Fast genus 2 arithmetic based on Theta functions

P. Gaudry

July 5, 2005

Abstract

In 1986, D. V. Chudnovsky and G. V. Chudnovsky proposed to use formulae coming from Theta functions for the arithmetic in Jacobians of genus 2 curves. We follow this idea and derive fast formulae for the scalar multiplication in the Kummer surface associated to a genus 2 curve, using a Montgomery ladder. Our formulae can be used to design very efficient genus 2 cryptosystems that should be faster than elliptic curve cryptosystems in some hardware configurations.

1 Introduction

In 1986 D. V. Chudnovsky and G. V. Chudnovsky [4] published an article containing many formulae for computing in an elliptic curve, with a view towards primality proving and factorization. At the very end of the paper, they mention that it might be interesting to look at genus 2 curves and they give some formulae for the duplication on the associated Kummer surface. This is the starting point of the work we present here. Our goal is to fill in all the details of their approach, in order to obtain useful formulae for cryptographic applications.

The first important point is that we do not work in the Jacobian of a genus 2 curve but in the Kummer surface associated to it. The Kummer surface is a variety obtained by grouping together two opposite points of the Jacobian of a genus 2 curve. More precisely, there is a map $\mathbf{J}(\mathcal{C}) \to \mathcal{K}$ such that each point of $\mathcal{K}$ has two preimages and these preimages are opposites elements of $\mathbf{J}(\mathcal{C})$. There are 16 exceptions that correspond to the 16 two-torsion points. Indeed a 2-torsion point is its own opposite by definition.

Hence the Kummer surface is the hyperelliptic equivalent of working only with the abscissa of the points of an elliptic curve. Some information is lost, but on the other hand the arithmetic can be sped-up. The Kummer surface does not naturally comes with a group structure. However the group law on the Jacobian endows a pseudo-group structure on the Kummer surface that is sufficient to define a scalar multiplication. This is done classically with a so-called Montgomery ladder (the classical binary method for the evaluation of Lucas sequences). For elliptic curves, the Montgomery ladder with abscissa only representation is widely used. A first generalization to Kummer surfaces of genus 2 curves was given [18] in 1999, and recently there has been some more work in that direction [5, 12].

In our case, following Chudnovsky and Chudnovsky, we use formulae for the arithmetic in the Kummer surface that comes from the theory of Theta functions. There are some advantages and some drawbacks of this approach. The main advantage is that the formulae are compact, elegant and very efficient compared to the formulae that are derived from Cantor's algorithm [2] or from the bilinear maps of [3]. The main drawback is that the formulae with Theta functions are valid a priori only over the complex numbers. Some work is required to apply them over a finite field and there is some rationality issue; in particular the two-torsion points must all be defined over the base field.

In the end, we obtain formulae that yield very competitive genus 2 cryptosystems. The operation count implies that this should beat elliptic curve cryptosystems in many contexts and in particular in cases where a Montgomery ladder has to be used. In constraint environments where doubling the size of the base field costs a big penalty, we expect our formulae to give a speedup of at least of factor of 2.

The paper is organized as follows: in Section 2 we recall the definitions of Theta functions and of Theta constants in genus 2. In Section 3, we define the Kummer surface and give the fromulae for the pseudo-group law that allow a fast scalar multiplication. In Section 4, we give some explicit algebraic formulae that relate a Kummer surface and the underlying genus 2 curve. We also give an explicit map from the Kummer surface to the Jacobian of that curve. In Section 5 we apply these results to cryptography; this includes some considerations about the validity of our formulae over finite fields. All the formulae that we use are known or are direct consequences of classical formulae. For the convenience of the reader we group some of them in an Appendix where precise references are given for proofs.

Acknowledgments

I am especially grateful to Tanja Lange whose enthusiasm was a big motivation for me to finish this work. She also made very interesting comments on draft versions. I thank Régis Dupont for sharing with me his knowledge about Theta constants.

2 Theta functions and theta constants in genus 2

We adopt the notations of Mumford's books [15, 16]. Let Ω be a matrix in the 2-dimensional Siegel upper-half-space $\mathcal{H}_2$, the set of symmetric 2×2 complex matrices with positive definite imaginary part. The *Riemann Theta function* is a function associated to Ω from $\mathbb{C}^2$ to $\mathbb{C}$, defined for $\mathbf{z}\in\mathbb{C}^2$ by

$$\vartheta(\mathbf{z},\Omega) = \sum_{n\in\mathbb{Z}^2} \exp\left(\pi i\, {}^t n\Omega n + 2\pi i\, {}^t n\cdot \mathbf{z}\right).$$

Remember that n and $\mathbf{z}$ are 2-dimensional (column) vectors and that the products involved in the formula are matrix products. The fact that the imaginary part of Ω is positive makes the series convergent, and even absolutely convergent over any compact, so that the function is analytic [15, page 118].

Of great interest are also the *Theta functions with characteristics*: up to a simple exponential factor, they are translates of ϑ. Let a and b be two vectors in $\mathbb{Q}^2$, we define

$$\vartheta[a;b](\mathbf{z},\Omega) = \exp\left(\pi i\, {}^t a\Omega a + 2\pi i\, {}^t a\cdot(\mathbf{z}+b)\right)\cdot\vartheta(\mathbf{z}+\Omega a + b,\Omega).$$

A scalar obtained by evaluating a Theta function with characteristic at $\mathbf{z}=(0,0)$ is called a *Theta constant.*

In the following, we shall concentrate on the characteristics $[a;b]$ where a and b are vectors whose entries are in $\{0,\frac{1}{2}\}$. There are 16 of them, yielding 16 Theta functions with characteristics and 16 Theta constants. Among them, 10 are even and 6 are odd; indeed, according to [15, page 167], we have

$$\vartheta[a;b](-\mathbf{z},\Omega) = (-1)^{4\,{}^t a\cdot b}\vartheta[a;b](\mathbf{z},\Omega).$$

Obviously, the 6 odd Theta functions with characteristics give trivial Theta constants. For a fixed Ω in $\mathcal{H}_2$, we give shorter names to the 16 Theta functions with characteristics: we denote them simply by $\vartheta_i(\mathbf{z})$, where i is in $[1,10]$ for the even functions and i is in $[11,16]$ for the odd functions. The full correspondance for the 16 characteristics is given in the Appendix. In the main part of the paper, we shall use only 4 of them that we call *fundamental Theta functions*:

$$\begin{array}{rcl} \vartheta_1(\mathbf{z}) & = & \vartheta[(0,0);(0,0)](\mathbf{z},\Omega) \\ \vartheta_2(\mathbf{z}) & = & \vartheta[(0,0);(\frac{1}{2},\frac{1}{2})](\mathbf{z},\Omega) \\ \vartheta_3(\mathbf{z}) & = & \vartheta[(0,0);(\frac{1}{2},0)](\mathbf{z},\Omega) \\ \vartheta_4(\mathbf{z}) & = & \vartheta[(0,0);(0,\frac{1}{2})](\mathbf{z},\Omega) \ . \end{array}$$

We will also need 4 other Theta functions, that are evaluated at 2Ω. We denote them with the capital letter Θ:

$$\begin{array}{rcl} \Theta_1(\mathbf{z}) & = & \vartheta[(0,0);(0,0)](\mathbf{z},2\Omega) \\ \Theta_2(\mathbf{z}) & = & \vartheta[(\frac{1}{2},\frac{1}{2});(0,0)](\mathbf{z},2\Omega) \\ \Theta_3(\mathbf{z}) & = & \vartheta[(0,\frac{1}{2});(0,0)](\mathbf{z},2\Omega) \\ \Theta_4(\mathbf{z}) & = & \vartheta[(\frac{1}{2},0);(0,0)](\mathbf{z},2\Omega) \ . \end{array}$$

Since the functions ϑ_i are associated to the matrix Ω and the functions Θ_i are associated to the matrix 2Ω, there are two abelian varieties involved, that are $(2,2)$-isogenous.

3 Pseudo-group law on a Kummer surface

3.1 Definition and equation of the Kummer surface

Definition 1 *Let Ω in $\mathcal{H}_2$. The Kummer surface associated to Ω is the locus of the images by the map φ from $\mathbb{C}^2$ to $\mathbb{P}^3(\mathbb{C})$ defined by*

$$\varphi \ : \ \mathbf{z} \mapsto (\vartheta_1(2\mathbf{z}), \vartheta_2(2\mathbf{z}), \vartheta_3(2\mathbf{z}), \vartheta_4(2\mathbf{z}))\ .$$

It can be proven that this map is well defined in the sense that the four ϑ_i cannot vanish simultaneously. Furthermore, the Theta functions verify the following periodicity condition: for all $\mathbf{z}$ in $\mathbb{C}^2$, for all b in $\{0,\frac{1}{2}\}^2$, and for all (m,n) in $\mathbb{Z}^2\times\mathbb{Z}^2$, we have (see [15, page 123])

$$\vartheta[0;b](\mathbf{z}+\Omega m+n) = \exp(-2i\pi {}^t b\cdot m - i\pi {}^t m\Omega m - 2i\pi {}^t m\cdot \mathbf{z})\cdot \vartheta[0;b](\mathbf{z}).$$

Therefore two vectors that differ by an element of the lattice $\mathbb{Z}^2+\Omega\mathbb{Z}^2$ are mapped to the same point by φ. This map is then to be seen as a map from the abelian variety $\mathbb{C}^2/(\mathbb{Z}^2+\Omega\mathbb{Z}^2)$.

An additional result is that the Kummer surface of Ω is a projective variety of dimension 2 (hence the denomination "surface"), that we denote by $\mathcal{K}(\Omega)$ or simply $\mathcal{K}$ if there is no ambiguity.

All these results have a theoretical funding that is described for instance in [11, Chapter 10]. However, the formulae that we use in that paper give an explicit version of all the results that we need. For instance, the fact that all the even functions can be deduced from $(\vartheta_1(\mathbf{z}),\vartheta_2(\mathbf{z}),\vartheta_3(\mathbf{z}),\vartheta_4(\mathbf{z}))$ allows us to find explicit maps between points of the Kummer surface and the Jacobian of an associated hyperelliptic curve.

The group law on the abelian variety $\mathbb{C}^2/(\mathbb{Z}^2+\Omega\mathbb{Z}^2)$ does not transport completely into a group law on $\mathcal{K}$. Indeed, since all the ϑ_i are even, φ is even and maps two opposite element to the same point in $\mathcal{K}$. However, we shall see that this is essentially the only obstruction for the map φ to be made a group homomorphism.

Assume that we have precomputed the Theta constants $\vartheta_i(0)$ and $\Theta_i(0)$, and that we are given the four coordinates of $\varphi(\mathbf{z})$ for some $\mathbf{z}$ in $\mathbb{C}^2$, but $\mathbf{z}$ is unknown to us. Then it possible to compute the four coordinates of $\varphi(2\mathbf{z})$ using the duplication formulae given below. Hence doubling on $\mathcal{K}$ is well defined.

On the other hand, if we are given $\varphi(\mathbf{z})$ and $\varphi(\mathbf{z}')$ in $\mathcal{K}$ for two different unknown vectors $\mathbf{z}$ and $\mathbf{z}'$, it is not possible to decide whether the first point comes from $\mathbf{z}$ or from $-\mathbf{z}$ and similarly whether the second point comes from $\mathbf{z}'$ or from $-\mathbf{z}'$. Hence the sum in $\mathcal{K}$ is ill-defined since we can not choose between computing $\varphi(\mathbf{z}+\mathbf{z}')$ and $\varphi(\mathbf{z}-\mathbf{z}')$, which are in general different. Still if one of $\varphi(\mathbf{z}+\mathbf{z}')$ and $\varphi(\mathbf{z}-\mathbf{z}')$ is known, the other can be deduced, again with some analytic addition formulae. We will come back to this later in this section.

In the end, our goal is to work algebraically and not analytically, therefore it is first necessary to obtain an algebraic equation for $\mathcal{K}$. This is not obvious that such an equation exists, since $\vartheta(\mathbf{z})$ is a transcendental function. Still, this is a very classical result already known in 19-th century.

We shall consider a Kummer surface $\mathcal{K} = \mathcal{K}_{a,b,c,d}$ parametrized by the Theta constants:

$$a = \vartheta_1(0),\ b = \vartheta_2(0),\ c = \vartheta_3(0),\ d = \vartheta_4(0),$$

and

$$A = \Theta_1(0),\ B = \Theta_2(0),\ C = \Theta_3(0),\ D = \Theta_4(0).$$

Their squares are linked by simple linear relations that are obtained by putting $\mathbf{z} = 0$ in Equations (2) and (3) in Section 7.2 of the Appendix:

$$\begin{aligned} 4A^2 &= a^2 + b^2 + c^2 + d^2, \\ 4B^2 &= a^2 + b^2 - c^2 - d^2, \\ 4C^2 &= a^2 - b^2 + c^2 - d^2, \\ 4D^2 &= a^2 - b^2 - c^2 + d^2. \end{aligned} \tag{1}$$

We write (x, y, z, t) the projective coordinates of a point on $\mathcal{K}$, that is

$$x = \lambda\vartheta_1(\mathbf{z}),\ y = \lambda\vartheta_2(\mathbf{z}),\ z = \lambda\vartheta_3(\mathbf{z}),\ d = \lambda\vartheta_4(\mathbf{z}),$$

for some $\mathbf{z}$ in $\mathbb{C}^2$ and some λ in $\mathbb{C}^*$. Then a projective equation of $\mathcal{K}$ can be derived by combining a few Frobenius identities [16, Section 7, Chapter IIIa]. We obtain

$$(x^4 + y^4 + z^4 + t^4) + 2Exyzt - F(x^2t^2 + y^2z^2) - G(x^2z^2 + y^2t^2) - H(x^2y^2 + z^2t^2) = 0,$$

where

$$\begin{aligned} E &= abcdA^2B^2C^2D^2/(a^2d^2 - b^2c^2)(a^2c^2 - b^2d^2)(a^2b^2 - c^2d^2) \\ F &= (a^4 - b^4 - c^4 + d^4)/(a^2d^2 - b^2c^2) \\ G &= (a^4 - b^4 + c^4 - d^4)/(a^2c^2 - b^2d^2) \\ H &= (a^4 + b^4 - c^4 - d^4)/(a^2b^2 - c^2d^2)\ . \end{aligned}$$

Note that since A^2, B^2, C^2, D^2 can be deduced linearly from a^2, b^2, c^2, d^2, the equation of $\mathcal{K}$ is fixed, once given a, b, c, d.

Hence the constants E, F, G, H can be precomputed for each new curve we want to work with.

Remark 1 The equation for $\mathcal{K}$ involves some denominators that could vanish. In fact, these denominators are even Theta constants (for instance $a^2d^2 - b^2c^2$ equals $\vartheta_5(0)^2\vartheta_6(0)^2$, see the Appendix). The values of Ω for which one of the even Theta constants vanishes are exceptionnal, as shown in [10, Chapter 9, Proposition 2]. This leads us to the following genericity condition that we will assume to be true in the sequel.

Genericity Condition 1 *Four scalars a, b, c, d verify the Genericity Condition 1 if they are non zero and the 6 other Theta constants that can be deduced if $(a, b, c, d) = (\vartheta_i(0))_{i=1,2,3,4}$, are also non zero.*

Remark 2 As the notation $\mathcal{K}_{a,b,c,d}$ and the Genericity Condition 1 suggest, our input will often be the scalars a, b, c, d instead of a matrix Ω in $\mathcal{H}_2$. This raises the question of the existence of such a Ω for which $(a, b, c, d) = (\vartheta_i(0))_{i=1,2,3,4}$. In general this is false, if we ask those two 4-uples to be equal. However, if we ask just those two 4-uples to represent the same point in the projective space $\mathbb{P}^3$, then this is true in general. Up to now, all the formulae are homogenous, so that the projective equality is enough. This is also true for the sequel. Therefore there is no problem in starting from the scalars a, b, c, d, for what we aim at.

3.2 Pseudo-group formulae

In the duplication formulae of Section 7.2 of the Appendix that we want to use now, we see that there are some problems if the Theta constants vanish, but also if A, B, C, D vanish. Therefore we strengthen our genericity assumptions.

Genericity Condition 2 *Four scalars a, b, c, d verify the Genericity Condition 2 if they verify the Genericity Condition 1 and furthermore A, B, C, D that are deduced from them are not zero.*

Let $\mathcal{K} = \mathcal{K}_{a,b,c,d}$ be a Kummer surface that verifies the Genericity Condition 2. We fix the following constants, that can be precomputed:

$$y_0 = a/b,\ z_0 = a/c,\ t_0 = a/d,$$

and

$$y_0' = (A/B)^2,\ z_0' = (A/C)^2,\ t_0' = (A/D)^2.$$

Doubling Algorithm: `DoubleKummer(P)`
Input: A point $P = (x, y, z, t)$ on $\mathcal{K}$;
Output: The double $2P = (X, Y, Z, T)$ in $\mathcal{K}$.

1. $x' = (x^2 + y^2 + z^2 + t^2)^2$;
2. $y' = y_0'(x^2 + y^2 - z^2 - t^2)^2$;
3. $z' = z_0'(x^2 - y^2 + z^2 - t^2)^2$;
4. $t' = t_0'(x^2 - y^2 - z^2 + t^2)^2$;
5. $X = (x' + y' + z' + t')$;

6. $Y = y_0(x' + y' - z' - t')$;
7. $Z = z_0(x' - y' + z' - t')$;
8. $T = t_0(x' - y' - z' + t')$;
9. Return (X, Y, Z, T).

The cost of the doubling algorithm is 8 squarings and 6 products by fixed constants. In Chudnovsky's paper [4, page 430], the formulae are slightly different. They correspond to taking $x = \vartheta_1(\mathbf{z})^2$, $y = \vartheta_2(\mathbf{z})^2$, etc.

Pseudo-addition Algorithm: `PseudoAddKummer(P, Q, R)`
Input: Two points $P = (x, y, z, t)$ and $Q = (\underline{x}, \underline{y}, \underline{z}, \underline{t})$ on $\mathcal{K}$ and $R = (\bar{x}, \bar{y}, \bar{z}, \bar{t})$ one of $P + Q$ and $P - Q$, with $\bar{x}\bar{y}\bar{z}\bar{t} \neq 0$.
Output: The point (X, Y, Z, T) in $\mathcal{K}$ among $P + Q$ and $P - Q$ which is different from R.

1. $x' = (x^2 + y^2 + z^2 + t^2)(\underline{x}^2 + \underline{y}^2 + \underline{z}^2 + \underline{t}^2)$;
2. $y' = y_0'(x^2 + y^2 - z^2 - t^2)(\underline{x}^2 + \underline{y}^2 - \underline{z}^2 - \underline{t}^2)$;
3. $z' = z_0'(x^2 - y^2 + z^2 - t^2)(\underline{x}^2 - \underline{y}^2 + \underline{z}^2 - \underline{t}^2)$;
4. $t' = t_0'(x^2 - y^2 - z^2 + t^2)(\underline{x}^2 - \underline{y}^2 - \underline{z}^2 + \underline{t}^2)$;
5. $X = (x' + y' + z' + t')/\bar{x}$;
6. $Y = (x' + y' - z' - t')/\bar{y}$;
7. $Z = (x' - y' + z' - t')/\bar{z}$;
8. $T = (x' - y' - z' + t')/\bar{t}$;
9. Return (X, Y, Z, T).

The cost of the pseudo-addition algorithm is of 8 squarings, 7 products, and 4 divisions. Of course, as we are dealing with projective coordinates, these 4 divisions can be replaced by 10 products.

Remark 3 If R has a coordinate which is zero, then the Pseudo-addition Algorithm does not work. This is a major problem of this approach. Fortunately, for the scalar multiplication algorithm, the point R is always the same, so that we can check at the beginning that the computation is possible.

Remark 4 The pseudo-group law that we just described is somewhat surprising, because it heavily relies on a (2,2)-isogenous abelian variety for the computation: for the doubling, the point is pushed through isogenies back and forth, thus obtaining a multiplication by 2 map. In the classical Cantor's group law for doubling elements of the Jacobian of a hyperelliptic curve, the decomposition of the multiplication by 2 map into isogenies is not at all visible. Therefore the formulae are inherently different. The really interesting fact is that also the (pseudo-)addition makes use of this step trough an isogenous variety.

3.3 Scalar multiplication in the Kummer surface

Scalar multiplication is well-defined in the Kummer surface, since the pseudo-group law is sufficient, as we recall now. This is similar to the situation with elliptic curves, where it is possible to compute the abscissa of n times a point for which only the abscissa is known.

Let P be a known point on $\mathcal{K}$. Assume that we also know nP and $(n+1)P$ for some integer $n > 0$. Then, the difference between $(n+1)P$ and nP is known, and we can use the pseudo-addition algorithm to compute their sum $(2n+1)P$. Furthermore, by doubling nP or $(n+1)P$, we can compute $2nP$ or $(2n+2)P$. Hence, a binary powering algorithm can be designed that works with pairs of consecutive points. At each step, the choice of which point we double is made according to the binary expansion of the multiplier. This is a well known strategy, but we recall it here for completeness.

Scalar Multiplication Algorithm.
Input: A point P on $\mathcal{K}$ with no zero coordinate and an integer $n > 1$.
Output: The point nP in $\mathcal{K}$.

1. If n is 2, then return `DoubleKummer`(P).
2. Let $n_0 n_1 \cdots n_k$ be the binary writing of n, where n_0 is the most significant bit (therefore, it is a 1).
3. $P_m = P$; $P_p =$ `DoubleKummer`(P);
4. For i from 1 to k do
   (a) $Q =$ `PseudoAddKummer`(P_p, P_m, P);
   (b) if n_i equals 1 then
      i. $P_p =$ `DoubleKummer`(P_p);
      ii. $P_m = Q$;
   (c) else
      i. $P_m =$ `DoubleKummer`(P_m);
      ii. $P_p = Q$;
5. Return P_m.

In the `PseudoAddKummer` algorithm and in the `DoubleKummer` algorithm, the first steps are similiar, since we start by computing four linear combinations of the squares of the coordinates for each input. In the scalar multiplication, the input of a doubling has always been the input of a pseudo-addition before. Therefore it makes sense to share this computation.

In fact, in the `PseudoAddKummer` algorithm we break the symmetry of the formulae: we compute (and store for subsequent application of `DoubleKummer`)

$$(x^2+y^2+z^2+t^2),\quad y_0'(x^2+y^2-z^2-t^2),\quad z_0'(x^2-y^2+z^2-t^2),\quad t_0'(x^2-y^2-z^2+t^2)$$

for the point that will have to be doubled thereafter, and compute (and do not store)

$$(x^2+y^2+z^2+t^2),\quad (x^2+y^2-z^2-t^2),\quad (x^2-y^2+z^2-t^2),\quad (x^2-y^2-z^2+t^2)$$

for the other point.

Remarking also that the final divisions in the `PseudoAddKummer` algorithm involve always the same input, namely the coordinates of P, it is possible to do the precomputation so that this last steps cost only 3 multiplications (remember that we are dealing with projective coordinates). Therefore the `PseudoAddKummer` algorithm has a cost of 10 products and 8 squares.

Then the subsequent `DoubleKummer` algorithm makes use of the precomputed data, so that only 6 products and 1 square are needed.

Theorem 1 *Computing n times a point with no zero coordinate in a Kummer surface that verifies the Genericity Condition 2 amounts to $9\,|n|_2$ squarings and $16\,|n|_2$ products plus a constant number of products for the precomputations, where $|n|_2$ denotes the size of the binary expansion of n.*

This result can be refined: among the $16\,|n|_2$ products, $6\,|n|_2$ products are multiplications by constants that depend only on the surface and $3\,|n|_2$ products are multiplications by numbers that depend only on the point we want to multiply by n. Therefore by choosing an apropriate surface, a few multiplications can be saved. Also we can note that up to 8 operations can be performed simultaneously, so that the depth of the scalar multiplication circuit is $4\,|n|_2$ times the depth of the multiplication in the base field, which could be of great importance if some kind of parallelism is available.

Note that for this kind of scalar multiplication algorithm, NAF and other windowing methods are not available, but some other speed-up strategies might be available [8].

Remark 5 If P is of known odd order p and if P has a coordinate that is zero. Then one can double P until the point $Q = 2^k \cdot P$ is found to have no zero coordinate. Then $n \cdot P$ is equal to $(n/2^k \bmod p) \cdot Q$, and this latter expression can be computed using Theorem 1. If n is about the same size of p and if we are lucky enough so that k is small, then the additional cost is negligible.

3.4 Nodes of the Kummer surface

The 16 two-torsion points on the complex torus $\mathbb{C}^2/(\mathbb{Z}^2 + \Omega\mathbb{Z}^2)$ are sent to 16 points on the Kummer surface $\mathcal{K}(\Omega)$. These points are the only ones that have only one pre-image in $\mathbb{C}^2/(\mathbb{Z}^2+\Omega\mathbb{Z}^2)$ and in fact they are nodes of $\mathcal{K}$. We refer the reader interested in the geometry of this object to [3] and to the references therein.

With our equation for $\mathcal{K} = \mathcal{K}_{a,b,c,d}$, the 16 nodes are

$$\begin{array}{l}(a,b,c,d),(a,b,-c,-d),(a,-b,c,-d),(a,-b,-c,d),\\(b,a,d,c),(b,a,-d,-c),(b,-a,d,-c),(b,-a,-d,c),\\(c,d,a,b),(c,d,-a,-b),(c,-d,a,-b),(c,-d,-a,b),\\(d,c,b,a),(d,c,-b,-a),(d,-c,b,-a),(d,-c,-b,a)\end{array}$$

and (a,b,c,d) is the particular node that is the image of 0, that is the neutral for the pseudo-group law.

4 Link with a genus 2 curve

In this section, we will give formulae that relate the Kummer surface and the underlying genus 2 hyperelliptic curve. For the cryptographic application we will develop in the next section, this link achieves three goals:

- Having an equation for the curve allows the use of the classical point counting algorithms.
- The ability to map the point back and forth between the Kummer surface and the Jacobian gives a security reduction.
- If a cryptographical protocol requires an addition and not only a scalar multiplication, this map makes it possible (see [17]).

4.1 Several choices

Let $\mathcal{K}_{a,b,c,d}$ be a Kummer surface that verifies the Genericity Condition 2. The first task is to compute the squares of the six other even Theta constants. This is done through the formulae of Section 7.3 in the Appendix. However, there are eight different choices that can be made (there are some square roots to take and some symmetries). This is not a big issue, because different choices give different equations for the underlying curve, but all these equations represent genus 2 curves that are isomorphic. The important point is to keep the same choice all along the computation.

For the equation of the curve, we will see that apart from a, b, c, d, we need only to know the square of the quotient of two additional Theta constants, and that there are only two choices for that ratio.

However, the explicit map we give from $\mathcal{K}$ to the Jacobian of the curve involves all the Theta constants, and therefore we assume that a consistent choice is made. It is worth to be noted that the formulae for that map could be rewritten so that they also require only the knowledge of the same ratio than for the equation of the curve. Therefore, in principle the choice to be made is only among two and not among 8 possibilities.

Remark 6 The work of van Wamelen [19] shows that we can use a different map φ from $\mathbb{C}^2$ to $\mathbb{P}^3(\mathbb{C})$ that makes the Kummer surface and the curve in Rosenhain form defined over the same base field. This map is slightly more complicated, so we prefer to stick to the classical one, and have some square roots and symmetries to solve.

4.2 Equation of the underlying curve

Let $\mathcal{C}$ be a curve of genus 2 given by an equation $y^2 = f(x)$ over $\mathbb{C}$, with f a squarefree polynomial of degree 5 or 6. In 19-th century, Thomae found explicit formulae relating roots of f and some Theta constants of the Jacobian of $\mathcal{C}$ with period matrix Ω. The generalization of these formulae that are of interest for us can be found in [19]. In order to simplify, we assume that the curve is in Rosenhain form:

$$y^2 = x(x-1)(x-\lambda)(x-\mu)(x-\nu).$$

There are several choices for relating λ, μ, ν to Theta constants, that depend on the choice we make for ordering the roots of f. We present here one of the possibilities, for which more details are given in Appendix.

We shall need the two following additional Theta functions with characteristics:

$$\vartheta_8(z) = \vartheta[(\frac{1}{2}, \frac{1}{2}); (0, 0)](z, \Omega)$$
$$\vartheta_{10}(z) = \vartheta[(\frac{1}{2}, \frac{1}{2}); (\frac{1}{2}, \frac{1}{2})](z, \Omega) \ ,$$

and the associated Theta constants are denoted by

$$e = \vartheta_8(0), \ \ f = \vartheta_{10}(0).$$

Then we have:

$$\lambda = \frac{a^2c^2}{b^2d^2}; \ \mu = \frac{c^2e^2}{d^2f^2}; \ \nu = \frac{a^2e^2}{b^2f^2}.$$

Using the formulae of Section 7.3, we can compute $\vartheta_8(0)$ and $\vartheta_{10}(0)$ (up to a few choices) from a, b, c, d. In fact these formulae simplify into

$$\frac{e^2}{f^2} = \frac{1 + \frac{CD}{AB}}{1 - \frac{CD}{AB}}.$$

Note that since from a, b, c, d we can compute only the squares of A, B, C, D, using Equations 1, the fraction CD/AB is defined up to sign. Changing the sign, means replacing e^2/f^2 by f^2/e^2. As said above, this transformation corresponds to equations of two isomorphic curves, so we can choose one or the other as long as we are consistent with this choice all along the computation.

4.3 Mapping point of $\mathcal{K}$ into the Jacobian of $\mathcal{C}$

Once we have an equation for the curve $\mathcal{C}$ associated to $\mathcal{K}$, a natural question is to give an explicit function that maps the points of $\mathcal{K}$ to class of divisors in the Jacobian of $\mathcal{C}$, for instance in their Mumford representation.

Again, although all of this is based on transcendental functions, in the end the map can be made algebraic, and even rational, up to the inherent obstruction that tells that we can not decide between a divisor and its opposite.

The formulae are obtained from Theorem IIIa.7.6 in [16], and the extension given by van Wamelen [19], Theorem 7.

We shall need additional genericity conditions.

Genericity Condition 3 *Four scalars a, b, c, d verify the Genericity Condition 3 if they verify the Genericity Conditions 1 and 2 and furthermore $\vartheta_7(0)^4 \neq \vartheta_9(0)^4$, $\vartheta_5(0)^4 \neq \vartheta_6(0)^4$, and $\vartheta_8(0)^4 \neq \vartheta_{10}(0)^4$.*

Let $\mathcal{K}_{a,b,c,d}$ be a Kummer surface that verifies these Genericity Conditions and assume that we have computed all the squares of Theta constants (making a choice for the square roots and symmetries involved). Let $P = (x, y, z, t)$ be a point on $\mathcal{K}_{a,b,c,d}$, that is not a node. Then using the formulae of Section 7.4 of the Appendix, it is possible to compute $\vartheta_i(\mathbf{z})^2$ for all $i \in [5, 16]$, corresponding to $(x, y, z, t) = (\vartheta_i(\mathbf{z}))_{i=1,2,3,4}$.

Then, let us define

$$u_0 = \lambda \frac{\vartheta_8(0)^2 \vartheta_{14}(\mathbf{z})^2}{\vartheta_{10}(0)^2 \vartheta_{16}(\mathbf{z})^2},$$

and

$$u_1 = (\lambda - 1)\frac{\vartheta_5(0)^2\vartheta_{13}(\mathbf{z})^2}{\vartheta_{10}(0)^2\vartheta_{16}(\mathbf{z})^2} - u_0 - 1.$$

Then the polynomial $u(x) = x^2 + u_1 x + u_0$ is the first polynomial in the Mumford representation of the divisor D_P corresponding to P. For a given u-polynomial, there are up to four v-polynomials that yield a valid Mumford representation of a divisor in the Jacobian of $\mathcal{C}$. These four choices are grouped into two pairs of opposite divisors. Since the Jacobian is a degree 2 cover of the Kummer surface, one should be able to decide which pair of opposite divisors is the real image of the point P. Generically, giving the square of the constant term of $v(x) = v_1 x + v_0$ is enough to decide. Here is the formula for v_0^2 in terms of Theta functions:

$$\begin{aligned} v_0^2 = &-\frac{\vartheta_1^4\vartheta_3^4\vartheta_8^2\vartheta_{14}(\mathbf{z})^2}{\left(\vartheta_2^2\vartheta_4^2\vartheta_{10}^2\vartheta_{16}(\mathbf{z})^2\right)^3}\Big(\vartheta_2^2\vartheta_3^2\vartheta_9^4\vartheta_7(\mathbf{z})^2\vartheta_{12}(\mathbf{z})^2 + \vartheta_1^2\vartheta_4^2\vartheta_7^4\vartheta_9(\mathbf{z})^2\vartheta_{11}(\mathbf{z})^2 \\ &+ 2\vartheta_1^2\vartheta_2^2\vartheta_3^2\vartheta_4^2\left(\vartheta_1(\mathbf{z})^2\vartheta_3(\mathbf{z})^2 + \vartheta_2(\mathbf{z})^2\vartheta_4(\mathbf{z})^2\right) \\ &- 2\vartheta_1\vartheta_2\vartheta_3\vartheta_4\vartheta_1(\mathbf{z})\vartheta_2(\mathbf{z})\vartheta_3(\mathbf{z})\vartheta_4(\mathbf{z})\left(\vartheta_1^2\vartheta_3^2 + \vartheta_2^2\vartheta_4^2\right)\Big), \end{aligned}$$

where we have noted ϑ_i instead of $\vartheta_i(0)$ for readibility.

At first sight, this formula involves not only the squares of the Theta functions and Theta constants but also the functions and constants themselves. However, it is easily checked that this only occurs in the last line of the formula, and that using the fact that the point P lies on the Kummer surface, the equation of $\mathcal{K}_{a,b,c,d}$ allows to rewrite everything in terms of squares. We prefer to leave it in this simpler form, since in our case, for indices $1, 2, 3, 4$, the Theta functions and the Theta constants are known, and not only their squares.

Finally, a formula for v_1 in terms of v_0, u_0 and u_1 can be deduced from the fact that $u(x)$ should divide $v^2(x) - f(x)$. Therefore we have formed a map from $\mathcal{K}$ to the Jacobian of $\mathcal{C}$ that maps a point P on $\mathcal{K}$ to a divisor D_P, up to a sign choice, that corresponds to choosing between D_P and $-D_P$.

Degenerate case If no Theta constant vanishes, the map is undefined in the case where $\vartheta_{16}(\mathbf{z})$ is zero. This corresponds to the case where the image of the map is a divisor for which the u-polynomial in the Mumford representation is of degree less than 2. Then the formula is now

$$u_0 = \frac{\lambda\vartheta_8(0)^2\vartheta_{14}(\mathbf{z})^2}{(\lambda-1)\vartheta_5(0)^2\vartheta_{13}(\mathbf{z})^2 - \lambda\vartheta_8(0)^2\vartheta_{14}(\mathbf{z})^2}.$$

The Mumford representation of the divisor is then $\langle x + u_0, \pm\sqrt{f(-u_0)}\rangle$.

Inverting the map Having the formulae for a map from $\mathcal{K}_{a,b,c,d}$ to the Jacobian of $\mathcal{C}$, it is possible to derive an algorithm for mapping a divisor on the Jacobian into $\mathcal{K}_{a,b,c,d}$. Indeed, since the formulae are algebraic formulae, inverting the map reduces to solving a system of polynomial equations. This can be done using resultants.

5 Application to cryptography

Up to now, everything has been made over the field of complex numbers. For cryptographic applications, it is first required to consider finite fields.

5.1 Validity of the formulae over finite fields

Let $\mathbb{F}_q$ be a finite field of odd characteristic with q elements, and let a, b, c, d be four elements of $\mathbb{F}_q$ that verify the Genericity Condition 3. In order to be able to compute the equation of the curve corresponding to $\mathcal{K}_{a,b,c,d}$, we need to take square roots. This is not always possible over a field that is not algebraically closed.

Rationality Condition 1 *Let a, b, c, d be four elements of a field k that verify the Genericity Condition 3. They are said to verify the Rationality Condition 1 if the corresponding value $\frac{C^2D^2}{A^2B^2}$ defined by the formulae 1 is a square in k.*

From now on, we assume that a, b, c, d verify the Rationality Condition 1 over $\mathbb{F}_q$, and that a choice has been made for the corresponding square root, so that an equation is given for $\mathcal{C}$ over $\mathbb{F}_q$.

In order to apply our pseudo-group formulae to $\mathcal{K}_{a,b,c,d}$, we need to check that they make sense over $\mathbb{F}_q$. This is deduced from the following construction that is done only in theory and does not require additional computation. We want to find a curve $\mathfrak{C}$ over a number field $\mathbb{K}$, and a prime ideal $\mathfrak{p}$ in $\mathcal{O}_{\mathbb{K}}$ with residue field $\mathbb{F}_q$, such that $\mathfrak{C}$ and its Jacobian have good reduction modulo $\mathfrak{p}$ and that $\mathfrak{C}$ reduces to $\mathcal{C}$.

Then each time we want to apply a pseudo-group algorithm in $\mathcal{K}_{a,b,c,d}$ over $\mathbb{F}_q$, we consider the divisors in the Jacobian of $\mathcal{C}$ that correspond to the points we give in input to the formula. These divisors can be lifted as divisors on the Jacobian of $\mathfrak{C}$ over an algebraic extension $\mathbb{L}$ of $\mathbb{K}$. Since $\mathbb{L}$ can be embedded in $\mathbb{C}$, the formulae for the pseudo-group law make sense over $\mathbb{L}$ and then the resulting divisor can be reduced to $\mathbb{F}_q$. The curve $\mathfrak{C}$ and its Jacobian have good reduction so that the formulae for the group law commute with the reduction map, and also the formulae for the pseudo-group law.

Hence the problem is to find such a lifted curve $\mathfrak{C}$. In the case where $\mathcal{C}$ is ordinary, a natural choice is the canonical lift that also preserves the endomorphism ring. The other cases are not so important for cryptographic applications but heuristically, a random lift has good reduction, so that we can choose several lifts until one is found that has good reduction (in that case, a few computations are needed).

Remark 7 The fact that the Kummer surface can be defined over $\mathbb{F}_q$ whereas the corresponding curve can be defined only over $\mathbb{F}_{q^2}$ can look strange. This rationality condition is an obstruction analogous to the one we can observe when applying Mestre's algorithm [14] for computing an equation for a curve for which Igusa invariants are given. Indeed, a, b, c, d (or at least the squares of their ratios) can be seen as invariants, and requiring them to be rational is not enough to guarantee the equation of the curve to be rational.

5.2 Orders of point in a Kummer surface over a finite field

Let $\mathbb{F}_q$ be a finite field of odd characteristic with q elements. Let $\mathcal{K} = \mathcal{K}_{a,b,c,d}$ be a Kummer surface over $\mathbb{F}_q$ that verifies the Genericity Condition 3 and the Rationality Condition 1. We assume that the underlying curve is ordinary so that the pseudo-group law formulae are valid over $\mathbb{F}_q$.

Since we are dealing with finite objects, the set of points of $\mathcal{K}$ defined over $\mathbb{F}_q$ is finite. Hence we can define the order of a point as the smallest non-zero integer n such that n times the points gives (a, b, c, d), which plays the role of zero in $\mathcal{K}$.

A point on $\mathcal{K}$ can be pulled back to the Jacobian of $\mathcal{C}$ or to the Jacobian of the quadratic twist of $\mathcal{C}$, and those two Jacobians intersects only at the neutral and at the two-torsion points. Then the orders of elements in $\mathcal{K}$ are clearly linked to the possible orders of points in $\mathbf{J}(\mathcal{C})$ or its twist.

Lemma 1 *Let P be a point of a Kummer surface $\mathcal{K}$ defined over $\mathbb{F}_q$ with conditions as above. Let Q be a point on the Jacobian of $\mathcal{C}$ or of its twist whose image in $\mathcal{K}$ is P. Then the order of P in $\mathcal{K}$ is the order of Q.*

5.3 Kummer-based cryptosystems

The formulae for mapping points between a Kummer surface and the associated Jacobian are not so simple. In several protocols, however, only the scalar multiplication and no addition is needed. This is the case for a classical Diffie-Hellmann key-exchange or any modern authenticated or password-based or multi-party variant. This is also the case for El-Gamal signature or encryption schemes in the version where one uses hash functions to replace additions with exclusive ors (thus relying on the Random Oracle Hypothesis). For all these protocols, there is no need at all to consider the underlying curve, except during the generation of parameters, in order to do the point-counting. Thereafter, everything is done in the Kummer surface and the complicated formulae for the map to the curve do not need to be implemented.

The generation of parameters can follow these lines:

1. Choose the base field $\mathbb{F}_q$ of odd characteristic;
2. Repeat
   (a) Choose random (a, b, c, d) verifying the Genericity Condition 3 and the Rationality Condition 1.
   (b) Compute the characteristic polynomial of the curve $\mathcal{C}$ associated to $\mathcal{K}_{a,b,c,d}$.
3. Until the group order of the Jacobian of $\mathcal{C}$ or of its twist is 16 times a prime p.
4. Pick random points with no zero coordinate in $\mathcal{K}$ until one is found that has order p.

The most complicated task is the point-counting, that can be done using a Schoof-like algorithm [6] or Kedlaya's algorithm [9] if the characteristic is small enough. The complex multiplication approach [20] is also compatible with this approach.

5.4 Security

Thanks to the explicit map between a Kummer surface and the Jacobian of the associated curve, the discrete logarithm problem in Kummer surfaces is easily seen to be polynomial time equivalent to the discrete logarithm problem in the corresponding Jacobian (this equivalence is already shown in [18]).

With this equivalence, it suffices to choose q (and hence $p \approx q^2/16$) of adequate size to counter the Pollard Rho attack which is the best known method for Jacobian of genus 2 curves with no additional structure.

5.5 Efficiency

The cost of the scalar multiplication is very low compared to the state of the art in explicit formulae for odd characteristic genus 2 arithmetic, since we need only 25 multiplications per bit. For instance in [13], using classical affine coordinates, an addition requires 1 inversion, 22 multiplications and 3 squares, and a doubling requires 1 inversion, 22 multiplications and 5 squares. If one wants to avoid inversions, then the mixed coordinates approach of [13] gives 34 multiplications and 7 squares for addition and 48 multiplications and 4 squares for doubling. The Montgomery ladder approach of [5] gives $62\,|n|_2$ multiplications and $7\,|n|_2$ squares for a multiplication by n. We conclude that our formulae improve on the current best formulae for genus 2 arithmetic.

On the other hand, the size of the base field must be enlarged by 2 bits for equivalent security, since a cofactor of 16 in the group order is mandatory for the Kummer surface to be defined over $\mathbb{F}_q$. We believe that these two bits are more than compensated by the speed of the scalar multiplication.

As an additional feature, we may note that the scalar multiplication in the Kummer surface is a Montgomery ladder, so that there is a built-in resistance against side channel analysis. More hacks, like point compression can of course be added.

Comparison with elliptic curves For elliptic curves, the cost of scalar multiplication using a Montgomery ladder is 10 multiplications per bit, with a base field which is twice as large as for genus 2 curves. In the range of parameters that correspond to cryptographic applications, the arithmetic in the base field is implemented using the school-book algorithm or using Karatsuba's trick. Therefore the cost of a multiplication is expected to be multiplied by at least 3, when the size is doubled, and 3×10 is higher than 25. So we expect that with our formulae, a genus 2 cryptosystem is faster than an elliptic curve cryptosystem if a Montgomery ladder is used. We won't say more in the general case, where the ratio between the costs of inversion and multiplication should be taken into account and in the end this depends very much on a particular implementation.

5.6 An example curve

We consider the finite field $\mathbb{F}_{3^{53}} = \mathbb{F}_3[t]/(t^{53} - t^4 - t^3 - t^2 + 1)$. After a few trials, the following parameters have been found to be suitable: Let a, b, c, d be defined by

$$a = t^7,\ b = t^5,\ c = t^3,\ d = t^{432}.$$

Then the genus 2 curve associated to $\mathcal{K} = \mathcal{K}_{a,b,c,d}$ is defined over $\mathbb{F}_q$ and the characteristic polynomial of its Jacobian (or its twist) is given by

$$s_1 = 7810481952544,\ s_2 = 38366199614160282937179494.$$

Therefore its group order is $16p$ where p is the 164-bit prime

$$p = 23481888288342804239694441280911210769617546128273.$$

It is then easy to find a point on $\mathcal{K}$ that has order p and can be used as a generator for the key-exchange protocols.

This curve is not "random" in the sense that we choosed parameters a, b, c, d so that they can be easily written on paper. But to our knowledge it has no special structure except the special form of the group order of its Jacobian.

For this example, the point-counting was done using Kedlaya's algorithm within the Magma computer algebra system [1].

6 Conclusion and further work

We have presented very efficient formulae for genus 2 arithmetic, that are based on duplication and addition formulae of Theta functions. There are still some drawbacks to our method, that could be addressed in future work:

- The equation we use for the Kummer surface is defined over the same base field as the two-torsion points. There exist equations for the Kummer surface that are defined over the same base field as the curve [3], but the group law is then more expensive [5]. There might be a compromise between those two situations.
- An extension to characteristic 2 is problematic. Our formulae rely on $(2,2)$-isogenies, and these objects behave differently in characteristic 2.

Note: Since there are quite a few formulae in this paper, the risk of a typographical error is high. Such an error could be very annoying for the reader who wants to use these formulae. Therefore, as a double check, we provide a Magma source file[1] that contains the few functions that make it possible to reproduce the computations that are described here.

7 Appendix: Some formulae for Theta functions in genus 2

The formulae that we recall in this section are well known and can be found in several places, including nineteen's century work. In the literature, these formulae are sometimes given in very general form, or using notations that are not easy to relate to ours. That is the reason why we put them here for the convenience of the reader, with our notations and specialized to the genus 2 case.

A reference for the duplication formulae is [7], page 141, Corollary of Theorem 2 of Chapter IV. The other formulae that are given here are easy consequences of the Frobenius' theta formula that can be found in Section 7 of Chapter IIIa in [16].

7.1 Numbering of Theta functions

We use the following numbering for the Theta functions, where the 10 even Theta functions come first. The first 4 Theta functions (resp. Theta constants if $\mathbf{z} = 0$) are called the fundamental Theta functions (resp. fundamental Theta constants).

$$\begin{aligned}
\vartheta_1(\mathbf{z}) &= \vartheta[(0,0);(0,0)](\mathbf{z},\Omega) \\
\vartheta_2(\mathbf{z}) &= \vartheta[(0,0);(\tfrac{1}{2},\tfrac{1}{2})](\mathbf{z},\Omega) \\
\vartheta_3(\mathbf{z}) &= \vartheta[(0,0);(\tfrac{1}{2},0)](\mathbf{z},\Omega) \\
\vartheta_4(\mathbf{z}) &= \vartheta[(0,0);(0,\tfrac{1}{2})](\mathbf{z},\Omega)
\end{aligned}$$

[1] `http://www.lix.polytechnique.fr/Labo/Pierrick.Gaudry/publis/kummer.mag`

$$
\begin{aligned}
\vartheta_5(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},0);(0,0)](\mathbf{z},\Omega)\\
\vartheta_6(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},0);(0,\tfrac{1}{2})](\mathbf{z},\Omega)\\
\vartheta_7(\mathbf{z}) &= \vartheta[(0,\tfrac{1}{2});(0,0)](\mathbf{z},\Omega)\\
\vartheta_8(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},\tfrac{1}{2});(0,0)](\mathbf{z},\Omega)\\
\vartheta_9(\mathbf{z}) &= \vartheta[(0,\tfrac{1}{2});(\tfrac{1}{2},0)](\mathbf{z},\Omega)\\
\vartheta_{10}(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},\tfrac{1}{2});(\tfrac{1}{2},\tfrac{1}{2})](\mathbf{z},\Omega)
\end{aligned}
$$

$$
\begin{aligned}
\vartheta_{11}(\mathbf{z}) &= \vartheta[(0,\tfrac{1}{2});(0,\tfrac{1}{2})](\mathbf{z},\Omega)\\
\vartheta_{12}(\mathbf{z}) &= \vartheta[(0,\tfrac{1}{2});(\tfrac{1}{2},\tfrac{1}{2})](\mathbf{z},\Omega)\\
\vartheta_{13}(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},0);(\tfrac{1}{2},0)](\mathbf{z},\Omega)\\
\vartheta_{14}(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},\tfrac{1}{2});(\tfrac{1}{2},0)](\mathbf{z},\Omega)\\
\vartheta_{15}(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},0);(\tfrac{1}{2},\tfrac{1}{2})](\mathbf{z},\Omega)\\
\vartheta_{16}(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},\tfrac{1}{2});(0,\tfrac{1}{2})](\mathbf{z},\Omega)
\end{aligned}
$$

For the duplication and addition formulae we also need the 4 Theta functions ϑ_1, ϑ_5, ϑ_7, ϑ_8 that are evaluated at 2Ω. Since they play a dual role to ϑ_1, ϑ_2, ϑ_3, ϑ_4, we renumber them and use the Θ symbol to mark the fact that they are evaluated at the isogenous abelian variety.

$$
\begin{aligned}
\Theta_1(\mathbf{z}) &= \vartheta[(0,0);(0,0)](\mathbf{z},2\Omega)\\
\Theta_2(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},\tfrac{1}{2});(0,0)](\mathbf{z},2\Omega)\\
\Theta_3(\mathbf{z}) &= \vartheta[(0,\tfrac{1}{2});(0,0)](\mathbf{z},2\Omega)\\
\Theta_4(\mathbf{z}) &= \vartheta[(\tfrac{1}{2},0);(0,0)](\mathbf{z},2\Omega)\ .
\end{aligned}
$$

7.2 Duplication and addition formulae

The main formulae on which this work relies are the following:

$$
\begin{aligned}
\vartheta_1(\mathbf{z})\vartheta_1(0) &= \Theta_1(\mathbf{z})^2+\Theta_2(\mathbf{z})^2+\Theta_3(\mathbf{z})^2+\Theta_4(\mathbf{z})^2\\
\vartheta_2(\mathbf{z})\vartheta_2(0) &= \Theta_1(\mathbf{z})^2+\Theta_2(\mathbf{z})^2-\Theta_3(\mathbf{z})^2-\Theta_4(\mathbf{z})^2\\
\vartheta_3(\mathbf{z})\vartheta_3(0) &= \Theta_1(\mathbf{z})^2-\Theta_2(\mathbf{z})^2+\Theta_3(\mathbf{z})^2-\Theta_4(\mathbf{z})^2\\
\vartheta_4(\mathbf{z})\vartheta_4(0) &= \Theta_1(\mathbf{z})^2-\Theta_2(\mathbf{z})^2-\Theta_3(\mathbf{z})^2+\Theta_4(\mathbf{z})^2\ ,
\end{aligned}
\tag{2}
$$

and

$$
\begin{aligned}
4\Theta_1(2\mathbf{z})\Theta_1(0) &= \vartheta_1(\mathbf{z})^2+\vartheta_2(\mathbf{z})^2+\vartheta_3(\mathbf{z})^2+\vartheta_4(\mathbf{z})^2\\
4\Theta_2(2\mathbf{z})\Theta_2(0) &= \vartheta_1(\mathbf{z})^2+\vartheta_2(\mathbf{z})^2-\vartheta_3(\mathbf{z})^2-\vartheta_4(\mathbf{z})^2\\
4\Theta_3(2\mathbf{z})\Theta_3(0) &= \vartheta_1(\mathbf{z})^2-\vartheta_2(\mathbf{z})^2+\vartheta_3(\mathbf{z})^2-\vartheta_4(\mathbf{z})^2\\
4\Theta_4(2\mathbf{z})\Theta_4(0) &= \vartheta_1(\mathbf{z})^2-\vartheta_2(\mathbf{z})^2-\vartheta_3(\mathbf{z})^2+\vartheta_4(\mathbf{z})^2\ .
\end{aligned}
\tag{3}
$$

And for all vectors $\mathbf{z}$ and $\mathbf{z}'$ in $\mathbb{C}^2$, we have

$$
\begin{aligned}
\vartheta_1(\mathbf{z}+\mathbf{z}')\vartheta_1(\mathbf{z}-\mathbf{z}') &= \Theta_1(2\mathbf{z})\Theta_1(2\mathbf{z}')+\Theta_2(2\mathbf{z})\Theta_2(2\mathbf{z}')+\Theta_3(2\mathbf{z})\Theta_3(2\mathbf{z}')+\Theta_4(2\mathbf{z})\Theta_4(2\mathbf{z}')\\
\vartheta_2(\mathbf{z}+\mathbf{z}')\vartheta_2(\mathbf{z}-\mathbf{z}') &= \Theta_1(2\mathbf{z})\Theta_1(2\mathbf{z}')+\Theta_2(2\mathbf{z})\Theta_2(2\mathbf{z}')-\Theta_3(2\mathbf{z})\Theta_3(2\mathbf{z}')-\Theta_4(2\mathbf{z})\Theta_4(2\mathbf{z}')\\
\vartheta_3(\mathbf{z}+\mathbf{z}')\vartheta_3(\mathbf{z}-\mathbf{z}') &= \Theta_1(2\mathbf{z})\Theta_1(2\mathbf{z}')-\Theta_2(2\mathbf{z})\Theta_2(2\mathbf{z}')+\Theta_3(2\mathbf{z})\Theta_3(2\mathbf{z}')-\Theta_4(2\mathbf{z})\Theta_4(2\mathbf{z}')\\
\vartheta_4(\mathbf{z}+\mathbf{z}')\vartheta_4(\mathbf{z}-\mathbf{z}') &= \Theta_1(2\mathbf{z})\Theta_1(2\mathbf{z}')-\Theta_2(2\mathbf{z})\Theta_2(2\mathbf{z}')-\Theta_3(2\mathbf{z})\Theta_3(2\mathbf{z}')+\Theta_4(2\mathbf{z})\Theta_4(2\mathbf{z}')\ ,
\end{aligned}
\tag{4}
$$

and

$$
\begin{aligned}
4\Theta_1(\mathbf{z}+\mathbf{z}')\Theta_1(\mathbf{z}-\mathbf{z}') &= \vartheta_1(\mathbf{z})\vartheta_1(\mathbf{z}') + \vartheta_2(\mathbf{z})\vartheta_2(\mathbf{z}') + \vartheta_3(\mathbf{z})\vartheta_3(\mathbf{z}') + \vartheta_4(\mathbf{z})\vartheta_4(\mathbf{z}') \\
4\Theta_2(\mathbf{z}+\mathbf{z}')\Theta_2(\mathbf{z}-\mathbf{z}') &= \vartheta_1(\mathbf{z})\vartheta_1(\mathbf{z}') + \vartheta_2(\mathbf{z})\vartheta_2(\mathbf{z}') - \vartheta_3(\mathbf{z})\vartheta_3(\mathbf{z}') - \vartheta_4(\mathbf{z})\vartheta_4(\mathbf{z}') \\
4\Theta_3(\mathbf{z}+\mathbf{z}')\Theta_3(\mathbf{z}-\mathbf{z}') &= \vartheta_1(\mathbf{z})\vartheta_1(\mathbf{z}') - \vartheta_2(\mathbf{z})\vartheta_2(\mathbf{z}') + \vartheta_3(\mathbf{z})\vartheta_3(\mathbf{z}') - \vartheta_4(\mathbf{z})\vartheta_4(\mathbf{z}') \\
4\Theta_4(\mathbf{z}+\mathbf{z}')\Theta_4(\mathbf{z}-\mathbf{z}') &= \vartheta_1(\mathbf{z})\vartheta_1(\mathbf{z}') - \vartheta_2(\mathbf{z})\vartheta_2(\mathbf{z}') - \vartheta_3(\mathbf{z})\vartheta_3(\mathbf{z}') + \vartheta_4(\mathbf{z})\vartheta_4(\mathbf{z}')\ .
\end{aligned}
\tag{5}
$$

Note the pretty symmetry and the ubiquity of the main orthogonal matrix.

7.3 Theta constants in terms of fundamental Theta's

We have the following equations that follow directly from Frobenius identities:

$$
\begin{aligned}
\vartheta_5(0)^4 + \vartheta_6(0)^4 &= \vartheta_1(0)^4 - \vartheta_2(0)^4 - \vartheta_3(0)^4 + \vartheta_4(0)^4 \\
\vartheta_5(0)^2\vartheta_6(0)^2 &= \vartheta_1(0)^2\vartheta_4(0)^2 - \vartheta_2(0)^2\vartheta_3(0)^2
\end{aligned}
$$

From these equations, we deduce that there are 4 solutions for $\vartheta_5(0)^2$, and then $\vartheta_6(0)^2$ is completely determined.

Unfortunately, $\vartheta_7(0)^2$ is not uniquely determined from the fundamental theta constants and $\vartheta_5(0)^2$. The most useful relation we can deduce from Frobenius identities is:

$$
\vartheta_7(0)^4 = \vartheta_3(0)^4 - \vartheta_4(0)^4 + \vartheta_5(0)^4.
$$

Hence there are 2 solutions for $\vartheta_7(0)^2$. Then the other Theta constants are completely determined using the following formulae:

$$
\begin{aligned}
\vartheta_8(0)^2 &= (\vartheta_1(0)^2\vartheta_5(0)^2 - \vartheta_4(0)^2\vartheta_6(0)^2)/\vartheta_7(0)^2, \\
\vartheta_9(0)^2 &= (\vartheta_1(0)^2\vartheta_3(0)^2 - \vartheta_2(0)^2\vartheta_4(0)^2)/\vartheta_7(0)^2, \\
\vartheta_{10}(0)^2 &= (\vartheta_2(0)^2\vartheta_5(0)^2 - \vartheta_3(0)^2\vartheta_6(0)^2)/\vartheta_7(0)^2.
\end{aligned}
$$

The pair $(\vartheta_5(0), \vartheta_6(0))$ plays a particular role in our formulae. However, it can be replaced by $(\vartheta_8(0), \vartheta_{10}(0))$ or by $(\vartheta_7(0), \vartheta_9(0))$, and we have analogous formulae:

$$
\begin{aligned}
\vartheta_8(0)^4 + \vartheta_{10}(0)^4 &= \vartheta_1(0)^4 + \vartheta_2(0)^4 - \vartheta_3(0)^4 - \vartheta_4(0)^4 \\
\vartheta_8(0)^2\vartheta_{10}(0)^2 &= \vartheta_1(0)^2\vartheta_2(0)^2 - \vartheta_3(0)^2\vartheta_4(0)^2,
\end{aligned}
$$

$$
\begin{aligned}
\vartheta_7(0)^4 + \vartheta_9(0)^4 &= -\vartheta_1(0)^4 + \vartheta_2(0)^4 - \vartheta_3(0)^4 + \vartheta_4(0)^4 \\
\vartheta_7(0)^2\vartheta_9(0)^2 &= \vartheta_1(0)^2\vartheta_3(0)^2 - \vartheta_2(0)^2\vartheta_4(0)^2.
\end{aligned}
$$

From the Frobenius identities, we can get only relations between squares of Theta constants. Therefore, there is no way to get information on the Theta constants themselves. Usually only their squares are needed anyway.

7.4 Theta functions in terms of fundamental Theta's

By combining two appropriate Frobenius identities, one gets the following expressions for the squares of non-fundamental Theta functions. In this section, for readibility we note simply ϑ_i for the Theta constant $\vartheta_i(0)$.

$$
\vartheta_5(z)^2 = \frac{\vartheta_3(z)^2\vartheta_8^2\vartheta_9^2 + \vartheta_2(z)^2\vartheta_7^2\vartheta_{10}^2 - \vartheta_4(z)^2\vartheta_9^2\vartheta_{10}^2 - \vartheta_1(z)^2\vartheta_7^2\vartheta_8^2}{\vartheta_9^4 - \vartheta_7^4}
$$

$$\vartheta_6(z)^2 = \frac{\vartheta_4(z)^2\vartheta_7^2\vartheta_8^2 + \vartheta_1(z)^2\vartheta_9^2\vartheta_{10}^2 - \vartheta_3(z)^2\vartheta_7^2\vartheta_{10}^2 - \vartheta_2(z)^2\vartheta_8^2\vartheta_9^2}{\vartheta_7^4 - \vartheta_9^4}$$

$$\vartheta_7(z)^2 = \frac{\vartheta_4(z)^2\vartheta_6^2\vartheta_8^2 + \vartheta_2(z)^2\vartheta_5^2\vartheta_{10}^2 - \vartheta_3(z)^2\vartheta_6^2\vartheta_{10}^2 - \vartheta_1(z)^2\vartheta_5^2\vartheta_8^2}{\vartheta_6^4 - \vartheta_5^4}$$

$$\vartheta_8(z)^2 = \frac{\vartheta_4(z)^2\vartheta_6^2\vartheta_7^2 + \vartheta_3(z)^2\vartheta_5^2\vartheta_9^2 - \vartheta_2(z)^2\vartheta_6^2\vartheta_9^2 - \vartheta_1(z)^2\vartheta_5^2\vartheta_7^2}{\vartheta_9^4 - \vartheta_7^4}$$

$$\vartheta_9(z)^2 = \frac{\vartheta_3(z)^2\vartheta_5^2\vartheta_8^2 + \vartheta_1(z)^2\vartheta_6^2\vartheta_{10}^2 - \vartheta_4(z)^2\vartheta_5^2\vartheta_{10}^2 - \vartheta_2(z)^2\vartheta_6^2\vartheta_8^2}{\vartheta_8^4 - \vartheta_{10}^4}$$

$$\vartheta_{10}(z)^2 = \frac{\vartheta_2(z)^2\vartheta_5^2\vartheta_7^2 + \vartheta_1(z)^2\vartheta_6^2\vartheta_9^2 - \vartheta_4(z)^2\vartheta_5^2\vartheta_9^2 - \vartheta_3(z)^2\vartheta_6^2\vartheta_7^2}{\vartheta_7^4 - \vartheta_9^4}$$

$$\vartheta_{11}(z)^2 = \frac{\vartheta_3(z)^2\vartheta_5^2\vartheta_{10}^2 + \vartheta_1(z)^2\vartheta_6^2\vartheta_8^2 - \vartheta_4(z)^2\vartheta_5^2\vartheta_8^2 - \vartheta_2(z)^2\vartheta_6^2\vartheta_{10}^2}{\vartheta_6^4 - \vartheta_5^4}$$

$$\vartheta_{12}(z)^2 = \frac{\vartheta_4(z)^2\vartheta_6^2\vartheta_{10}^2 + \vartheta_2(z)^2\vartheta_5^2\vartheta_8^2 - \vartheta_3(z)^2\vartheta_6^2\vartheta_8^2 - \vartheta_1(z)^2\vartheta_5^2\vartheta_{10}^2}{\vartheta_8^4 - \vartheta_{10}^4}$$

$$\vartheta_{13}(z)^2 = \frac{\vartheta_3(z)^2\vartheta_7^2\vartheta_8^2 + \vartheta_2(z)^2\vartheta_9^2\vartheta_{10}^2 - \vartheta_4(z)^2\vartheta_7^2\vartheta_{10}^2 - \vartheta_1(z)^2\vartheta_8^2\vartheta_9^2}{\vartheta_7^4 - \vartheta_9^4}$$

$$\vartheta_{14}(z)^2 = \frac{\vartheta_4(z)^2\vartheta_6^2\vartheta_9^2 + \vartheta_3(z)^2\vartheta_5^2\vartheta_7^2 - \vartheta_2(z)^2\vartheta_6^2\vartheta_7^2 - \vartheta_1(z)^2\vartheta_5^2\vartheta_9^2}{\vartheta_7^4 - \vartheta_9^4}$$

$$\vartheta_{15}(z)^2 = \frac{\vartheta_3(z)^2\vartheta_9^2\vartheta_{10}^2 + \vartheta_2(z)^2\vartheta_7^2\vartheta_8^2 - \vartheta_4(z)^2\vartheta_8^2\vartheta_9^2 - \vartheta_1(z)^2\vartheta_7^2\vartheta_{10}^2}{\vartheta_7^4 - \vartheta_9^4}$$

$$\vartheta_{16}(z)^2 = \frac{\vartheta_4(z)^2\vartheta_5^2\vartheta_7^2 + \vartheta_3(z)^2\vartheta_6^2\vartheta_9^2 - \vartheta_2(z)^2\vartheta_5^2\vartheta_9^2 - \vartheta_1(z)^2\vartheta_6^2\vartheta_7^2}{\vartheta_7^4 - \vartheta_9^4}$$

7.5 Rosenhain form

From Theorem IIIa.7.6 of [16], we can deduce explicit formulae relating Rosenhain invariants to Theta constants (see also [19]). Since there is some choice, we give some details on the ordering we choose. The finite branch points are

$$B = \{\nu, \mu, \lambda, 1, 0\},$$

in this order, so that the set U of "odd" branch points is

$$U = \{\nu, \lambda, 0\}.$$

Taking appropriate sets V in equation (20) of [19], we readily get

$$\lambda = \frac{\vartheta_1(0)^2\vartheta_3(0)^2}{\vartheta_2(0)^2\vartheta_4(0)^2}, \quad \mu = \frac{\vartheta_3(0)^2\vartheta_8(0)^2}{\vartheta_4(0)^2\vartheta_{10}(0)^2}, \quad \nu = \frac{\vartheta_1(0)^2\vartheta_8(0)^2}{\vartheta_2(0)^2\vartheta_{10}(0)^2}.$$

The following expressions for differences of branch points are also useful to simplify the expressions for the map between $\mathcal{K}$ and the Jacobian of $\mathcal{C}$.

$$\lambda - 1 = \frac{\vartheta_7(0)^2\vartheta_9(0)^2}{\vartheta_2(0)^2\vartheta_4(0)^2}, \quad \mu - 1 = \frac{\vartheta_5(0)^2\vartheta_9(0)^2}{\vartheta_4(0)^2\vartheta_{10}(0)^2}, \quad \nu - 1 = \frac{\vartheta_5(0)^2\vartheta_7(0)^2}{\vartheta_2(0)^2\vartheta_{10}(0)^2},$$

$$\mu - \lambda = \frac{\vartheta_3(0)^2\vartheta_6(0)^2\vartheta_9(0)^2}{\vartheta_2(0)^2\vartheta_4(0)^2\vartheta_{10}(0)^2}, \quad \nu - \lambda = \frac{\vartheta_1(0)^2\vartheta_6(0)^2\vartheta_7(0)^2}{\vartheta_2(0)^2\vartheta_4(0)^2\vartheta_{10}(0)^2}, \quad \nu - \mu = \frac{\vartheta_5(0)^2\vartheta_6(0)^2\vartheta_8(0)^2}{\vartheta_2(0)^2\vartheta_4(0)^2\vartheta_{10}(0)^2}.$$

References

[1] W. Bosma and J. Cannon. *Handbook of Magma functions*, 1997. `http://www.maths.usyd.edu.au:8000/u/magma/`.

[2] D. G. Cantor. Computing in the Jacobian of a hyperelliptic curve. *Math. Comp.*, 48(177):95–101, 1987.

[3] J. W. S. Cassels and E. V. Flynn. *Prolegomena to a middlebrow arithmetic of curves of genus 2*, volume 230 of *London Math. Soc. Lecture Note Ser.* Cambridge University Press, 1996.

[4] D. V. Chudnovsky and G. V. Chudnovsky. Sequences of numbers generated by addition in formal groups and new primality and factorization tests. *Adv. in Appl. Math.*, 7:385–434, 1986.

[5] S. Duquesne. Montgomery scalar multiplication for genus 2 curves. In D. Buell, editor, *ANTS-VI*, volume 3076 of *Lecture Notes in Comput. Sci.*, pages 153–168. Springer–Verlag, 2004.

[6] P. Gaudry and E. Schost. Construction of secure random curves of genus 2 over prime fields. In C. Cachin and J. Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Comput. Sci.*, pages 239–256. Springer-Verlag, 2004.

[7] J.-I. Igusa. *Theta functions*, volume 194 of *Die Grundlehren der mathematischen Wissenschaften*. Springer, 1972.

[8] M. Joye and S.-M. Yen. The Montgomery powering ladder. In B. Kaliski, Ç. Koç, and C. Paar, editors, *CHES 2002*, volume 2523 of *Lecture Notes in Comput. Sci.*, pages 291–320. Springer-Verlag, 2003.

[9] Kiran S. Kedlaya. Counting points on hyperelliptic curves using Monsky-Washnitzer cohomology. *J. Ramanujan Math. Soc.*, 16(4):323–338, 2001.

[10] H. Klingen. *Introductory lectures on Siegel modular forms*, volume 20 of *Cambridge studies in advanced mathematics*. Cambridge University Press, 1990.

[11] H. Lange and Ch. Birkenhake. *Complex abelian varieties*, volume 302 of *Grundlehren der mathematischen Wissenschaften*. Springer–Verlag, 1992.

[12] T. Lange. Montgomery addition for genus two curves. In D. Buell, editor, *ANTS-VI*, volume 3076 of *Lecture Notes in Comput. Sci.*, pages 309–317. Springer–Verlag, 2004.

[13] T. Lange. Formulae for arithmetic on genus 2 hyperelliptic curves. *Appl. Algebra Engrg. Comm. Comput.*, 15(5):295–328, 2005.

[14] J.-F. Mestre. Construction de courbes de genre 2 à partir de leurs modules. In T. Mora and C. Traverso, editors, *Effective methods in algebraic geometry*, volume 94 of *Progr. Math.*, pages 313–334. Birkhäuser, 1991. Proc. Congress in Livorno, Italy, April 17–21, 1990.

[15] D. Mumford. *Tata lectures on theta I*, volume 28 of *Progr. Math.* Birkhauser, 1983.

[16] D. Mumford. *Tata lectures on theta II*, volume 43 of *Progr. Math.* Birkhauser, 1984.

[17] O. Okeya and K. Sakurai. Efficient elliptic curve cryptosystems from a scalar multiplication algorithm with recovery of the y-coordinate on a Montgomery-form elliptic curve. In Ç. Koç, D. Naccache, and C. Paar, editors, *CHES 2001*, volume 2162 of *Lecture Notes in Comput. Sci.*, pages 126–141. Springer–Verlag, 2001.

[18] N. Smart and S. Siksek. A fast Diffie-Hellman protocol in genus 2. *J. of Cryptology*, 12:67–73, 1999.

[19] P. van Wamelen. Equations for the jacobian of a hyperelliptic curve. *Trans. Amer. Math. Soc.*, 350(8):3083–3106, 1998.

[20] A. Weng. Constructing hyperelliptic curves of genus 2 suitable for cryptography. *Math. Comp.*, 72:435–458, 2003.