An analysis of the infrastructure in real function fields

David J. Mireles Morales

Mathematics Department
Royal Holloway, University of London
david.mireles@gmail.com

Abstract. We construct a map injecting the set of infrastructure ideals in a real function field into the class group of the correspoding hyperelliptic curve. This map respects the 'group-like' structure of the infrastructure, as a consequence of this construction we show that calculating distances in the set of infrastructure ideals is equivalent to the DLP in the underlying hyperelliptic curve. We also give a precise description of the elements missing in the infrastructure to be a group.

1 Introduction

1.1 Historical overview

Let K be a quadratic number field and let $\mathcal{O}$ be an order in K with discriminant D. If K is an imaginary number field, there is a well-known bijection between the set of *reduced, positive definite* quadratic forms $\mathbf{F}$, and the ideal class group $\mathrm{Cl}(\mathcal{O})$ of $\mathcal{O}$. (see Lenstra [9])

When K is a real quadratic field this bijection no longer exists. Instead, there is a many-to-one map $\psi : \mathbf{F} \longrightarrow \mathrm{Cl}(\mathcal{O})$ from the group of reduced quadratic forms $\mathbf{F}$ to the ideal class group $\mathrm{Cl}(\mathcal{O})$. Shanks realized that the set $\mathcal{R} = \psi^{-1}(\mathcal{O})$ of quadratic forms mapping into the principal class could be given an algebraic structure, associating a *distance* to every element [14].

The set $\mathcal{R}$, together with the underlying structure described by Shanks is known as the infrastructure of $\mathcal{O}$. The fastest algorithms to compute the regulator of $\mathcal{O}$ known to-date use infrastructure, these algorithms can be found in [3]. A very good account of the theory behind these algorithms is presented in [13].

Buchmann and Williams presented a key exchange algorithm very similar to the Diffie-Hellman proposal using the infrastructure of an order $\mathcal{O}$ in a real number field K [1]. This proposal had a number of problems, including the need to deal with approximations to certain algebraic numbers, high bandwidth and ambiguity problems. This problems were overcome with the proposal by Scheidler, Stein and Williams in [12] to use the set of infrastructure ideals in the function field associated to a hyperelliptic curve given by a real model over a finite field. It is this proposal, and its succesive adaptations[11, 8, 7], that we study in this article.

1.2 Description of the article

The main result of this paper is Theorem 1, proving that there exists a map from the set of infrastructure ideals into the class group of the underlying hyperelliptic curve that preserves the "group-like"structure of the infrastructure. As a consequence of this result we show that calculating distances in the set of infrastructure ideals is equivalent to the DLP in the underlying hyperelliptic curve. This is a significant contribution as this result was previously known only in genus 1.

It has been claimed that Algorithm 6 provides the unique Diffie-Hellman-like key exchange protocol implemented over a non-group algebraic structure. We will show that there is a simple (and very natural) embedding of the infrastructure ideals into the class group of the curve that makes the group operations in $\mathrm{Cl}^0(C)$ compatible with those of the infrastructure. We also show that every algorithm using the infrastructure to obtain cryptographic primitives can be implemented more efficiently in the class group of the corresponding hyperelliptic curve C. This is not only because the class group of the curve fills the 'holes' that prevent $\mathcal{R}$ from being a group, but also because the representation of the elements of $\mathrm{Cl}^0(C)$ used when working with the infrastructure is not optimal.

2 Hyperelliptic Curves

All curves considered in this article will be hyperelliptic curves given by a plane real model over a field k with $\mathrm{char}(k) \neq 2$. Such a curve C will be given by an equation of the form $C : y^2 = F(x)$, where $F(x)$ is a monic polynomial in $k[x]$ with even degree and no repeated roots. The curve C will have two k-rational points at infinity, which we will denote ∞^+ and ∞^-. The hyperelliptic conjugate of a point P in C will be denoted $\overline{P}$.

2.1 The Group of Divisors

Definition 1. *Let C be an algebraic curve defined over a field k. The* group of divisors *on C is the group of finite formal sums $D = \sum n_i P_i$, for integers n_i and points P_i on $C(\overline{k})$. It is denoted as* $\mathrm{Div}(C)$.

Definition 2. *A divisor $D = \sum n_i P_i$ is said to be* effective *if every coefficient n_i is non-negative. The support of the divisor $D = \sum n_i P_i$ is the set of points P_i for which the coefficient n_i is nonzero. We say that the divisor*

$$D_z = \sum_i \max(n_i, 0) P_i,$$

is the divisor of zeros *of D. Analogously, the divisor*

$$D_p = \sum_i \min(n_i, 0) P_i,$$

is the divisor of poles *of D.*

The degree of the divisor $D = \sum n_i P_i$ is the integer $\deg(D) = \sum n_i$. We will denote the subgroup of degree 0 divisors as $\mathrm{Div}^0(C)$.

2.2 Principal Divisors

Given a curve C defined over the field k, let $\overline{k}(C)$ define the field of $\overline{k}$-valued rational functions on C. To every function $f \in \overline{k}(C)^*$ one can associate the divisor

$$\operatorname{div}(f) = \sum_{P \in C(\overline{k})} \operatorname{ord}_P(f),$$

where $\operatorname{ord}_P(f)$ is the order of the function f at P [6]. For every function f, the degree of the divisor $\operatorname{div}(f)$ is 0 [5][Corollary II.6.10].

Definition 3. *We say that a divisor D is* principal *if it is the divisor associated to a function $D = \operatorname{div}(f)$. We will denote the group of principal divisors on C as* $\operatorname{Prin}(C)$*.*

Definition 4. *Two divisors D_0 and D_1 are* linearly equivalent*, denoted $D_0 \equiv D_1$, if there is a function $f \in \overline{k}(C)$ such that*

$$\operatorname{div}(f) = D_1 - D_0.$$

2.3 The Divisor Class Group

Definition 5. *The* divisor class group *of the curve C is the group of divisor classes modulo principal divisors linear equivalence. We will denote it as* $\operatorname{Cl}(C)$*. The class of a divisor D in $\operatorname{Cl}(C)$ will be denoted by $[D]$.*

Definition 6. *Given two divisors D_1 and D_2, we will denote the set of pairs of integers ω^+, ω^- such that*

$$D_1 \equiv D_2 + \omega^+\infty^+ + \omega^-\infty^-,$$

as $\omega(D_1, D_2)$. We say that the numbers ω^+ and ω^- are counterweights *for D_1 and D_2 if $(\omega^+, \omega^-) \in \omega(D_1, D_2)$.*

The set $\omega(D_1, D_2)$ may be empty. If $[\infty^+ - \infty^-]$ is a torsion point on $\operatorname{Cl}^0(C)$, and the set $\omega(D_1, D_2)$ is not empty, then it is infinite; however this will not affect our algorithms. Given two divisors D_1 and D_2, calculating the values of the counterweights relating them is a difficult problem. When these values are needed in our algorithms, there will be a simple way to calculate them. Since every principal divisor has degree 0. It follows that the degree function is well defined on divisor classes. We define $\operatorname{Cl}^0(C)$ as the degree zero subgroup of $\operatorname{Cl}(C)$.

Definition 7. *If the curve C is defined over a non algebraically closed field k, we say that a divisor D is k-rational if it is invariant under the action of the Galois group* $\operatorname{Gal}(\overline{k}/k)$*. The group of k-rational divisors is denoted as* $\operatorname{Div}_k(C)$*.*

Note that points in the support of a k-rational divisor D might not be k-rational. Analogously, we say that a divisor class is k-rational if it is $\operatorname{Gal}(\overline{k}/k)$-stable. We denote the group of k-rational divisor classes as $\operatorname{Cl}_k(C)$.

Proposition 1. *Let D_∞ be a k-rational degree g divisor, and let $D \in \mathrm{Div}^0(C)$ be a k-rational divisor on the hyperelliptic curve C. Then $[D]$ has a unique representative in $\mathrm{Cl}^0(C)$ of the form $[D_0 - D_\infty]$, where D_0 is an effective k-rational divisor of degree g whose affine part is reduced.*

Proof. See [4]. □

2.4 Points at infinity

Let C be a hyperelliptic curve given by a real model, and denote the two points at infinity on C as ∞^+ and ∞^-. It is possible to prove that

$$(y/x^{g+1})(\infty^+) = 1, \quad (y/x^{g+1})(\infty^-) = -1.$$

Hence, for $p(x)$ a polynomial of the form $p(x) = (x^{g+1} + \sum_{0 \le i \le g} b_i x^i)$, the function $y - p(x)$ will have valuation strictly larger than $-(g+1)$ at ∞^+ and valuation $-(g+1)$ at ∞^-.

Definition 8. *In the notation of the previous paragraph, among all monic degree $g+1$ polynomials, there is a unique polynomial in $k[x]$ for which the valuation of the function at ∞^+ is maximal; we will denote this polynomial by H^+. Define the polynomial H^- as $H^-(x) = -H^+(x)$.*

If $C(x,y)$ is the equation of the curve, then $H^+(x)$ and $H^-(x)$ are the polynomials with leading coefficient 1 and -1 such that $C(x, H^\pm(x))$ has minimal degree. Their coefficients can thus be found recursively. The polynomials $H^\pm(x)$ are just a technical tool to specify a point at infinity, similar to the choice of sign when computing the square root of a complex number.

Definition 9. *Given two divisors D_1 and D_2, we will denote the set of pairs of integers ω^+, ω^- such that*

$$D_1 \equiv D_2 + \omega^+ \infty^+ + \omega^- \infty^-,$$

as $\omega(D_1, D_2)$. We say that the numbers ω^+ and ω^- are counterweights *for D_1 and D_2 if $(\omega^+, \omega^-) \in \omega(D_1, D_2)$.*

The set $\omega(D_1, D_2)$ may be empty. If $[\infty^+ - \infty^-]$ is a torsion point on $\mathrm{Cl}^0(C)$, and the set $\omega(D_1, D_2)$ is not empty, then it is infinite; however this will not affect our results.

2.5 Mumford representation

Definition 10. *We say that an effective divisor $D = \sum_i P_i$ on a hyperelliptic curve C is* semi-reduced *if $i \neq j$ implies $P_i \neq \overline{P}_j$. If the hyperelliptic curve C has genus g, we say that a divisor D on C is* reduced *if it is semi-reduced, and has degree $d \le g$. We will denote the degree of a divisor D_i as d_i.*

Let C be a hyperelliptic curve given by the equation $y^2 = F(x)$. To every pair of polynomials $(u(x), v(x))$ such that $F(x) - v(x)^2 \equiv 0 \mod u(x)$, we associate a divisor as follows

$$\text{If } u(x) = \prod_i (x - r_i), \text{ then } (u(x), v(x)) \mapsto \sum_i (r_i, v(r_i)).$$

Note that the divisor associated to the pair $(u(x)v(x))$ is always an effective affine semi-reduced divisor.

Conversely, if D is an effective, affine, semi-reduced divisor, there exists a pair of polynomials $(u(x), v(x))$ with $F(x) - v(x)^2 \equiv 0 \mod u(x)$ and such that the divisor associated to $(u(x), v(x))$ is D

Definition 11. *In the notation of the previous paragraph, we say that* $(u(x), v(x))$ *is a Mumford representation for D, and we will denote this as* $D = \operatorname{div}(u(x), v(x))$.

Remark 1. Note that for D is an effective reduced affine divisor, the second polynomial $v(x)$ in its Mumford representation $D = \operatorname{div}(u(x), v(x))$ is only well defined modulo $u(x)$. If we require that $\deg(v) < \deg(u)$ we can get a unique Mumford representation for every affine semi-reduced divisor.

3 Algorithms on the Mumford Representation of Divisors

Algorithm 1 Modified Composition

```
INPUT: Semi-reduced affine divisors $D_1 = \operatorname{div}(u_1, v_1)$ and $D_2 = \operatorname{div}(u_2, v_2)$.
OUTPUT: A semi-reduced affine divisor $D_3 = \operatorname{div}(u_3, v_3)$ and a pair $(\omega^+, \omega^-)$, such
    that $(\omega^+, \omega^-) \in \omega(D_1 + D_2, D_3)$.
Compute $s$ (monic), $f_1, f_2, f_3 \in k[x]$ such that
        $s = \gcd(u_1, u_2, v_1 + v_2 + h) = f_1 u_1 + f_2 u_2 + f_3(v_1 + v_2 + h).$
Set $u_3 := u_1 u_2 / s^2$ and $v_3 := (f_1 u_1 v_2 + f_2 u_2 v_1 + f_3(v_1 v_2 + F)) / s \mod u_3$
return $\operatorname{div}(u_3, v_3)$ and $(\deg(s), \deg(s))$.
```

The result D_3 of Algorithm 1 will be denoted $D_3, (\omega^+, \omega^-) = \operatorname{comp}(D_1, D_2)$. Algorithm 1 is also known as *divisor composition.*

Proposition 2. *Given two semi-reduced affine divisors* $D_1 = \operatorname{div}(u_1, v_1)$ *and* $D_2 = \operatorname{div}(u_2, v_2)$*, the divisor of the function s from Algorithm 1 is*

$$\operatorname{div}(s) = D_1 + D_2 - D_3 - \frac{d_1 + d_2 - d_3}{2}(\infty^+ + \infty^-), \tag{1}$$

in particular, $(\omega^+, \omega^-) \in \omega(D_1 + D_2, D_3)$.

Given an affine semi-reduced divisor D_0, of degree $d_0 \geq g+2$, Algorithm 2 finds another affine semi-reduced divisor D_1 with smaller degree d_1, and a pair of integers (ω^+, ω^-) such that

$$(\omega^+, \omega^-) \in \omega(D_0, D_1) \tag{2}$$

Algorithm 2 Modified Reduction

```
INPUT: A semi-reduced affine divisor $D_0 = \mathrm{div}(u_0, v_0)$, with $d_0 \geq g+2$.
OUTPUT: A semi-reduced affine divisor $D_1 = \mathrm{div}(u_1, v_1)$ and a pair $(\omega^+, \omega^-)$, such
    that $d_1 < d_0$ and Equation (2) holds.
Set $u_1 := (v_0^2 + hv_0 - F)/u_0$ made monic.
Let $v_1 := (-v_0 - h) \mod u_1$.
if the leading term of $v_0$ is $x^{g+1}$ then
    Let $(\omega^+, \omega^-) := (d_0 - g - 1, g + 1 - d_1)$.
else if the leading term of $v_0$ is $-x^{g+1}$ then
    Let $(\omega^+, \omega^-) := (g + 1 - d_1, d_0 - g - 1)$.
else
    Let $(\omega^+, \omega^-) := (\frac{d_0 - d_1}{2}, \frac{d_0 - d_1}{2})$.
end if
return $\mathrm{div}(u_1, v_1), (\omega^+, \omega^-)$.
```

Proposition 3. *Let $D_0 = \mathrm{div}(u_0, v_0)$ be a divisor of degree $d_0 \geq g+2$. Then, in the notation of Algorithm 2 we have*

$$\mathrm{div}\left(\frac{y - v_0(x)}{u_0}\right) = D_0 - D_1 - \omega^+ \infty^+ - \omega^- \infty^-. \tag{3}$$

Algorithm 3 Composition at Infinity and Reduction

```
INPUT: A semi-reduced affine divisor $D_0 = \mathrm{div}(u_0, v_0)$ of degree $d_0 \leq g+1$.
OUTPUT: A reduced affine divisor $D_1 = \mathrm{div}(u_1, v_1)$ and a pair of integers $(\omega^+, \omega^-)$
    such that $(\omega^+, \omega^-) \in \omega(D_0, D_1)$.
$v_1' := H^+ + (v_0 - H^+ \mod u_0)$,
$u_1 := (v_1'^2 - F)/u_0$ made monic.
$v_1 := -v_1' \mod u_1$.
Let $(\omega^+, \omega^-) := (d_0 - g - 1, g + 1 - d_1)$.
return $\mathrm{div}(u_1, v_1), (\omega^+, \omega^-)$.
```

Algorithm 3 is only defined for affine semi-reduced divisors on curves given by a real model. If it were applied on a divisor of degree at least $g+2$, Algorithm 3 would coincide with Algorithm 2. When applied on a divisor D_0 degree at most $g+1$, Algorithm 3 can be interpreted as composing the divisor D_0 with some divisor at infinity, followed by Algorithm 2. The polynomial v_1' in this algorithm is the equivalent to polynomial v_3 in Algorithm 1. The result D_1 of this algorithm will be denoted as $D_1, (\omega^+, \omega^-) = \mathrm{red}_\infty(D_0)$. Formally, the action of this algorithm is given by the following.

Proposition 4. *[Proposition 2 [4]] Given an effective semi-reduced divisor with affine support D_0, with Mumford representation $\mathrm{div}(u_0, v_0)$ and degree $d_0 \leq g+1$. If $D_1, (\omega^+, \omega^-) = \mathrm{red}_\infty(D_0)$, then in the notation of Algorithm 3,*

$$\mathrm{div}\left(\frac{y - v_1'(x)}{u_1}\right) = D_0 - D_1 - \omega^+\infty^+ - \omega^-\infty^-. \qquad (4)$$

In particular $(\omega^+, \omega^-) \in \omega(D_0, D_1)$.

Remark 2. If we abuse notation, it is possible to prove that the function $\mathrm{red}_\infty : \mathcal{R} \longrightarrow \mathcal{R}$ is a bijection. It is then possible to define a function ${\mathrm{red}_\infty}^{-1}$ on $\mathcal{R}$, this will be used in Proposition 8. Another way to calculate ${\mathrm{red}_\infty}^{-1}$ in $\mathcal{R}$ is simply by running Algorithm 3 using H^- in Step 1 instead. In this case the counterweights also need to be adapted.

4 Infrastructure

Let C be a genus g hyperelliptic curve given by a real model $C : y^2 = F(x)$ over a field k with $\mathrm{char}(k) \neq 2$. Denote its function field as $K = k(C)$, and let $\mathcal{O}$ be the affine coordinate ring of C, i.e. $\mathcal{O} = k[x, y]/(y^2 - F(x))$.

Every ideal $\mathfrak{a}$ of $\mathcal{O}$ has an $\mathcal{O}$-basis of the form $\mathfrak{a} = [SQ, S(y + P)]$, where S, Q, P are polynomials in $k[x]$ such that Q divides $P^2 - F$. The polynomials S and Q are uniquely defined up to multiplication by elements of k^*, and the polynomial P is only defined modulo Q. To have a unique basis for the ideal $\mathfrak{a}$ we will assume that $\deg(P) < \deg(Q)$.

Remark 3. Throughout this chapter, when we refer to the basis of an ideal we will assume that the basis has the form described in the previous paragraph.

Definition 12. *If the ideal $\mathfrak{a}$ has basis $[SQ, S(y+P)]$ as described in the previous paragraph, we define the* degree *of the ideal $\mathfrak{a}$ as $\deg(SQ)$.*

Definition 13. *We say that an ideal is* primitive *if the polynomial S can be taken to be $S = 1$.*

Definition 14. *We say that an ideal $\mathfrak{a} = [Q, P + y]$ is* reduced *if it is primitive and $\deg(\mathfrak{a}) \leq g$.*

Definition 15. *Let $\mathcal{R}$ be the set of principal reduced ideals of $\mathcal{O}$. We say that $\mathcal{R}$ is the set of infrastructure ideals of $\mathcal{O}$ [12].*

Definition 16. *Let $\mathfrak{a}$ be an infrastructure ideal. By definition $\mathfrak{a} = (\alpha)$ for some function α. We define the distance $\delta(\mathfrak{a})$ of the ideal $\mathfrak{a}$ as $\delta(\mathfrak{a}) = \mathrm{ord}_{\infty^+}(\alpha)$ [12].*

Example 1. The ring $\mathcal{O}$ can be seen as the ideal generated by the element 1. It is an element of $\mathcal{R}$ with basis $\mathcal{O} = [1, 0]$, and by definition it has distance 0.

If there is a unit β in $\mathcal{O}$ with non-zero valuation at ∞^+, then there is a least positive integer R for which there exists a unit β_R with valuation R at ∞^+. In this case, the distance of an ideal is only defined modulo R. The integer R is known as the *regulator* of $\mathcal{O}$.

Remark 4. The divisor of a unit β in $\mathcal{O}$ has to be supported exclusively at ∞^+ and ∞^-, and have degree 0. It follows that the regulator of $\mathcal{O}$ is given by the order of the element $\infty^+ - \infty^-$ in $\mathrm{Cl}^0(C)$. We prove some stronger results in Theorems 1 and 2.

Ideas related to the set of infrastructure ideals have found their main applications in cryptography. For these applications the curve C is defined over a finite field and the set $\mathcal{R}$ is finite.

Definition 17. *Given ideals $\mathfrak{a}_1, \mathfrak{a}_2 \in \mathcal{R}$, we define the* distance between $\mathfrak{a}_1$ and $\mathfrak{a}_2$ *to be*

$$\delta(\mathfrak{a}_1, \mathfrak{a}_2) = \delta(\mathfrak{a}_1) - \delta(\mathfrak{a}_2).$$

Given two random infrastructure ideals $\mathfrak{a}_1$ and $\mathfrak{a}_2$, finding the distance between them is a hard problem (see Theorem 3). A very good description of the ideas and techniques used in the infrastructure of a hyperelliptic curve given by a real model can be found in [10].

5 Operations on the Infrastructure Ideals

Let $\mathfrak{a}$ be a primitive ideal of $\mathcal{O}$ with basis $[Q, y + P]$. Note that the pair of polynomials (Q, P) satisfy all the conditions to be the Mumford representation of a divisor. In other words, there is an effective, affine, semi-reduced divisor D on the curve C such that $D = \mathrm{div}(Q, P)$.

Definition 18. *Given a primitive ideal $\mathfrak{a}$ of $\mathcal{O}$ with basis $[Q, y+P]$, we define the divisor associated to $\mathfrak{a}$ as the divisor $D = \mathrm{div}(\mathfrak{a})$ whose Mumford representation is (Q, P).*

Since the basis $[Q, y + P]$ of the ideal $\mathfrak{a}$ could also be thought of as the Mumford representation of a divisor, we can use Algorithm 1 (composition), Algorithm 2 (reduction) and Algorithm 3 (composition at infinity and reduction) on elements of $\mathcal{R}$. The idea of using these algorithms (or rather, a variant that does not compute counterweights) on ideals is not new (see [2, 10]). The interpretation of the action of these algorithms on infrastructure ideals is not obvious. In this section we give an interpretation both of the action of the algorithms on ideals of $\mathcal{O}$, and of the counterweights returned by the algorithms in terms of the distance.

Proposition 5. *Let $\mathfrak{a}_1$ and $\mathfrak{a}_2$ be two primitive ideals of $\mathcal{O}$. If $\mathfrak{a}_3, (\omega^+, \omega^-) = \mathrm{comp}(\mathfrak{a}_1, \mathfrak{a}_2)$ is the result obtained from applying Algorithm 1 on the basis of $\mathfrak{a}_1$ and $\mathfrak{a}_2$, we get*

$$\mathfrak{a}_1 \cdot \mathfrak{a}_2 = s \cdot \mathfrak{a}_3$$
$$\omega^+ = \mathrm{ord}_{\infty^+}(s),$$

where s denotes the polynomial obtained in Step 1 of Algorithm 1.

Proof. The first property is a classic result. See [12, Theorem 3.4]. The second follows from the fact that the order of a polynomial $p(x)$ at ∞^+ is given by the degree $\deg p(x)$.

Proposition 6. *Let $\mathfrak{a}_0$ be a primitive divisor with basis $\mathfrak{a}_0 = [Q_0, y + P_0]$ such that $\deg(Q_0) \geq g + 2$. If we let $\mathfrak{a}_1, (\omega^+, \omega^-) = \mathrm{red}(\mathfrak{a}_0)$ be the result of applying Algorithm 2 to the ideal $\mathfrak{a}_0$, then*

$$\mathfrak{a}_1 = \left(\frac{y - P_0}{Q_0}\right) \cdot \mathfrak{a}_0$$
$$\omega^+ = -\,\mathrm{ord}_{\infty^+}\left(\frac{y - P_0}{Q_0}\right).$$

Proof. If the ideal $\mathfrak{a}_0$ has basis $[Q_0, y + P_0]$, then the ideal $((y - P_0)/Q_0) \cdot \mathfrak{a}_0$ has basis $[y - P_0, (y^2 - P_0^2)/Q_0]$, which is by defintion a basis of $\mathfrak{a}_1$. The second assertion follows simply from Equation (3).

Proposition 7. *Let $\mathfrak{a}_0$ be a primitive divisor with basis $\mathfrak{a}_0 = [Q_0, y + P_0]$ such that $\deg(Q_0) \leq g + 1$. If we let $\mathfrak{a}_1, (\omega^+, \omega^-) = \mathrm{red}_\infty(\mathfrak{a}_0)$ be the result of applying Algorithm 3 to the ideal $\mathfrak{a}_0$, then*

$$\mathfrak{a}_1 = \left(\frac{y - P_0}{Q_0}\right) \cdot \mathfrak{a}_0,$$
$$\omega^+ = -\,\mathrm{ord}_{\infty^+}\left(\frac{y - P_0}{Q_0}\right).$$

Proof. The proof of the first property is analogous to the proof given for Proposition 6. The second property follows from Equation (4).

Corollary 1. *Let $\mathfrak{a}_0$ be an infrastructure ideal. If $\mathfrak{a}_1, (\omega^+, \omega^-) = \mathrm{red}_\infty(\mathfrak{a}_0)$, then*

$$\omega^+ = -\delta(\mathfrak{a}_0, \mathfrak{a}_1).$$

Proof. Proposition 7 shows that there is a function α with $\omega^+ = -\,\mathrm{ord}_{\infty^+}(\alpha)$ such that $\mathfrak{a}_1 = \alpha\mathfrak{a}_0$, the result follows from the definition of δ.

Definition 19. *Suppose that the regulator R is a positive integer. Given an integer n between 0 and $R - 1$, let*

$$\delta(n) = \max\{\delta(\mathfrak{a}) | \mathfrak{a} \in \mathcal{R} \text{ and } \delta(\mathfrak{a}) \leq n\},$$

and let $\mathfrak{a}_n$ be the ideal in $\mathcal{R}$ such that $\delta(\mathfrak{a}_n) = \delta(n)$. We say that $\mathfrak{a}_n$ is the ideal closest to the left *of n [12].*

The following result shows that in principle it is possible to find all the infrastructure ideals using only the algorithms we have presented, we omit the proof, but refer the reader to [12].

Proposition 8. *Let $\mathfrak{a}$ be an infrastructure ideal. Then the set $\{\mathrm{red}_\infty^i(\mathfrak{a})\}_{i\in\mathbf{Z}}$ is the set of infrastructure ideals $\mathcal{R}$.*

Proof. See [12, Section 3.1].

6 A Cryptographic Interlude

The cryptographic applications of infrastructure have been the motivation for most of the work done in the area. In this section we present the cryptographic protocols presented in [12] which use the set of infrastructure ideals as underlying algebraic structure. It has been claimed that this is the unique Diffie-Hellman-like key exchange protocol that doesn't use a group as underlying algebraic structure, we analyse this claim in the next section, see for example Theorem 1.

Algorithm 4 Constant Addition

```
INPUT: An ideal $\mathfrak{a}_0 \in \mathcal{R}$ and an integer $k$.
OUTPUT: The ideal $\mathfrak{a}_1$ closest to the left of $\delta(\mathfrak{a}_0) + k$.
if $k$ is positive then
    Use $H^+$ in the $\mathrm{red}_\infty$ steps.
    Let $\mathfrak{a}_2, (\omega^+, \omega^-) := \mathrm{red}_\infty(\mathfrak{a}_0)$.
    Let $\mathfrak{a}_1 := \mathfrak{a}_0,$, $n := \omega^+$.
    while $n < k$ do
        $\mathfrak{a}_1 := \mathfrak{a}_2$.
        $\mathfrak{a}_2, (\omega^+, \omega^-) := \mathrm{red}_\infty(\mathfrak{a}_0)$, $n := n + \omega^+$.
    end while
else if $k$ is negative then
    Use $H^-$ in the $\mathrm{red}_\infty$ steps.
    Let $\mathfrak{a}_2, (\omega^+, \omega^-) := \mathrm{red}_\infty(\mathfrak{a}_0)$.
    Let $\mathfrak{a}_1 := \mathfrak{a}_0$, $n := \omega^+$.
    while $n \geq k$ do
        $\mathfrak{a}_1 := \mathfrak{a}_2$.
        $\mathfrak{a}_2, (\omega^+, \omega^-) := \mathrm{red}_\infty(\mathfrak{a}_0)$, $n := n + \omega^+$.
    end while
end if
return $\mathfrak{a}_1$.
```

Given an infrastructure ideal $\mathfrak{a}_0$ with distance δ_0 and an integer k, Algorithm 4 finds the ideal closest to the left of $k + \delta_0$. We denote the result of Algorithm 4 as $\mathfrak{a}_1 = \mathrm{CA}(\mathfrak{a}_0, k)$.

In the context of affine semi-reduced effective divisors, we mentioned in Section 3 that Algorithm 3 (composition at infinity and reduction) has the same

Algorithm 5 Ideal Multiplication

```
INPUT: Ideals $\mathfrak{a}_1, \mathfrak{a}_2 \in \mathcal{R}$.
OUTPUT: The ideal $\mathfrak{a}_3$ closest to the left of $\delta(\mathfrak{a}_1) + \delta(\mathfrak{a}_2)$.
$\mathfrak{a}_3, (\omega^+, \omega^-) := \mathrm{comp}(\mathfrak{a}_1, \mathfrak{a}_2)$, $n := \omega^+$.
while $\deg(\mathfrak{a}_3) \geq g + 2$ do
    $\mathfrak{a}_3, (\omega^+, \omega^-) := \mathrm{red}(\mathfrak{a}_3)$, $n := n + \omega^+$.
end while
if $\deg(\mathfrak{a}_3) = g + 1$ then
    $\mathfrak{a}_3, (\omega^+, \omega^-) := \mathrm{red}_\infty(\mathfrak{a}_3)$, $n := n + \omega^+$.
end if
$\mathfrak{a}_3 := \mathrm{CA}(\mathfrak{a}_3, n)$.
return $\mathfrak{a}_3$.
```

action on a divisor D as Algorithm 2 (reduction) if $\deg(D) \geq g + 2$. In the context of infrastructure, it is customary to use only Algorithm 3; we have chosen to differentiate its use in Algorithm 5 precisely to separate the cases when only a reduction is taking place from those when some composition at infinity is also carried out. From a practical perspective the algorithms are identical, but we belive that conceptually they deserve being treated differently.

It can be proved that the maximum value of n in Step 8 in Algorithm 5 is bounded by $(g + 1)/2$ (see [12]). Algorithm 5 shows that given two ideals with distances δ_1 and δ_2, it is possible to find the ideal closest to the left of $\delta_1 + \delta_2$.

Combining Algorithm 5 with Algorithm 4, given an infrastructure ideal $\mathfrak{a}$ with distance δ and an integer k, it is possible to find the ideal closest to the left of $k\delta$, even if δ is not known, in time $O(\log(k))$ (see Algorithm POWER in [12]). This construction can then be used to implement a key-exchange protocol modeled on Diffie-Hellman using the infrastructure ideals.

Algorithm 6 Infrastructure Diffie-Hellman

```
PUBLIC INFORMATION: An ideal $\mathfrak{a} \in \mathcal{R}$ and its distance $\delta = \delta(\mathfrak{a})$.
1: Alice generates a random ideal $\mathfrak{a}_A$ with distance $\delta(\mathfrak{a}_A) = a\delta$. Alice knows $a$.
2: Bob generates a random ideal $\mathfrak{a}_B$ with distance $\delta(\mathfrak{a}_B) = b\delta$. Bob knows $b$.
3: Alice and Bob exchange $\mathfrak{a}_A$ and $\mathfrak{a}_B$.
4: Alice and Bob compute the ideal $\mathfrak{a}_C$ closes on the left to $ab\delta$.
5: Alice and Bob use $\mathfrak{a}_C$ as the key in a symmetric encryption scheme.
```

7 A map into the class group

As mentioned before, it is claimed that Algorithm 6 provides a Diffie-Hellman-type key construction algorithm in a non-group structure. In this section we will explain that the failure of $\mathcal{R}$ to be a group is somewhat artificial. The results in this section are based on the construction of a map relating the set of infrastructure ideals with certain divisor classes in $\mathrm{Cl}^0(C)$.

Definition 20. *Given $\mathfrak{a} \in \mathcal{R}$ an infrastructure ideal, define*

$$\begin{aligned}\psi :& \mathcal{R} \longrightarrow \mathrm{Cl}^0(C)\\ &\mathfrak{a} \mapsto [\mathrm{div}(\mathfrak{a}) - \deg(\mathfrak{a})\infty^-],\end{aligned}$$

where $\mathrm{div}(\mathfrak{a})$ *refers to the affine effective semi-reduced divisor associated to* $\mathfrak{a}$ *(see Definition 18).*

Proposition 9. *Let $\mathfrak{a}_1$ and $\mathfrak{a}_2$ be two infrastructure ideals. If δ is the distance $\delta(\mathfrak{a}_1, \mathfrak{a}_2)$ between $\mathfrak{a}_1$ and $\mathfrak{a}_2$, then*

$$\psi(\mathfrak{a}_1) + \delta[\infty^+ - \infty^-] = \psi(\mathfrak{a}_2). \tag{5}$$

Proof. Proposition 8 shows that one can reach any element of $\mathcal{R}$ using succesive applications of red_∞ on $\mathfrak{a}_1$, so it suffices to prove this result for ideals $\mathfrak{a}_1$ and $\mathfrak{a}_2, (\omega^+, \omega^-) = \mathrm{red}_\infty(\mathfrak{a}_1)$. Let $D_1 = \mathrm{div}(\mathfrak{a}_1)$ and $D_2 = \mathrm{div}(\mathfrak{a}_2)$ be affine divisors of degrees d_1 and d_2 respectively. Step 4 in Algorithm 3 says that $(\omega^+, \omega^-) = (d_1 - g - 1, g + 1 - d_2)$, and using Equation (4) we get

$$D_1 \equiv D_2 + (d_1 - (g+1))\infty^+ + (g + 1 - d_2)\infty^-.$$

This implies

$$(D_1 - d_1\infty^-) + (g + 1 - d_1)(\infty^+ - \infty^-) \equiv (D_2 - d_2\infty^-),$$

Since $\mathfrak{a}_1 = \mathrm{red}_\infty(\mathfrak{a}_0)$, Corollary 1 proves that $\delta = -\omega^+$, and since $\omega^+ = d_1 - g - 1$. We can finally conclude that

$$\psi(a) + \delta P = \psi(b).$$

Theorem 1. *The map $\psi : \mathcal{R} \longrightarrow \mathrm{Cl}^0(C)$ sends an ideal $\mathfrak{a}$ with distance $\delta = \delta(\mathfrak{a})$ to the element $\psi(\mathfrak{a}) = \delta[\infty^+ - \infty^-]$ of $\mathrm{Cl}^0(C)$. Nota that this implies that the map ψ respects the 'group-like' structure of the infrastructure.*

Proof. The result is trivial for $\mathfrak{a} = \mathcal{O}$, since we have mentioned that $\deg(\mathcal{O}) = 0$ and by definition $\psi(\mathcal{O}) = 0$. It extends inductively to $\mathcal{R}$ using Proposition 9.

Corollary 2. *Let $\mathfrak{a}$ be an infrastructure ideal with distance δ, then the Mumford representation of the affine part of the canonical representative of $\delta[\infty^+ - \infty^-]$ using base divisor $D_\infty = g\infty^-$, is given by the polynomials in the basis of $\mathfrak{a}$ (see Remark 3 for the non-uniqueness of the basis of $\mathfrak{a}$).*

Remark 5. Let g be the genus of the curve C, and let $d = \lceil \frac{g}{2} \rceil$. In [8][Section 3.3], it is observed that if one composes two ideals with distances $\delta_1 - d$ and $\delta_2 - d$, and applies a reduction chain to the result, with high probability the first reduced ideal obtained will be the ideal with distance $\delta_1 + \delta_2 - d$. This can be explained simply by observing that changing the ideal with distance δ for the ideal with distance $\delta - d$ can be interpreted as changing the base divisor in Corollary 2 from $D_\infty = g\infty^-$ to $D_\infty = g/2(\infty^+ + \infty^-)$ if the genus is even (or to $D_\infty = (g+1)/2\infty^+ + (g-1)/2\infty^-$ if the genus is odd) while staying in a fixed divisor class. The fact that no extra operations are needed when the base divisor is balanced is indeed one of the main observations from [4].

We have proved that there is a simple map ψ sending the infrastructure ideals into the class group $\mathrm{Cl}^0(C)$ that is compatible with the *group-like* structure of $\mathcal{R}$. Using the explicit description of ψ, we can describe exactly the elements missing in $\mathcal{R}$ to be a group. Let $\mathrm{div}((u,v),n)$ denote the element

$$[\mathrm{div}(u,v) + n\infty^+ + (g - \deg(u) - n)\infty^- - g\infty^-],$$

and let $\mathbf{G} = \langle[\infty^+ - \infty^-]\rangle$ be the subgroup of $\mathrm{Cl}^0(C)$ generated by $[\infty^+ - \infty^-]$. Using this notation we have the following

Theorem 2. *The image $\psi(\mathcal{R})$ of the infrastructure ideals under ψ, consists of the elements of $\mathbf{G}$ of the form* $\mathrm{div}((u,v),0)$.

Proof. A different way to state this theorem is by saying that a divisor class $[D]$ in $\mathbf{G}$ is in the image of ψ if and only if the coefficient of ∞^+ in its canonical representative with base divisor $g\infty^-$ is zero.

By construction, all ideals in the image $\psi(\mathcal{R})$ have the indicated form. We will prove the converse by induction, and we will only prove it for positive multiples of $[\infty^+ - \infty^-]$, as the proof for negative multiples is either not necessary or analogous.

Let $\mathfrak{a}_0$ be an ideal with distance $\delta_0 = \delta(\mathfrak{a}_0)$. Denote $\mathfrak{a}_1, (\omega^+, \omega^-) = \mathrm{red}_\infty(\mathfrak{a}_0)$ and let $\delta_1 = \delta(\mathfrak{a}_1)$. Since $\omega^+ = \deg(\mathfrak{a}_0) - g - 1$, Corollary 1 proves that $\delta_1 - \delta_0 = g + 1 - \deg(\mathfrak{a}_0)$. We will show that none of the elements $n[\infty^+ - \infty^-]$, for $\delta_0 < n < \delta_1$, has the indicated form (if $\delta_1 - \delta_0 = 1$ this is a vacuous statement).

We know from Theorem 1 that $\psi(\mathfrak{a}_0) = \delta_0[\infty^+ - \infty^-]$, and by definition

$$\psi(\mathfrak{a}_0) = [\mathrm{div}(\mathfrak{a}_0) + (g - \deg(\mathfrak{a}_0))\infty^- - g\infty^-].$$

For every n such that $\delta_0 < n < \delta_1$ the divisor

$$\big(\mathrm{div}(\mathfrak{a}_0) + (g - \deg(\mathfrak{a}_0))\infty^- - g\infty^-\big) + (n - \delta_0)(\infty^+ - \infty^-),$$

gives a representative of the divisor class $n[\infty^+ - \infty^-]$. This divisor can be rewritten as

$$\mathrm{div}(\mathfrak{a}_0) + (n - \delta_0)\infty^+ + (g - \deg(\mathfrak{a}_0) - n + \delta_0)\infty^- - g\infty^-. \tag{6}$$

Since $\delta_0 < n < \delta_1$ and $\delta_1 - \delta_0 = g + 1 - \deg(\mathfrak{a}_0)$, we have $n - \delta_0 > 0$, and $\deg(\mathfrak{a}_0) - g - n + \delta_0 \geq 0$. It follows that the divisor given by Equation (6) is the canonical representative of $n[\infty^+ - \infty^-]$ in $\mathrm{Cl}^0(C)$ with base divisor $g\infty^-$. But the coefficient of ∞^+ in this divisor is not zero, hence it does not have the form $\mathrm{div}((u,v),0)$ and the result follows.

Corollary 3. *Let $\mathfrak{a}_1, \mathfrak{a}_2$ and $\mathfrak{a}_3$ be infrastructure ideals with*

$$\delta(\mathfrak{a}_1) + \delta(\mathfrak{a}_2) = \delta(\mathfrak{a}_3),$$

then the operations needed to calculate $\mathfrak{a}_3$ from $\mathfrak{a}_1$ and $\mathfrak{a}_2$ are the same as the operations needed to add $\delta(\mathfrak{a}_1)[\infty^+ - \infty^-]$ and $\delta(\mathfrak{a}_2)[\infty^+ - \infty^-]$ in $\mathrm{Cl}^0(C)$, when these two ideal classes are given by their canonical representatives with base divisor $D_\infty = g\infty^-$.

Remark 6. Theorem 1 and Corollary 3 show that the use of infrastructure in cryptographic protocols is equivalent to the implementation of these protocols in the class group $\mathrm{Cl}^0(C)$ of the corresponding hyperelliptic curve C, with the disadvantage that the infrastructure has some 'holes', as proven in Theorem 2, while $\mathrm{Cl}^0(C)$ is a group. Corollary 3 also shows that the representation of the elements of $\mathrm{Cl}^0(C)$ used when working with the infrastructure is non-optimal, and it would be better to work with the representation using a balanced divisor at infinity as described in the article [4].

It is possible to use our results to properly assess the difficulty of computing the distance of a random infrastructure ideal $\mathfrak{a}$. The only arguments known in this direction show that the problem of finding distances in the set of infrastructure ideals associated to a real model for an elliptic curve E is equivalent to the DLP in E; it is then argued that if an algorithm existed to compute distances in the infrastructure of *all* hyperelliptic curves, then this algorithm could be used to solve the DLP in an elliptic curve. This argument is not satisfactory, and we now present a more refined analysis.

Theorem 3. *Let $\mathcal{R}$ be the set of infrastructure ideals associated to the hyperelliptic curve C given by a real model. The problem of computing the distance $\delta(\mathfrak{a})$ of a random ideal $\mathfrak{a}$ in $\mathcal{R}$ is equivalent to the DLP in the subgroup $\mathbf{G} = \langle[\infty^+ - \infty^-]\rangle$ of the class group $\mathrm{Cl}^0(C)$.*

Proof. Let $\mathfrak{a}$ be an ideal in $\mathcal{R}$ with distance $\delta = \delta(\mathfrak{a})$. Theorem 1 shows that $\psi(\mathfrak{a}) = \delta[\infty^+ - \infty^-]$. The element $\psi(\mathfrak{a})$ can be computed in polynomial time, and the problem of finding δ is thus reduced to finding the discrete logarithm of $\psi(\mathfrak{a})$ with respect to $[\infty^+ - \infty^-]$.

To prove the reverse implication, note that Theorem 2 shows that a random element of $\mathbf{G}$ will belong to the image of $\mathcal{R}$ under ψ with high probability. Hence, given two divisor classes $[D_1]$ and $[D_2]$, one can find an integer n, relatively prime to the regulator R, such that $n[D_1]$ and $n[D_2]$ lie in $\psi(\mathcal{R})$ in probabilistic polynomial time. The map ψ can be inverted in constant time, and if δ_1 and δ_2 are the distances of the ideals $\psi^{-1}(n[D_1])$ and $\psi^{-1}(n[D_1])$, then the discrete logarithm of $[D_2]$ with respect to $[D_1]$ is given by $\delta_2/\delta_1 \mod R$.

8 Conclusions

The main computational applications of infrastructure in the arithmetic of real quadratic number fields are the computation of the regulator and of a fundamental unit. In the case of the infrastructure of a hyperelliptic curve given by a real model over a finite field, there exist efficient algorithms to solve both of these problems.

Calculating the regulator of a hyperelliptic curve C over a finite field $\mathbf{F}_q$ can be done by finding the number of points on the class group $\mathrm{Cl}^0(C)$ of C. The best techniques available to count the number of points on the class group of a hyperelliptic curve do not use the infrastructure of the curve, but rather sofisticated algorithms depending on the genus of the curve and the size of the base field.

If the regulator of the hyperelliptic curve C is known, the problem of finding a fundamental unit in the affine coordinate ring of a hyperelliptic curve C can be solved in polynomial time using Miller's algorithm. Hence, the only computational task depending on the infrastructure of the curve C would be the Diffie-Hellman-like key exchange algorithm (Algorithm 6).

It has been claimed that Algorithm 6 provides the unique Diffie-Hellman-like key exchange protocol implemented over a non-group algebraic structure. In this article we have shown that there is a simple (and very natural) embedding of the infrastructure ideals into the class group of the curve that makes the group operations in $\mathrm{Cl}^0(C)$ compatible with those of the infrastructure. We have shown that every algorithm using the infrastructure to obtain cryptographic primitives can be implemented more efficiently in the class group of the corresponding hyperelliptic curve C. This is not only because the class group of the curve fills the 'holes' that prevent $\mathcal{R}$ from being a group, but also because the representation of the elements of $\mathrm{Cl}^0(C)$ used when working with the infrastructure is not optimal.

Acknowledgements

The author wishes to thank Steven D. Galbraith for his help during the elaboration of this article. The author has been supported in part by a scholarship from the Mexican Council of Science and Technology CONACYT and the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT. The information in this document reflects only the author's views, is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

References

1. Buchmann, J., and Williams, H. C. A key exchange system based on real quadratic fields. In *CRYPTO* (1989), G. Brassard, Ed., vol. 435 of *Lecture Notes in Computer Science*, Springer, pp. 335–343.
2. Cantor, D. G. Computing in the Jacobian of a hyperelliptic curve. *Math. Comp. 48*, 177 (1987), 95–101.
3. Cohen, H. *A course in computational algebraic number theory*, vol. 138 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, 1993.
4. Galbraith, S. D., Harrison, M., and Mireles Morales, D. J. Efficient hyperelliptic arithmetic using balanced representation for divisors. *ANTS 2008 procceedings* (2008). to appear.
5. Hartshorne, R. *Algebraic geometry*. Springer-Verlag, New York, 1977. Graduate Texts in Mathematics, No. 52.
6. Hindry, M., and Silverman, J. H. *Diophantine geometry*, vol. 201 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2000. An introduction.

7. Jacobson, M., Scheidler, R., and Stein, A. Fast arithmetic on hyperelliptic curves via continued fraction expansions. In *Advances in Coding Theory and Cryptography* (2007), T. Shaska, W. Huffman, D. Joyner, and V. Ustimenko, Eds., vol. 3 of *Series on Coding Theory and Cryptology*, World Scientific Publishing, pp. 201–244.
8. Jacobson, M. J., Scheidler, R., and Stein, A. Cryptographic protocols on real hyperelliptic curves. *Adv. Math. Commun. 1*, 2 (2007), 197–221.
9. Lenstra, Jr., H. W. On the calculation of regulators and class numbers of quadratic fields. In *Number theory days, 1980 (Exeter, 1980)*, vol. 56 of *London Math. Soc. Lecture Note Ser.* Cambridge Univ. Press, Cambridge, 1982, pp. 123–150.
10. Paulus, S., and Rück, H.-G. Real and imaginary quadratic representations of hyperelliptic function fields. *Math. Comp. 68*, 227 (1999), 1233–1241.
11. Scheidler, R. Cryptography in quadratic function fields. *Des. Codes Cryptography 22*, 3 (2001), 239–264.
12. Scheidler, R., Stein, A., and Williams, H. C. Key-exchange in real quadratic congruence function fields. *Des. Codes Cryptography 7*, 1-2 (1996), 153–174.
13. Schoof, R. Computing arakelov class groups. *MSRI Publications 44* (2008), 447495.
14. Shanks, D. The infrastructure of a real quadratic field and its applications. In *Proceedings of the Number Theory Conference (Univ. Colorado, Boulder, Colo., 1972)* (Boulder, Colo., 1972), Univ. Colorado, pp. 217–224.