TWISTED HESSIAN ISOGENIES

THINH DANG AND DUSTIN MOODY

Abstract. Elliptic curves are typically defined by Weierstrass equations. Given a kernel, the well-known Vélu's formula shows how to explicitly write down an isogeny between Weierstrass curves. However, it is not clear how to do the same on other forms of elliptic curves without isomorphisms mapping to and from the Weierstrass form. Previous papers have shown some isogeny formulas for (twisted) Edwards, Huff, and Montgomery forms of elliptic curves. Continuing this line of work, this paper derives an explicit formula for isogenies between elliptic curves in (twisted) Hessian form.

1. Introduction

An elliptic curve is defined as a nonsingular irreducible projective curve of genus one, with a specified point on the curve. An elliptic curve is said to be defined over a field k if the curve is defined over k and the specified point is k-rational.

Let E be an elliptic curve defined over k with a specified point O. It is well known that there exist functions $x, y \in k(E)$ such that the rational map ϕ defined over k by $\phi = (x : y : 1)$ is an isomorphism from E to an elliptic curve in Weierstrass form:

$$Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3$$

and $\phi(O) = (0 : 1 : 0)$, where $a_1, a_2, \ldots, a_6 \in k$ ([1, III.3.1]). Therefore, elliptic curves are typically identified by curves defined by such a Weierstrass equation with the specified point $(0 : 1 : 0)$.

Let E and E' be elliptic curves with specified points O and O' respectively. An isogeny from E to E' is defined as a morphism $\phi : E \to E'$ such that $\phi(O) = O'$. It is a theorem (see [1, III.4.8]) that an isogeny is also a group homomorphism. As a corollary, the kernel of an isogeny is a finite subgroup of the domain. Conversely, if F is a finite subgroup of E, there exists an elliptic curve E' and a separable isogeny $\phi : E \to E'$ such that the kernel of ϕ is F ([1, III.4.12]). Given E and F, Vélu's

2010 *Mathematics Subject Classification.* Primary: 14K02; Secondary: 14H52, 11G05, 11Y16.

Key words and phrases. Elliptic curve; isogeny; Hessian curve.

formula ([2]) shows an explicit expression for ϕ and E', where E and E' are both in Weierstrass form.

However, the Weierstrass equation is only one way to represent an elliptic curve. Other forms of elliptic curves are possible and have been proposed, some with applications in cryptography. Examples include Montgomery curves ([3, 4]), (twisted) Edwards curves ([5, 6, 7]), Huff curves ([8, 9]), and (twisted) Hessian curves ([10]). The first formulas for isogenies defined directly for non-Weierstrass curves was for (twisted) Edwards curves and Huff curves [11]. Shortly thereafter, similar work [12], [13] showed formulas for computing isogenies on Montgomery curves. In this paper, we derive a formula for isogenies on twisted Hessian curves and consider the computational cost of computing image points.

Isogenies have found applications in counting the number of points on an elliptic curve over a finite field (e.g. see [14] and [15]), analyzing the complexity of elliptic-curve discrete logarithms [16], and cryptographic constructions (e.g. [17], [18], and [19]). More efficient isogeny formulas could lead to performance benefits in the above applications.

The organization of the paper is as follows. Section 2 introduces Hessian curves and their generalization called twisted Hessian curves. A summary of the point addition formulas on twisted Hessian curves is included. Section 3 derives formulas for 3-isogenies. Section 4 states and proves the main result for isogenies with a kernel of size $\ell \not\equiv 0 \pmod 3$. Finally, Section 5 examines the main formula's computational cost of computing image points. Some open problems and directions for future work are given in Section 6.

2. Twisted Hessian Curves

A Hessian curve in projective coordinates is defined by the equation

$$X^3 + Y^3 + Z^3 = dXYZ$$

with $27 - d^3 \neq 0$. The Hessian form of elliptic curves has been studied, for example, in [20], [21], and [22], to optimize point addition and scalar multiplication formulas. In addition, as a step towards resistance against side-channel attacks, the Sylvester's addition formula (described below) on Hessian curves can also be used for point doubling and subtraction after a permutation of input coordinates [23]. A generalization of Hessian curves, called twisted Hessian curves, is defined by the equation

$$aX^3 + Y^3 + Z^3 = dXYZ$$

with $a(27a - d^3) \neq 0$. Twisted Hessian curves were used in [10] to provide a complete unified addition formula and improve efficiency for point doubling and tripling over fields of arbitrary characteristic. Other works that tried to optimize arithmetic on (twisted) Hessian curves include [24], [25], and [26].

Definition 1. A *twisted Hessian curve* over a field k is a projective curve $H(a,d)$ defined by the polynomial $aX^3 + Y^3 + Z^3 = dXYZ$ with specified point $(0 : -1 : 1)$ in the projective space $\mathbb{P}(k)^2$, with $a, d \in k$ and $a(27a - d^3) \neq 0$. If $a = 1$, the curve is called a *Hessian curve.*

As an elliptic curve, each twisted Hessian curve must be isomorphic over k to a curve given by a Weierstrass equation. Over a finite field of characteristic not equal to 3, we can find an explicit isomorphism from any twisted Hessian curve to a Weierstrass curve, and conversely, from any Weierstrass curve with a point of order 3 to a twisted Hessian curve. Such isomorphisms are given in [10, Theorem 5.3 and 5.4] and [27].

For convenience, we summarize below the formulas for point addition on twisted Hessian curves. Let $(X_1 : Y_1 : Z_1)$ and $(X_2 : Y_2 : Z_2)$ be points on $H(a,d)$. The inverse of $(X_1 : Y_1 : Z_1)$ is

$$-(X_1 : Y_1 : Z_1) = (X_1 : Z_1 : Y_1).$$

The (Sylvester) standard addition formula is given by:

$$\begin{aligned} X_3 &= X_1^2 Y_2 Z_2 - X_2^2 Y_1 Z_1, \\ Y_3 &= Z_1^2 X_2 Y_2 - Z_2^2 X_1 Y_1, \\ Z_3 &= Y_1^2 X_2 Z_2 - Y_2^2 X_1 Z_1. \end{aligned}$$

If $(X_3, Y_3, Z_3) \neq (0,0,0)$, then $(X_1 : Y_1 : Z_1) + (X_2 : Y_2 : Z_2) = (X_3 : Y_3 : Z_3)$. Another addition formula, called *rotated addition*, is defined by the formula:

$$\begin{aligned} X_3' &= Z_2^2 X_1 Z_1 - Y_1^2 X_2 Y_2, \\ Y_3' &= Y_2^2 Y_1 Z_1 - a X_1^2 X_2 Z_2, \\ Z_3' &= a X_2^2 X_1 Y_1 - Z_1^2 Y_2 Z_2. \end{aligned}$$

If $(X_3', Y_3', Z_3') \neq (0,0,0)$, then $(X_1 : Y_1 : Z_1) + (X_2 : Y_2 : Z_2) = (X_3' : Y_3' : Z_3')$. The completeness follows because $(X_3, Y_3, Z_3) \neq (0,0,0)$ or $(X_3', Y_3', Z_3') \neq (0,0,0)$ by [10, Theorem 4.7].

3. 3-isogenies

In this section, we show how to compute 3-isogenies on twisted Hessian curves, and in the next section, we provide a formula for ℓ-isogenies with $\ell \not\equiv 0 \pmod 3$. To compute an isogeny with kernel of size divisible by 3, we can write the kernel as an internal product of a subgroup of size ℓ not divisible by 3 and one or more subgroups of size 3, and compose the formulas for each factor.

To derive the result for 3-isogenies, we begin by characterizing all points of order 3 on a twisted Hessian curve. Let c be a cubic root of a. It can be easily verified that the point $(1 : 0 : -c)$ and its inverse $(1 : -c : 0)$ both have order 3. In addition, if $\omega^3 = 1$ and $\omega \neq 1$, then $(0 : -\omega : 1)$ and its inverse $(0 : 1 : -\omega)$ have order 3. The verification has been done in [10, Theorem 5.1]. In fact, these are the only points of order 3 on a twisted Hessian curve. We prove this in the next theorem.

Theorem 1. *Let $P = (X : Y : Z) \neq (0 : -1 : 1)$ be a point on $H(a, d)$. Then P has order 3 if and only if $XYZ = 0$.*

Proof. Suppose $XYZ = 0$. If $X = 0$, by the defining equation of $H(a, d)$, it follows that $P = (0 : -\omega : 1)$ where $\omega^3 = 1$. When $\omega \neq 1$, then $(0 : -\omega : 1)$ has order 3. If $Y = 0$, we must have $P = (1 : 0 : -c)$ where $c^3 = a$, and $(1 : 0 : -c)$ has order 3. Similarly, if $Z = 0$, then $P = (1 : -c : 0)$ has order 3.

For the converse, suppose P has order 3 and $XYZ \neq 0$. By the rotated addition law,

$$\begin{aligned} 2P &= (X(Z^3 - Y^3) : Z(Y^3 - aX^3) : Y(aX^3 - Z^3)) \\ &= -P = (X : Z : Y). \end{aligned}$$

So $Z^3 - Y^3 = Y^3 - aX^3 = aX^3 - Z^3 \neq 0$.

Consider two cases, depending on the characteristic of k. Suppose first that k has characteristic $\neq 3$. Then,

$$2(Y^3 - aX^3) = (Z^3 - Y^3) + (aX^3 - Z^3) = aX^3 - Y^3,$$

which implies $aX^3 - Y^3 = 0$. This is a contradiction. Alternatively, if k has characteristic 3, then $Z^3 - Y^3 = aX^3 - Z^3$, which implies $aX^3 = 2Z^3 + 2Y^3$. Substituting this into $aX^3 + Y^3 + Z^3 = dXYZ$ gives $d = 0$, since XYZ is assumed to be nonzero. Therefore, $a(27a - d^3) = 0$, and this contradicts the definition of $H(a, d)$.

□

We now turn to formulas for 3-isogenies of twisted Hessian curves. As seen in the proof, a kernel of size 3 is either generated by $(0 : -\omega : 1)$ with $\omega^3 = 1$ and $\omega \neq 1$ or by $(1 : -c : 0)$ with $c^3 = a$. First, we consider 3-isogenies with their kernel generated by $(0 : -\omega : 1)$. Such a map can be obtained by composing the 3-isogeny given in [10, Theorem 5.4] from a twisted Hessian curve to a Weierstrass curve of the form

$Y^2Z + a_1XYZ + a_3YZ^2 = X^3$ with the isomorphism given in [10, Theorem 5.4] between such a Weierstrass curve and a twisted Hessian curve. The result of such composition is stated in Theorem 2.

Theorem 2. *Let $\omega^3 = 1$ and $\omega \neq 1$. The map*

$$(X : Y : Z) \mapsto (XYZ : aX^3 + \omega^2Y^3 + \omega Z^3 : aX^3 + \omega Y^3 + \omega^2Z^3)$$

is an isogeny from $H(a, d)$ to $H(d^3 - 27a, 3d)$ with the kernel

$$\langle(0 : -\omega : 1)\rangle = \langle(0 : -\omega^2 : 1)\rangle = \{(0 : -1 : 1), (0 : -\omega : 1), (0 : -\omega^2 : 1)\}.$$

Proof. We leave the straightforward verification to the reader. □

Next, we consider 3-isogenies with kernel generated by the point $(1 : -c : 0)$, where $c^3 = a$. The only formula for such isogenies that we are aware of is given in [28, Proposition 4] for Hessian curves over characteristic 3. We restate the result here.

Theorem 3. *Let k have characteristic 3. The map $\sigma : H(1, d^{3^{i+1}}) \to H(1, d^{3^i})$ defined by*

$$\sigma(X : Y : Z) = (d^{2\cdot 3^i}XYZ : Y^2Z + X^2Y + XZ^2 : XY^2 + X^2Z + YZ^2)$$

is an isogeny. Moreover, $f : H(1, d^{3^i}) \to H(1, d^{3^{i+1}})$ defined by $f(X : Y : Z) = (X^3 : Y^3 : Z^3)$ is an isogeny, and $f \circ \sigma(P) = 3P$ for each P on $H(k, 1, d^{3^{i+1}})$. The kernel of σ is $\{(0 : -1 : 1), (-1 : 1 : 0), (-1 : 0 : 1)\}$.

We generalize Theorem 3 to 3-isogenies on twisted Hessian curves $H(a, d)$ over any characteristic with kernel $\langle(1 : -c : 0)\rangle$, where $c^3 = a$.

Theorem 4. *The rational map*

$$\phi = \big(XYZ : c^2X^2Z + cXY^2 + YZ^2 : c^2X^2Y + cXZ^2 + Y^2Z\big).$$

is an isogeny from $H(a, d)$ to $H(A, D)$, where $c^3 = a$,

$$A = d^2c + 3dc^2 + 9a \text{ and } D = d + 6c$$

with kernel

$$\langle(1 : -c : 0)\rangle = \langle(1 : 0 : -c)\rangle = \{(0 : -1 : 1), (1 : -c : 0), (1 : 0 : -c)\}.$$

Proof. Let $f = xy$, $g = c^2x^2 + cxy^2 + y$, and $h = c^2x^2y + cxz^2 + y^2$ be the dehomogenized coordinate maps. Also let A and D be as given in the theorem statement. Then,

$$Af^3 + g^3 + h^3 - Dfgh = (ax^3y^3 - cdx^2y^2 + ax^3 + y^3)(ax^3 + y^3 + 1 - dxy).$$

This shows that the range of the rational map ϕ is indeed $H(A, D)$. It remains to check that the kernel is as claimed. Let $P = (X : Y : Z)$ and suppose $\phi(P) = (0 : -1 : 1)$, then $XYZ = 0$.

(1) If $X = 0$, then $YZ^2 = -Y^2Z$, i.e. $Z = -Y$ and $P = (0 : -1 : 1)$.
(2) If $Y = 0$, then $c^2X^2Z = -cXZ^2$, i.e. $cX = -Z$ and $P = (1 : 0 : -c)$.
(3) If $Z = 0$, then $cXY^2 = -c^2X^2Y$, i.e. $Y = -cX$ and $P = (1 : -c : 0)$.

Conversely, by straightforward calculation, we see that $\phi(P) = (0 : -1 : 1)$ for each such P. □

4. Isogenies of degree ℓ, where $\ell \neq 3$

In this section we look at the ℓ-isogeny formulas, where $\ell \neq 3$. One approach for obtaining such an ℓ-isogeny between twisted Hessian curves is to compose the isogeny given by Vélu's formula with isomorphisms to and from Weierstrass curves. This approach, however, doesn't lead to a simple formula. Moreover, the resulting codomain twisted Hessian curve is dependent on the choice of point of order 3 on the codomain Weierstrass curve produced by Vélu's formula. We prove our main twisted Hessian isogeny result as follows.

Theorem 5. *Let $F = \{(0 : -1 : 1)\} \cup \{(s_i : t_i : 1)\}_{i=1}^n$ be a finite subgroup of $H(a, d)$ of size $\ell = n + 1$, where ℓ is not divisible by 3. Then, F is the kernel of an isogeny from $H(a, d)$ to $H(A, D)$ defined by*

$$\phi(P) = \left(\prod_{R \in F} X(P + R) : \prod_{R \in F} Y(P + R) : \prod_{R \in F} Z(P + R) \right).$$

where $A = a^\ell$ and

$$D = \frac{(1 - 2n)d + 6\sum_{i=1}^n 1/(s_it_i)}{\prod_{i=1}^n s_i}.$$

Note that in the equation for ϕ, for each point $P + R$, the choice of representative of $P + R$ in projective coordinates does not affect the result $\phi(P)$. Also, by Theorem 1, $s_it_i \neq 0$ for each $i \in \{1, 2, \ldots, n\}$.

Proof. We start by writing down a rational form of the map ϕ given in the theorem, which is derived from the standard addition formula. Let

$$\phi_Y := \frac{y}{x} \prod_i \frac{xy - s_it_i}{s_i^2 y - t_i x^2} \text{ and } \phi_Z := \frac{1}{x} \prod_i \frac{t_i^2 x - s_i y^2}{s_i^2 y - t_i x^2}.$$

That is, $\phi(x : y : 1) = (1 : \phi_Y : \phi_Z)$. Define

$$G = A + \phi_Y^3 + \phi_Z^3 - D\phi_Y\phi_Z \in k(H),$$

where $A, D \in k$ are to be determined.

Our goal is show that $G = 0$ for $A, D \in k$ as stated in the theorem. To this end, by Proposition [1, II.1.2], it suffices to show that G has no poles and $G(Q) = 0$ for some Q on H. By the definitions of ϕ_Y and ϕ_Z, if P is a pole of G, then $X(\phi(P)) = 0$, which is equivalent to $X(P + R) = 0$ for some $R \in F$. Let $Q = P + R$. From the formula of ϕ, it can be seen that ϕ is invariant under translation by any point in F. So $\phi(P) = \phi(Q)$ and $X(Q) = 0$. Therefore, if G has a pole at some point P, then G also has a pole at some point Q with $X(Q) = 0$. By subsituting $X = 0$ into the defining equation of H, we find that the only points Q with $X(Q) = 0$ are $\{(0 : -\omega : 1) \mid \omega^3 = 1\}$.

Let $P = (0 : -\omega : 1)$ with $\omega^3 = 1$. We'll show that P is not a pole of G for some A and D in k and hence by the arguments in the preceding paragraph, G has no pole at all and thus is constant.

First, we assume that the characteristic of k is not 3. Then, a uniformizer for $k[H]_P$ is x (by [29, Theorem 1 of Section 3.2]). We need the following facts:

- $k[H]_P$ is a discrete valuation ring (by [1, Proposition II.1.1]).
- $k[H]_P$ has a unique maximal ideal $M_P := \{q \in k[H]_P \mid q(P) = 0\}$ ([29, Section 2.4]).
- $k(H)$ is the field of quotients of $k[H]_P$.
- The field k is a subring of $k[H]_P$, and the map $b \mapsto b + M_p$ from k to $k[H]_p/M_P$ is a field isomorphism.

We can conclude that the function that maps each element in $k(H)$ to its Laurent series expansion in $k((x))$ is a one-to-one ring homomorphism [29, Problem 2.32]. We write $f = \sum_{i=m}^{r} c_i x^i$ where $m \in \mathbb{Z}$ and $r \in \mathbb{Z} \cup \{\infty\}$ to mean that f has the Laurent series expansion $\sum_{i=m}^{r} c_i x^i$.

Next, we find the series expansion of y in terms of x. The order of y at P is $\mathrm{ord}_P(y) = 0$, since y is defined and is nonzero at P. Thus y has a power series expansion $y = \sum_{i=0}^{\infty} c_i x^i$. As $ax^3 + y^3 + 1 - dxy$ is zero in $k(H)$ and the function that maps each element in $k(H)$ to its Laurent series expansion is a one-to-one ring homomorphism,

$$ax^3 + (\sum_{i=0}^{\infty} c_i x^i)^3 + 1 - dx(\sum_{i=0}^{\infty} c_i x^i) = 0.$$

Since $y - c_0$ vanishes at P, we have $c_0 = -\omega$. Then, solving for c_1 and c_2 gives

$$y = -\omega - \frac{d}{3\omega} x + O(x^3).$$

In the remainder of the proof, we use the definition $S := \prod_{i=1}^{n} s_i$, and since $-(s_i : t_i : 1) = (s_i/t_i : 1/t_i : 1)$, we have

$$\prod_{i=1}^{n} t_i = 1, \quad \sum_i \frac{t_i^2}{s_i} = \sum_i \frac{1}{s_i t_i}, \quad \text{and} \sum_{i<j} \frac{t_i^2 t_j^2}{s_i s_j} = \sum_{i<j} \frac{1}{s_i s_j t_i t_j}.$$

Moreover, we also use the following formula for the product of power series:

$$\begin{aligned}\prod_{i=1}^{n} c_i^{(0)} + & c_i^{(1)} z + c_i^{(2)} z^2 + O(z^3) \\ &= \prod_{i=1}^{n} c_i^{(0)} + \Big(\prod_{i=1}^{n} c_i^{(0)}\Big)\Big(\sum_{i=1}^{n} \frac{c_i^{(1)}}{c_i^{(0)}}\Big) z \\ &\qquad + \Big(\prod_{i=1}^{n} c_i^{(0)}\Big)\Big(\sum_{i=1}^{n} \frac{c_i^{(2)}}{c_i^{(0)}} + \sum_{1\le i<j\le n}^{n} \frac{c_i^{(1)} c_j^{(1)}}{c_i^{(0)} c_j^{(0)}}\Big) z^2 + O(z^3).\end{aligned}$$

Substitution into G, with some additional simplifying yields

$$G = G_{-3} x^{-3} + G_{-2} x^{-2} + G_{-1} x^{-1} + O(1),$$

where

$$\begin{aligned} G_{-3} &= 0, \\ G_{-2} &= \frac{\omega}{S^3}\Big((2n-1)d - 6\sum_{i=1}^{n} \frac{1}{s_i t_i} + DS\Big), \\ G_{-1} &= \frac{\omega^2 d}{3S^3}\Big((2n-1)d - 6\sum_{i=1}^{n} \frac{1}{s_i t_i} + DS\Big). \end{aligned}$$

Hence, $G_{-2} = G_{-1} = 0$ if

$$D = \frac{(1-2n)d + 6\sum_{i=1}^{n} \frac{1}{s_i t_i}}{S};$$

i.e. G has no pole and thus is constant.

Finally, we consider the case when k has characteristic 3. In particular, x is not a uniformizer for $k[H]_P$. Instead, $\omega = 1$, and $u = y + 1$ is a uniformizer for $k[H]_P$. Since x is defined and vanishes at P, i.e. $\mathrm{ord}_P(x) \ge 1$, x has a power series expansion $x = \sum_{i=0}^{\infty} b_i u^i$ with $b_0 = 0$. Hence,

$$a(\sum_{i=0}^{\infty} b_i u^i)^3 + (u-1)^3 + 1 - d(\sum_{i=0}^{\infty} b_i u^i)(u-1) = 0.$$

Solving for $b_1, b_2, \ldots$, we get

$$x = -\frac{1}{d}(u^3 + u^4 + \cdots + u^8) + \frac{a - d^3}{d^4}(u^9 + \cdots + u^{14}) + O(u^{15}).$$

Note that in characteristic 3, by the definition of twisted Hessian curves, $d \neq 0$. Then,

$$\begin{aligned}\frac{xy - s_it_i}{s_i^2y - t_ix^2} = &\frac{t_i}{s_i}(1 + x + x^2) + (\frac{t_i}{s_i} - \frac{1}{ds_i^2})(x^3 + x^4 + x^5)\\ &+ (\frac{t_i}{s_i} - \frac{1}{ds_i^2} - \frac{t_i^2}{d^2s_i^3})(x^6 + x^7 + x^8) + O(x^9),\end{aligned}$$

and

$$\begin{aligned}\frac{t_i^2x - s_iy^2}{s_i^2y - t_ix^2} = &\frac{1}{s_i}(1 - x) + \frac{t_i^2}{ds_i^2}(x^3 - x^4)\\ &+ (\frac{t_i^2}{s_i^2} - \frac{t_i}{d^2s_i^3})(x^6 - x^7) + O(x^9).\end{aligned}$$

Hence,

$$\prod_{i=1}^{n} \frac{xy - s_it_i}{s_i^2y - t_ix^2} = U_0 + U_1u + \cdots + U_8u^8 + O(u^9),$$

where

$$\begin{aligned}U_i = \frac{1}{S}\Bigg(&\binom{n+i-1}{i} + \binom{n+i-4}{i-3}\sum_i \frac{-1}{ds_it_i}\\ &+\frac{1}{2}\binom{n+i-7}{i-6}\Big(\sum_i \frac{-1}{ds_it_i}\Big)^2 + \binom{n+i-7}{i-6}\sum_i -\frac{1}{2d^2s_i^2t_i^2} - \frac{t_i}{d^2s_i^2}\Bigg),\end{aligned}$$

and

$$\prod_i \frac{t_i^2x - s_iy^2}{s_i^2y - t_ix^2} = V_0 + V_1u + \cdots + V_8u^8 + O(u^9),$$

where

$$\begin{aligned}V_i = \frac{(-1)^i}{S}\Bigg(&\binom{n}{i} - \binom{n}{i-3}\sum_i \frac{t_i^2}{ds_i}\\ &+\frac{1}{2}\binom{n}{i-6}\Big(\sum_i \frac{t_i^2}{ds_i}\Big)^2 + \binom{n}{i-6}\sum_i -\frac{t_i}{d^2s_i^2} + \frac{t_i^2}{s_i} - \frac{t_i^4}{2d^2s_i^2}\Bigg).\end{aligned}$$

In the above, we define $\binom{p}{q} = 0$ if $q < 0$. One can verify the U_i and V_i by straightforward induction on n. Substitution into G and simplifying (remember k has characteristic 3) using the identities:

$$\sum_i \frac{t_i^2}{s_i} = \sum_i \frac{1}{s_i t_i} \quad \text{and} \quad \sum_i \frac{t_i^4}{s_i^2} = \sum_i \frac{1}{s_i^2 t_i^2},$$

gives

$$G = G_{-6}u^{-6} + G_{-3}u^{-3} + O(1),$$

where

$$G_{-6} = \frac{d^2(d + dn + 2DS)}{2S^3} \quad \text{and} \quad G_{-3} = \frac{d^2(d + dn^5 + 2DS)}{S^3}.$$

In characteristic 3, we have $n^5 = n$ for integers n. Therefore, if

$$D = \frac{-d - dn}{2S} = \frac{(1 - 2n)d}{S},$$

then $G_{-6} = G_{-3} = 0$ and G is constant.

We note that if $G(Q) = 0$ for some Q, then $G = 0$. Next, we find $A \in k$ such that G vanishes at $Q = (1 : -c : 0) \in H$ where $c^3 = a$. By [10, Theorem 4.1], i.e. $(X : Y : Z) + (1 : -c : 0) = (Y : cZ : c^2X)$,

$$\begin{aligned}
\phi(Q) &= \left(\prod_{R \in F} X(Q + R) : \prod_{R \in F} Y(Q + R) : \prod_{R \in F} Z(Q + R) \right) \\
&= \left(\prod_{R \in F} Y(R) : c^\ell \prod_{R \in F} Z(R) : c^{2\ell} \prod_{R \in F} X(R) \right) \\
&= \left(\prod_{R \in F} Y(R)/Z(R) : c^\ell : 0 \right) \\
&= (-1 : c^\ell : 0).
\end{aligned}$$

So $G(Q) = A - c^{3\ell} = A - a^\ell$. Solving $G(Q) = 0$ for A gives $A = a^\ell$.

It remains to check that the kernel of ϕ is indeed F. It's clear that $\phi(P) = (0 : -1 : 1)$ if $P \in F$. For the converse, suppose $\phi(P) = (0 : -1 : 1)$. Then $X(Q) = 0$ where $Q = P + R$ for some $R \in F$. So $Q = (0 : -1 : 1)$ or $Q = (0 : -\omega : 1)$ for some $\omega \neq 1$ such that $\omega^3 = 1$. If $Q = (0 : -1 : 1)$, $P = -R \in F$. Else, by [10, Theorem 4.6],

$$\phi(Q) = \phi(0 : -\omega : 1) = (0 : -\omega^\ell : 1) \neq (0 : -1 : 1)$$

since $3 \nmid \ell$. However, this contradicts $\phi(Q) = \phi(P) = (0 : -1 : 1)$. That concludes the proof. □

5. Cost of computing image points

In this section, we examine the computational complexity of the isogeny formula in Theorem 5. We do so by counting the number of multiplications (denoted by M), squarings, (denoted by S), and inversions (denoted by I). Since the rotated addition formula is complete if a is not a cube, we use the rotated addition formula in this section.

In general, the computational cost depends on many factors, for example, how the points are represented: projective, affine, or both (mixed), whether we want to avoid inversions entirely, or how the coordinate maps are represented (e.g. polynomials or rational functions). Here, we mainly focus on the purely projective case with coordinate maps given by homogeneous polynomials of the same degree, and the purely affine case with coordinate maps given by rational functions.

For simplicity, we assume that the size of the kernel is odd. We let the kernel be

$$\begin{aligned} F &= \{O\} \cup \{R_i\}_{i=1}^{s} \cup \{-R_i\}_{i=1}^{s} \\ &= \{(0 : -1 : 1)\} \cup \{(\alpha_i : \beta_i : \gamma_i)\}_{i=1}^{s} \cup \{(\alpha_i : \gamma_i : \beta_i)\}_{i=1}^{s}. \end{aligned}$$

We separate the computation into two phases: processing the kernel points and computing the image of an input point. By the rotated addition formula, $(X : Y : Z) + (\alpha_i : \beta_i : \gamma_i) = (X' : Y' : Z')$ where

$$\begin{aligned} X' &= XZ\gamma_i^2 - Y^2\alpha_i\beta_i, \\ Y' &= YZ\beta_i^2 - X^2 a\alpha_i\gamma_i, \\ Z' &= XYa\alpha_i^2 - Z^2\beta_i\gamma_i, \end{aligned}$$

and $(X : Y : Z) + (\alpha_i : \gamma_i : \beta_i) = (X'' : Y'' : Z'')$ where

$$\begin{aligned} X'' &= XZ\beta_i^2 - Y^2\alpha_i\gamma_i, \\ Y'' &= YZ\gamma_i^2 - X^2 a\alpha_i\beta_i, \\ Z'' &= XYa\alpha_i^2 - Z^2\beta_i\gamma_i. \end{aligned}$$

For processing the kernel, we can pre-compute

$$\alpha_i^2, a\alpha_i^2, \beta_i^2, \gamma_i^2, \alpha_i\beta_i, a\alpha_i\beta_i, \alpha_i\gamma_i, a\alpha_i\gamma_i, \beta_i\gamma_i,$$

for all i, which takes $3sS + 6sM$. Computing an image point

$$\Big(X\prod_{i=1}^{s}(XZ\gamma_i^2 - Y^2\alpha_i\beta_i)(XZ\beta_i^2 - Y^2\alpha_i\gamma_i) : \\ Y\prod_{i=1}^{s}(YZ\beta_i^2 - X^2a\alpha_i\gamma_i)(YZ\gamma_i^2 - X^2a\alpha_i\beta_i) : \\ Z\big(\prod_{i=1}^{s}XYa\alpha_i^2 - Z^2\beta_i\gamma_i\big)^2\Big)$$

then takes $3S$ for (XZ, YZ, XY), $3S$ for (X^2, Y^2, Z^2), $6sM$ for the x-coordinate, $6sM$ for the y-coordinate, and $3sM + 1S$ for the z-coordinate. In total, computing an image point takes $(15s+3)M + 4S$.

In affine coordinates, let

$$F = \{(0, -1)\} \cup \{(\alpha_i, \beta_i)\}_{i=1}^{s} \cup \{(\alpha_i/\beta_i, 1/\beta_i)\}_{i=1}^{s},$$

and the formula then becomes

$$(x, y) \mapsto \Big(x\prod_{i=1}^{s}\frac{(x - \alpha_i\beta_i y^2)(\beta_i^2 x - \alpha_i y^2)}{(a\alpha_i^2 xy - \beta_i)^2}, y\prod_{i=1}^{s}\frac{(\beta_i^2 y - a\alpha_i x^2)(y - a\alpha_i\beta_i x^2)}{(a\alpha_i^2 xy - \beta_i)^2}\Big).$$

For processing the kernel, we pre-compute

$$a\alpha_i^2, \beta_i^2, \alpha_i\beta_i, a\alpha_i\beta_i$$

which takes $2sS + 3sM$. Computing an image point then takes $(12s + 1)M + 3S + I$. We do not claim these operation counts are optimal.

For comparison, consider the isogeny formula from [11] for Edwards curves. The authors reported the cost of $(6s + 1)M + 2S + I$ in affine coordinates or $(6s+3)M+4S$ in projective coordinates, for computing an image point. However, in each case, up to sI were required for processing the kernel. We can do better. Suppose the kernel is

$$F = \{(0 : 1 : 1)\} \cup \{(\alpha_i : \beta_i : \gamma_i)\}_{i=1}^{s} \cup \{(-\alpha_i : \beta_i : \gamma_i)\}_{i=1}^{s}.$$

The isogeny is

$$(x : y : z) \mapsto \Big(x\prod_{i=1}^{s}\beta_i^2\gamma_i^4x^2z^2 - \alpha_i^2\gamma^4y^2z^2 : \\ y\prod_{i=1}^{s}\beta_i^2\gamma_i^4y^2z^2 - \alpha_i^2\gamma_i^4x^2z^2 : \\ z\prod_{i=1}^{s}\beta_i^2\gamma_i^4z^4 - d^2\alpha_i^2\beta_i^4x^2y^2\Big).$$

For processing the kernel, one can compute $\beta_i^2\gamma_i^4, \alpha_i^2\gamma^4$, and $d^2\alpha_i^2\beta_i^4$, for all i, with $(5s+1)S+4sM$. For computing the image point, x^2z^2, y^2z^2, x^2y^2, and z^4, take $3M$ and $4S$. If the characteristic is not 2, we can compute each pair of $2(\beta_i^2\gamma_i^4x^2z^2 - \alpha_i^2\gamma^4y^2z^2)$ and $2(\beta_i^2\gamma_i^4y^2z^2 - \alpha_i^2\gamma_i^4x^2z^2)$ for the x and y coordinates with only $2M$ using the identities:

$$\begin{aligned}2(ax-by) &= (a-b)(x+y)+(a+b)(x-y) \text{ and}\\ 2(ay-bx) &= (a-b)(x+y)-(a+b)(x-y).\end{aligned}$$

Each factor $\beta_i^2\gamma_i^4z^4 - d^2\alpha_i^2\beta_i^4x^2y^2$ in the z coordinate takes $2M$, and let $cost(2^s)$ be the cost of computing 2^s. Multiplication of all the factors in the x and y coordinates takes $2sM$, and multiplication of the factors in the z coordinate including 2^s takes $(s+1)M$. Therefore, the total cost of computing an image point is $4S+(7s+1)M+cost(2^s)$.

Similarly, in affine coordinates, we can compute the Edwards isogeny map

$$(x,y) \mapsto \left(x\prod_{i=1}^{s}\frac{\beta_i^2x^2-\alpha_i^2y^2}{\beta_i^2-d^2\alpha_i^2\beta_i^4x^2y^2}, y\prod_{i=1}^{s}\frac{\beta_i^2y^2-\alpha_i^2x^2}{\beta_i^2-d^2\alpha_i^2\beta_i^4x^2y^2}\right)$$

using $(3s+1)S+2sM$ for processing the kernel and $(6s+1)M+2S+I+cost(2^s)$.

The formula for Huff curves from the same paper [11] doesn't seem to have an efficient expression when projectivized, so we will not analyze that here and hence use the analysis from the original paper.

Figure 1 summarizes the comparison. We obtained the cost for Vélu's formula by straightforward counting. We note that computing the image point of an isogeny seems to be fastest on the Edwards model of elliptic curves. The twisted Hessian isogeny formulas in this paper are roughly about the same cost as using Velu's formula on Weierstrass curves.

6. Conclusion

In this work we looked at computing isogenies between elliptic curves represented as twisted Hessian curves. There still exist other models of curves for which direct isogeny formulas are not known, such as Jacobi quartics and Jacobi intersections [30, 31]. It would be interesting to see if simple isogeny formulas exist for these models. We note that the original Velu isogeny formulas are expressed as a sum, while the more recent Edwards, Hessian, and Montgomery formulas all involve a product of expressions involving the kernel points. Is there a multiplicative version of Velu's formulas? Or additive expressions for isogenies of the alternate models of elliptic curves?

Formula	Process	Operations			
		S	M	I	Others
Twisted Hessian (projective)	kernel	$3s$	$6s$		
	input point	4	$15s+3$		
Twisted Hessian (affine)	kernel	$2s$	$3s$		
	input point	3	$12s+1$	1	
Edwards (projective)	kernel	$5s+1$	$4s$		
	input point	4	$7s+1$		$cost(2^s)$
Edwards (affine)	kernel	$3s+1$	$2s$		
	input point	2	$6s+1$	1	$cost(2^s)$
Huff (affine)	kernel	$2s+2$	$2s$		
	input point	2	$6s-2$	2	
Vélu's	kernel	s	$9s$		
	input point	1	$13s+1$	1	

FIGURE 1. A comparison of isogeny computation costs for various models of elliptic curves. For each formula, the first row shows the number of operations for processing the kernel and the second row shows the number of operations dependent on input point.

We leave it as future work to further optimize the formulas presented. This would include finding efficient formulas for low degree isogenies, such as 2-isogenies and 3-isogenies, of twisted Hessian curves. Low degree isogenies are used in post-quantum cryptographic isogeny schemes, and if optimized formulas can be found, it may lead to implementing these isogeny cryptosystems using twisted Hessian curves.

REFERENCES

[1] Joseph H Silverman. *The arithmetic of elliptic curves*. Graduate Texts in Mathematics. Springer, 2nd edition, 2009.

[2] Jacques Vélu. Isogénies entre courbes elliptiques. *CR Acad. Sci. Paris, Séries A*, 273:305–347, 1971.

[3] Peter L Montgomery. Speeding the pollard and elliptic curve methods of factorization. *Mathematics of computation*, 48(177):243–264, 1987.

[4] Katsuyuki Okeya, Hiroyuki Kurumatani, and Kouichi Sakurai. Elliptic curves with the montgomery-form and their cryptographic applications. In *International Workshop on Public Key Cryptography*, pages 238–257. Springer, 2000.

[5] Harold Edwards. A normal form for elliptic curves. *Bulletin of the American mathematical society*, 44(3):393–422, 2007.

[6] Daniel J Bernstein and Tanja Lange. Faster addition and doubling on elliptic curves. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 29–50. Springer, 2007.

[7] Daniel J Bernstein, Peter Birkner, Marc Joye, Tanja Lange, and Christiane Peters. Twisted edwards curves. In *International Conference on Cryptology in Africa*, pages 389–405. Springer, 2008.

[8] Marc Joye, Mehdi Tibouchi, and Damien Vergnaud. Huffs model for elliptic curves. In *International Algorithmic Number Theory Symposium*, pages 234–250. Springer, 2010.

[9] Hongfeng Wu and Rongquan Feng. Elliptic curves in huffs model. *Wuhan University Journal of Natural Sciences*, 17(6):473–480, 2012.

[10] Daniel J Bernstein, Chitchanok Chuengsatiansup, David Kohel, and Tanja Lange. Twisted hessian curves. In *International Conference on Cryptology and Information Security in Latin America*, pages 269–294. Springer, 2015.

[11] Dustin Moody and Daniel Shumow. Analogues of vélus formulas for isogenies on alternate models of elliptic curves. *Mathematics of Computation*, 85(300):1929–1951, 2016.

[12] Craig Costello and Huseyin Hisil. A simple and compact algorithm for SIDH with arbitrary degree isogenies. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 303–329. Springer, 2017.

[13] Joost Renes. Computing isogenies between montgomery curves using the action of (0, 0). In *The Eighth International Conference on Post-Quantum Cryptography, PQCrypto*, pages 229–247. Springer, 2017.

[14] Tetsuya Izu, Jun Kogure, Masayuki Noro, and Kazuhiro Yokoyama. Efficient implementation of schoofs algorithm. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 66–79. Springer, 1998.

[15] Reynald Lercier and François Morain. Computing isogenies between elliptic curves over _ {} using couveigness algorithm. *Mathematics of Computation*, 69(229):351–370, 2000.

[16] David Jao, Stephen D Miller, and Ramarathnam Venkatesan. Do all elliptic curves of the same order have the same difficulty of discrete log? In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 21–40. Springer, 2005.

[17] Edlyn Teske. An elliptic curve trapdoor system. *Journal of cryptology*, 19(1):115–133, 2006.

[18] Denis X Charles, Kristin E Lauter, and Eyal Z Goren. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22(1):93–113, 2009.

[19] Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.

[20] Nigel P Smart. The hessian form of an elliptic curve. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 118–125. Springer, 2001.

[21] Huseyin Hisil, Gary Carter, and Ed Dawson. New formulae for efficient elliptic curve arithmetic. In *International Conference on Cryptology in India*, pages 138–151. Springer, 2007.

[22] Huseyin Hisil, Kenneth Koon-Ho Wong, Gary Carter, and Ed Dawson. Faster group operations on elliptic curves. In *Proceedings of the Seventh Australasian*

Conference on Information Security-Volume 98, pages 7–20. Australian Computer Society, Inc., 2009.

[23] Marc Joye and Jean-Jacques Quisquater. Hessian elliptic curves and side-channel attacks. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 402–410. Springer, 2001.

[24] Reza R Farashahi and Marc Joye. Efficient arithmetic on hessian curves. In *International Workshop on Public Key Cryptography*, pages 243–260. Springer, 2010.

[25] Reza R Farashahi, Hongfeng Wu, and Chang-An Zhao. Efficient arithmetic on elliptic curves over fields of characteristic three. In *International Conference on Selected Areas in Cryptography*, pages 135–148. Springer, 2012.

[26] David Kohel. The geometry of efficient arithmetic on elliptic curves. *Arithmetic, Geometry, Coding Theory and Cryptography*, 637:95–109, 2015.

[27] Dustin Moody and Hongfeng Wu. Families of elliptic curves with rational 3-torsion. *Journal of Mathematical Cryptology*, 5(3-4):225–246, 2012.

[28] Trond Stølen Gustavsen and Kristian Ranestad. A simple point counting algorithm for hessian elliptic curves in characteristic three. *Applicable Algebra in Engineering, Communication and Computing*, 17(2):141–150, 2006.

[29] William Fulton. *Algebraic curves: An introduction to algebraic geometry.* 2008. `http://www.math.lsa.umich.edu/~wfulton/CurveBook.pdf`.

[30] Olivier Billet and Marc Joye. The jacobi model of an elliptic curve and side-channel analysis. In Marc Fossorier, Tom Høholdt, and Alain Poli, editors, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, pages 34–42, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.

[31] P. Y. Liardet and N. P. Smart. Preventing spa/dpa in ecc systems using the jacobi form. In Çetin K. Koç, David Naccache, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems — CHES 2001*, pages 391–401, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

Computer Security Division, National Institute of Standards and Technology, 100 Bureau Drive, Gaithersburg, MD, 20899-8930, USA.
E-mail address: `thinh.dang@nist.gov`

Computer Security Division, National Institute of Standards and Technology, 100 Bureau Drive, Gaithersburg, MD, 20899-8930, USA.
E-mail address: `dustin.moody@nist.gov`