 1981.

[6] H. Cohen. *A Course in Computational Algebraic Number Theory.* Volume 138 of Graduate Texts in Mathematics. Springer-Verlag, New York, 1993.

[7] J. Denef and F. Vercauteren. An extension of Kedlaya's algorithm to Artin–Schreier curves in characteristic 2. In C. Fieker and D.R. Kohel, editors, *Algorithmic Number Theory. 5th International Symposium. ANTS-V*, pages 308–323. Volume 2369 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2002.

[8] J. Denef and F. Vercauteren. Counting points on C_{ab} curves using Monsky–Washnitzer cohomology. Available at http://www.cs.bris.ac.uk/~frederik/, 2003.

[9] N. Elkies. Elliptic and modular curves over finite fields and related computational issues. In D. A. Buell and J. T. Teitelbaum, editors, *Computational Perspectives on Number Theory*, pages 21–76. American Mathematical Society/International Press, Providence, RI/Somerville, MA, 1998.

[10] R. Elkik. Solutions d'équations a coefficients dans un anneau henselien. *Ann. Sci. École Norm. Sup.*, 6(4):553–604, 1973.

[11] G. Frey and H.-G. Rück. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Math. Comp.*, 62(206):865–874, 1994.

[12] S. Galbraith. Supersingular curves in cryptography. In C. Boyd, editor, *Advances in Cryptology - ASIACRYPT* 2001, pages 495–513. Volume 2248 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2001.

[13] S. Galbraith. Weil descent of Jacobians. *Discrete Appl. Math.*, 128(1):165–180, 2003.

[14] S. Galbraith, S. Paulus, and N. Smart. Arithmetic on superelliptic curves. *Math. Comp.*, 71(237):393–405, 2002.

[15] P. Gaudry and N. Gürel. An extension of Kedlaya's algorithm for counting points on superelliptic curves. In C. Boyd, editor, *Advances in Cryptology - ASIACRYPT* 2001, pages 480–494. Volume 2248 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2001.

[16] P. Gaudry and R. Harley. Counting points on hyperelliptic curves over finite fields. In Wieb Bosma, editor, *Algorithmic Number Theory. 4th International Symposium. ANTS-IV*, pages 313–332. Volume 1838 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2000.

[17] P. Gaudry and É. Schost. Construction of secure random curves of genus 2 over prime fields. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology - EUROCRYPT* 2004, pages 239–256. Volume 3027 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2004.

[18] A. Karatsuba and Y. Ofman. Multiplication of multidigit numbers on automata. *Soviet Phys. Dokl.*, 7:595–596, 1963.

[19] K.S. Kedlaya. Counting points on hyperelliptic curves using Monsky–Washnitzer cohomology. *J. Ramanujan Math. Soc.*, 16:323–338, 2001.

[20] N. Koblitz. Elliptic curve cryptosystems. *Math. Comp.*, 48(177):203–209, 1987.

[21] N. Koblitz. Hyperelliptic cryptosystems. *J. Cryptology*, 1(3):139–150, 1989.

[22] A.G.B. Lauder and D. Wan. Counting points on varieties over finite fields of small characteristic. In J.P. Buhler and P. Stevenhagen, editors, *Algorithmic Number Theory*: *Lattices*, *Number Fields*, *Curves and Cryptography*. Cambridge University Press, Cambridge, 2002.

[23] A.G.B. Lauder and D. Wan. Computing zeta functions of Artin–Schreier curves over finite fields. *LMS J. Comput. Math.*, 5:34–55 (electronic), 2002.

[24] A.G.B. Lauder and D. Wan. Computing zeta functions of Artin–Schreier curves over finite fields, II. *J. Complexity*, 20:331–349, 2004.

[25] R. Lercier. Algorithmique des courbes elliptiques dans les corps finis. Ph.D. thesis, Laboratoire d'Informatique de l'École polytechnique (LIX), 1997.

[26] R. Lercier and D. Lubicz. A quasi quadratic time algorithm for hyperelliptic curve point counting. Preprint, 2003.

[27] J.-F. Mestre. Lettre adressée à Gaudry et Harley, December 2000. Available at `http://www.math.jussieu.fr/~mestre/`.

[28] J.-F. Mestre. Algorithmes pour compter des points en petite caractéristique en genre 1 and 2. Available at `http://www.math.univ-rennes1.fr/crypto/2001-02/mestre.ps`.

[29] V. S. Miller. Use of elliptic curves in cryptography. In H. C. Williams, editor, *Advances in Cryptology - CRYPTO* 1985, pages 417–426. Volume 218 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 1986.

[30] R.T. Moenck. Fast computation of GCDs. *Fifth Annual ACM Symposium on Theory of Computing*, pages 142–151, 1973.

[31] P. Monsky. Formal cohomology, II: The cohomology sequence of a pair. *Ann. of Math.*, 88:218–238, 1968.

[32] P. Monsky. Formal cohomology, III: Fixed point theorems. *Ann. of Math.*, 93:315–343, 1971.

[33] P. Monsky. *p-Adic Analysis and Zeta Functions*. Lectures in Mathematics, Department of Mathematics Kyoto University. 4. Tokyo, Japan, 1970.

[34] P. Monsky and G. Washnitzer. Formal cohomology, I. *Ann. of Math.*, 88:181–217, 1968.

[35] J. Pila. Frobenius maps of abelian varieties and finding roots of unity in finite fields. *Math. Comp.*, 55(192):745–763, 1990.

[36] T. Satoh. The canonical lift of an ordinary elliptic curve over a finite field and its point counting. *J. Ramanujan Math. Soc.*, 15:247–270, 2000.

[37] A. Schönhage and V. Strassen. Schnelle Multiplikation grosser Zahlen. *Computing* (*Arch. Elektron. Rechnen*), 7:281–292, 1971.

[38] R. Schoof. Elliptic curves over finite fields and the computation of square roots mod p. *Math. Comp.*, 44(170):483–494, 1985.

[39] M. van der Put. The cohomology of Monsky and Washnitzer. *Mém. Soc. Math. France*, 23:33–60, 1986.

[40] F. Vercauteren. Computing zeta functions of hyperelliptic curves over finite fields of characteristic 2. In M. Yung, editor, *Advances in Cryptology - CRYPTO* 2002, pages 369–384. Volume 2442 of Lecture Notes in Computer Science. Springer-Verlag, Berlin, 2002.

[41] F. Vercauteren. Computing Zeta Functions of Curves over Finite Fields. Ph.D. thesis, Katholieke Universiteit Leuven, November 2003. Available at `http://www.cs.bris.ac.uk/~frederik/`.

[42] D. Wan. Computing zeta functions over finite fields. In *Finite Fields: Theory, Applications, and Algorithms*, pages 131–141. Volume 225 of *Contemporary Mathematics*, American Mathematical Society, Providence, RI, 1999.