Superglue: Fast formulae for (2,2)-gluing isogenies

Max Duparc

EPFL, Lausanne, Switzerland
max.duparc@epfl.ch

Abstract. Following Mumford's theory, theta structures on products of elliptic curves are induced by symmetries whose eigenvectors correspond to 4-torsion points on the Kummer line. These symmetries introduce a rich pattern of self-similarities within the theta structure that we exploit to enhance the computation of gluing isogenies. Focusing on the dimension-2 case, we show how theta structures can be computed projectively, thereby avoiding costly modular inversions. Moreover, by leveraging the sparsity of certain specific 4-torsion points and the action of the canonical 2-torsion points in the Kummer line, we derive new formulae for the evaluation of $(2, 2)$-gluing isogenies. These formulae require significantly fewer precomputations and arithmetic operations than previous methods. Additionally, our formulae also support the evaluation of points on the quadratic twist at negligible additional cost, without requiring operations in an extended field.

Keywords: High-dimensional isogenies · Theta structures · Gluing isogenies · Quadratic twist

1 Introduction

The interest for high-dimensional Abelian varieties and the isogenies connecting them has significantly increased following the attacks on SIDH [5,23,31] and the realisation that, through Kani's Lemma [16], one can embed generic-degree isogenies between elliptic curves into smooth degree, high-dimensional[1] isogenies. This insight has led to many new cryptographic constructions, such as, to cite a few, the SQIsignHD constructions [10,2,29,12,1], DeuringVRF [17], QFESTA [28], LIT-SiGamal [24], POKÉ [3], SCALLOP-HD [6], PEGASIS [9]...
All these new cryptographic protocols call for fast and efficient implementations. However, due to the intrinsic mathematical complexity of high-dimensional Abelian varieties, this question has, up until recently, received less attention compared to its elliptic curve counterpart. Currently, one of the most promising approaches relies on the use of *theta structures*, as explored in [15,30,19,21,20,22,8,11]. The latter proposed new formulae to compute $(2, 2)$ isogeny chains between products of elliptic curves. Among the various subroutines it presents, the most

[1] In this paper, high-dimensional refers to dimension 2 and above.

complex and subtle is the computation of $(2,2)$-*gluing isogenies*, i.e. isogenies whose domain are product of elliptic curves and whose codomain are Jacobian of hyperelliptic curves. Compared to generic $(2,2)$-isogenies, gluing isogenies are approximately seven times more expensive and are the only step requiring modular inversion. Moreover, although they are defined on products of Kummer lines, their formulation does not support evaluation on points from the quadratic twist, a property that is desired and used in some cryptographic schemes such as [12,17,3]. Can we therefore remedy those issues and improve gluing isogenies?

Contributions We show in this paper that it is possible. More specifically, we introduce *Superglue*, which are new formulae to efficiently compute $(2,2)$-gluing isogenies based on the key idea of expressing the image of an arbitrary point as a quartic homogeneous combination of the images of the 4-torsion points lying above the points $(0:1)$ in the Kummer lines. These formulae depend on the relative positions of these torsion points within the symplectic basis underlying the theta structure and are often sparse — typically involving only two nonzero coefficients — which yields clear computational advantages. Our contributions include the following:

- We define a representation of symmetric elements that remains homogeneous across all 4-torsion points. Exploiting the algebraic relations among these elements enables the efficient derivation of theta structures without requiring modular inversion.
- We propose a novel representation for the combined addition and subtraction of two points on an elliptic curve. This formulation is not only computationally efficient, but also naturally extends to points on the quadratic twist.
- We derive and present the nine distinct duplication formulae arising from different choices of symplectic bases in the theta structure. We also provide algorithms for efficiently computing them.

Taken together, our methods significantly reduce both the cost of precomputation and the overall number of operations required to evaluate points through $(2,2)$-gluing isogenies compared to previous approaches such as those in [11,1]. Detailed cost comparisons are provided in table 4.

Outline. The remainder of this paper is organized as follows. In section 2, we recall the basics of theta structures and explain their role in computing isogenies. In section 3, we describe how to improve the computation of the theta structure over product of elliptic curves. Section 4 presents more efficient formulae for evaluating points through $(2,2)$-gluing isogenies. In section 5, we detail the algorithms used to implement these new formulae. Finally, in section 6, we explain how our formulae can be naturally adapted to efficiently evaluate points lying on a quadratic twist. Additionally, we provide in appendix A the exhaustive list of all Superglue formulae.

Acknowledgment I would like to give special thanks to Damien Robert for the insightful comments he provided me during the writing of this paper. I would also like to thank the anonymous reviewers of Asiacrypt 2025 for their detailed and thorough reviews that significantly improved this paper's presentation.

2 Background

Throughout this paper, we work over a base finite field $\mathbb{F}_q$ of characteristic p, with $q = p^{2n}$ for some prime p and denote $\mathbf{i} = \sqrt{-1} \in \mathbb{F}_q$ and $\overline{\mathbb{F}_q}$ the algebraic closure of $\mathbb{F}_q$. We denote respectively as $\mathbf{M}_{i,*}$ and $\mathbf{M}_{*,j}$ the i-th line and j-th column of a matrix $\mathbf{M}$. Additionally, we represent as $\odot$ and $\otimes$ the Hadamard and Kronecker products.

2.1 Abelian varieties

Abelian varieties are projective algebraic varieties endowed with a commutative group law. In this work, we focus primarily on Abelian varieties of dimension 1 and 2, namely elliptic curves and Abelian surfaces.

Elliptic curves Elliptic curves are the prototypical examples of Abelian varieties as they are the easiest to understand and work with. We work with Montgomery curves, a specific parametrisation of elliptic curves.

Definition 1 (Montgomery curves). *For any $A \in \mathbb{F}_q$ and $B \in \mathbb{F}_q$ non-quadratic residue, the* Montgomery curve E_A *and its* quadratic twist $E_A^\top$ *are defined as the solution of the equation*

$$E_A : zy^2 = x^3 + Ax^2z + xz^2 = x(x - \alpha z)(x - \alpha^{-1}z)$$
$$E_A^\top : Bzy^2 = x^3 + Ax^2z + xz^2 = x(x - \alpha z)(x - \alpha^{-1}z)$$

The addition formulae for Montgomery curves are detailed in [7,14]. We call C the canonical 2-torsion point $(0 : 0 : 1)$. Notably, though the curves E_A and $E_A^\top$ share the same j-invariant[2], their isomorphism $\eta : (x : y : z) \mapsto (x : \sqrt{B}y : z)$ is only defined over $\mathbb{F}_{q^2}$.
Montgomery curves are particularly advantageous due to their ability to facilitate fast differential arithmetic by omitting the y-coordinate and performing computations in the *Kummer line* [7]. Elliptic curves are genus 1 curves and by the Riemann-Roch theorem [33, Section 2.5], their Jacobians (i.e., the Picard varieties $\mathrm{Pic}^0(E)$) are one-dimensional Abelian variety that are in fact isomorphic to the curve itself, as detailed in [33, Section 3.3].

[2] $j(E_A) = \frac{256(A^2-3)^2}{A^2-4} = j(E_A^\top)$

Abelian sufaces Abelian surfaces[3] generalise elliptic curves to dimension 2. Apart from product of elliptic curves, Abelian surfaces are induced by curves of genus 2. Those *hyperelliptic curves* can be described using the *Rosenhain model*:

Definition 2 (Rosenhain model).
For any $\lambda, \mu, \nu \in \mathbb{F}_q$, the Rosenhain hyperelliptic curve $C_{\lambda,\mu,\nu}$ is given by the solutions of the equation

$$C_{\lambda,\mu,\nu} : z^3y^2 = x(x-z)(x-\lambda z)(x-\mu z)(x-\nu z).$$

We have that $J_{\lambda,\mu,\nu} = \mathrm{Pic}^0(C_{\lambda,\mu,\nu})$, the Jacobian of $C_{\lambda,\mu,\nu}$, is an Abelian surface and following [35,32], any (principally polarized) Abelian surface is isomorphic to either the jacobian of a hyperelliptic curve in the Rosenhain model $J_{\lambda,\mu,\nu}$ or a product of elliptic curves $E_1 \times E_2$. This distinction is not evenly distributed: up to isomorphism, among all Abelian surfaces defined over $\mathbb{F}_q$, approximately $\Theta(q^3)$ correspond to Jacobians of hyperelliptic curves, whereas only $\Theta(q^2)$ correspond to products of elliptic curves. Consequently, for cryptographic field sizes, the proportion of products becomes negligible.

There exist reasonably efficient algorithms to compute the arithmetic and the isogenies on the Jacobians of hyperelliptic curves using Cantor's algorithm [4] and Richelot's equations [34]. However, the main drawback of working with Jacobians is that this involves computations in polynomial function fields, which are inherently complex. Moreover, most known algorithms for Jacobians of genus-2 curves do not generalize nicely to higher dimensions. For these reasons, recent research has focused on an alternative representation of Abelian varieties: the *theta model*.

2.2 Theta structures

The theta model is based on the concept of theta structures, principally developed in the seminal work of Mumford [25,26,27]. They have found practical applications in cryptography following the work of Damien Robert [30,19,21,22,11]. Although naturally described using the language of scheme theory, we present them from a more combinatorial point of view, which we hope, could enable wider understanding and interest.

Symplectic basis Let A be a principally polarized Abelian variety of dimension g. Since there exists a natural isomorphism between A and its dual, there exists a non-degenerate, bilinear, and alternating N-Weil pairing:

$$e_N : A[N] \times A[N] \to \mu_N$$

for all N coprime to q, where $\mu_N = \{x \in \overline{\mathbb{F}_q} | x^N = 1\}$.

[3] More specifically principally polarized Abelian surface, which are Abelian variety A equipped with a natural isomorphism to its dual, $A^\vee = \mathrm{Pic}^0(A)$.

This Weil pairing imposes significant structural constraints on $A[N]$. In particular, it ensures that $A[N]$ is a $\mathbb{Z}_N$-module of rank $2g$. We represent elements of $A[N] \cong \mathbb{Z}_N^{2g}$ in a way that is compatible with the Weil pairing. That is, we define a *symplectic representation*:

$$\pi : A[N] \cong \mathbb{Z}_N^g \times \widehat{\mathbb{Z}_N^g},$$

such that for any points $P, Q \in A[N]$:

$$\pi(P) = (x_P, \widehat{x_P}), \quad \text{where} \quad e_N(P,Q) = \omega^{(\widehat{x_Q}\cdot x_P)-(\widehat{x_P}\cdot x_Q)}.$$

with ω a N-th root of unity and with $\widehat{x_Q}\cdot x_P$ denoting the cross product in $\mathbb{Z}_N^g$.[4] Symplectic representations of $A[N]$ are in fact in a one-to-one correspondence with very specific bases of $A[N]$, known as *symplectic bases*.

Definition 3 (Symplectic basis). *Let $A[N] \cong \mathbb{Z}_N^{2g}$ be the N-torsion subgroup of A. A* symplectic basis *of $A[N]$ is a basis $\{S_1, \ldots, S_g; T_1, \ldots, T_g\}$ such that: $e_N(S_i, S_j) = e_N(T_i, T_j) = 1$, and $e_N(S_i, T_j) = \omega^{\delta_{ij}}$, where ω is a primitive N-th root of unity.*

This equivalence is clear, as:

- Given a symplectic basis $\{S_1, \ldots, S_g, T_1, \ldots, T_g\}$, we can construct a valid structure π by setting: $\pi(S_i) = (\boldsymbol{\delta_i}, 0), \quad \pi(T_i) = (0, \boldsymbol{\delta_i})$ where $(\boldsymbol{\delta_i})_j = \delta_{ij}$.
- Given any valid structure π, the points $S_i = \pi^{-1}(\boldsymbol{\delta_i}, 0)$ and $T_i = \pi^{-1}(0, \boldsymbol{\delta_i})$ define a symplectic basis of $A[N]$.

Using a slight abuse of notation, we denote a symplectic basis as $\langle S_1, \ldots, S_g\rangle \oplus \langle T_1, \ldots, T_g\rangle$.

Theta structures We now restrict ourselves to the case $N = 2$. Using symplectic basis, we define theta structure as follows

Definition 4 (Theta structure). *Let A be a principally polarized Abelian variety of dimension g. A (symmetric)* level-2 theta structure *is a morphism $\theta^A : A/_{\pm 1} \to \mathbb{P}^{2^g-1}$ induced by a symplectic basis of $A[2]$ such that, for any $P \in A$ and $X \in A[2]$, we have that:*

$$\theta_i^A(P + X) = (-1)^{\widehat{x}\cdot i}\theta_{i+x}^A(P),$$

where θ_i^A are the components of θ^A and $\pi(X) = (x, \widehat{x})$, with π the symplectic representation of $A[2]$ induced by the symplectic basis.

[4] We represent elements in $\widehat{\mathbb{Z}_N^g}$ as elements in $\mathbb{Z}_N^g$ instead of characters (group morphisms $\chi : \mathbb{Z}_N^g \to \mu_N$). This is because we make use in this paper of duality that permutes x and $\widehat{x}$. Having them shared the same notation therefore simplifies already heavy notations.

The image of A by θ^A is called the *Kummer variety*, and is denoted $\mathcal{K}_A$. Importantly, for a given symplectic basis over $A[2]$, there exists several valid theta structures. Following Mumford's work, [25, Remark 4, p. 319], this is because theta structures are in fact induced by symplectic basis over $A[4]$.

In the case of Montgomery curves, we construct theta structure using *symmetric elements* $\mathfrak{g}_{T'}$, which are involutions acting on the global sections of the Kummer line[5], mapping T' to itself, with T' a point of 4 torsion in the symplectic basis underlying our theta structure. Following [11, Algorithm 1], for a 4-torsion point $T' = (x : y : z)$ satisfying $[2]T' = (u : 0 : w)$, $\mathfrak{g}_{T'} \in \mathrm{GL}_2(\mathbb{F}_q^2)$ as detailed in eq. (1) is a symmetric element, as $\mathfrak{g}_{T'} \cdot \binom{x}{z} = \binom{x}{z}$ and $\mathfrak{g}_{T'}^2 = \mathfrak{g}_0 = \mathsf{id}$

$$\mathfrak{g}_{T'} = \begin{pmatrix} uz\delta^{-1} & -ux\delta^{-1} + xz^{-1} \\ zw\delta^{-1} & -uz\delta^{-1} \end{pmatrix}, \quad \text{where} \quad \delta = xw - zu. \tag{1}$$

Using the fact that for any point $X = (a : b)$ on the Kummer line, $X + [2]T' = \mathfrak{g}_{T'} \cdot \binom{a}{b}$, we have that the theta structure induced by a symplectic basis $\langle S', T' \rangle = E[4]$ is given by the linear map $\theta^E(X) = \theta^E \cdot \binom{a}{b}$ such that

$$\theta^E = \begin{pmatrix} \theta_0^E \\ \theta_1^E \end{pmatrix} \text{ with } \begin{cases} \theta_i^E \cdot \mathfrak{g}_{T'} = (-1)^i \theta_i^E \\ \theta_i^E \cdot \mathfrak{g}_{S'} = \theta_{i+1}^E \end{cases} \implies \begin{cases} \theta_0^E = [\mathfrak{g}_0 + \mathfrak{g}_{T'}]_{0,*} \\ \theta_1^E = \theta_0^E \cdot \mathfrak{g}_{S'} \end{cases}$$

Example 1. Let E_A be an elliptic curve and let $\langle (u : v) \rangle \oplus \langle (1 : -1) \rangle$ be a symplectic basis of $E_A[4]$ in the Kummer line, the theta structure associate to this symplectic basis is

$$\theta^{E_A}(P) = \big((u+v)(x-z) : (u-v)(x+z)\big)$$

We list in appendix B all possible forms of theta structures on elliptic curves, depending on the choice of (symplectic) basis.

A crucial invariant in this setting is the image of the neutral element, denoted $\theta^A(0)$ and known as the *theta null point*. Notably, the theta null point uniquely determines the Abelian variety A up to isomorphism. This is evident from our previous example, where u, v satisfy $\big((u+v)^2 : (u-v)^2\big) = (\alpha^{\pm 1} + 1 : \alpha^{\pm 1} - 1)$, with α a non-trivial root of the curve equation, allowing us to recover the A coefficient and thus the j-invariant of E_A. Another example can be found in Gaudry's work [15], which establishes the connection between Kummer surfaces and hyperelliptic curves in the Rosenhain model.

Changes of theta structure Because they are induced by symplectic basis, changes of symplectic basis induce changes of theta structure and therefore changes of theta coordinates. Let $\mathcal{B} = \langle S_1, \cdots, S_g \rangle \oplus \langle T_1, \cdots, T_g \rangle$ and

[5] More specifically, the section induced by the x and z coordinates $(x : z) \to (x, z) \in \mathbb{F}_q^2$.

$\widetilde{\mathcal{B}} = \langle \widetilde{S_1}, \cdots, \widetilde{S_g} \rangle \oplus \langle \widetilde{T_1}, \cdots, \widetilde{T_g} \rangle$ be two symplectic basis with $\theta^{\mathcal{B}}, \theta^{\widetilde{\mathcal{B}}}$ the associated theta structures. Then, given $\mathbf{N} \in \mathrm{Sp}_{2g}(\mathbb{Z}_4)$ such that $\widetilde{\mathcal{B}} = \mathbf{N} \cdot \mathcal{B}$, we can compute a change of coordinates matrix $\mathbf{M} \in \mathrm{GL}_{2^g}(\mathbb{F}_q)$ such that $\theta^{\widetilde{B}} = \mathbf{M} \cdot \theta^B$. The exact computations are detailed in [8, Theorem 12]. The most important change of theta structure is given by permuting all S_i and T_i together. In this case, we have that the change of coordinates matrix is in fact the Hadamard matrix. This in fact induces a notion of *duality* and, given θ^A, we define its Hadamard transform $\widetilde{\theta}^A = \mathcal{H}(\theta^A)$ as its *dual*. Duality is central in describing the symmetries of theta structure that are described through the *Riemann relations* [25, p. 334-335] that enables high dimensional differential algebra over Kummer varieties [21,36].

Additionally, theta structures are nicely compatible with products. More specifically, given A_1 and A_2 two Abelian variety of dimension g_1, g_2 with respective theta structure θ^{A_1} and θ^{A_2} induced by the symplectic basis $\langle S_1^1, \cdots, S_{g_1}^1 \rangle \oplus \langle T_1^1, \cdots, T_{g_1}^1 \rangle$ and $\langle S_1^2, \cdots, S_{g_2}^2 \rangle \oplus \langle T_1^2, \cdots, T_{g_2}^2 \rangle$. We can construct a *product theta structure* $\theta^{A_1 \times A_2}$ over $A_1 \times A_2$ defined as:

$$\theta^{A_1 \times A_2}(P_1, P_2) = \sigma(\theta^{A_1}(P_1), \theta^{A_2}(P_2))$$

using σ the Segre embedding[6], this product theta structure been induced by the symplectic basis

$$\langle (S_1^1, 0) \cdots, (S_{g_1}^1, 0), (0, S_1^2) \cdots, (0, S_{g_2}^2) \rangle \oplus \langle (T_1^1, 0) \cdots, (T_{g_1}^1, 0), (0, T_1^2) \cdots, (0, T_{g_2}^2) \rangle$$

2.3 Isogenies and theta structures

Like their elliptic curve counterparts, which they generalise, high-dimensional (polarized) isogenies are also determined by their kernels. In this work, we focus on $(2, \cdots, 2)$-isogenies, which are isogenies generated by *isotropic kernels*, which are subgroups $K \subset A[2]$ such that for all $P, Q \in K$, we have $e_2(P, Q) = 1$.

Due to polarization duality, the kernel of a $(2, \cdots, 2)$-isogeny must be a maximal isotropic subgroup, as following [25, p. 302], the polarization over A naturally extend to the codomain of our isogenies. This establishes a fundamental link between theta structures and isogenies through symplectic basis, making theta structures particularly well-suited for computing isogenies. This relationship is formalized by the *Isogeny Theorem* [25, Theorem 4, p.302], which directly leads to the *Duplication Formula*. [11, Section 2.3] (see also [25, Equation A, p. 332] and [30, Theorem 3.6.4].)

Theorem 1 (Duplication Formula). *Let $K_1 \oplus K_2$ be the symplectic basis of $A[4]$ inducing the theta structure θ^A and let $\Phi : A \to B$ be the associated $(2, \ldots, 2)$-isogeny with $\ker(\Phi) = [2]K_2$. For any θ^B a theta structure over B compatible with Φ, and for all points $P, Q \in A$, we have that:*

$$\mathcal{H}\Big(\theta^A\big(P+Q\big) \odot \theta^A\big(P-Q\big)\Big) = \widetilde{\theta}^B\big(\Phi(P)\big) \odot \widetilde{\theta}^B\big(\Phi(Q)\big).$$

[6] $\sigma\big((x_i)_{1 \le i \le n}, (y_j)_{1 \le j \le m}\big) = (x_i y_j)_{1 \le i,j \le n,m}$ using lexicographical order.

Following [10, Theorem 56], among the possible compatible θ^B, one is in fact induced by $\Phi(K_1) \oplus \Phi(K_2')$, with K_2' a 8-torsion isotropic subgroup such that $[2]K_2' = K_2$.
The duplication formula allows for the efficient computation and evaluation of $(2, \dots, 2)$-isogenies and serves as the underlying mechanism of efficient degree 2 isogeny computation, as detailed in [11,8]. However, there is one particular case where applying the duplication formula becomes significantly more intricate, computationally expensive, and subtle: the computation of *gluing isogenies*. In the two-dimensional case, these are isogenies whose domain is a product of elliptic curves, while their codomain is the Jacobian of a hyperelliptic curve.

Unlike standard $(2, 2)$-isogenies, which can be composed naturally through successive duplications, gluing isogenies involve additional constraints, making their computation more technical. Intuitively, this arises from the fact that a gluing isogeny $\Phi : E_1 \times E_2 \to J_1$ maps points $(\pm P, \pm Q)$ on the product of Kummer lines $E_1/_{\pm 1} \times E_2/_{\pm 1}$ to a point $\pm\Phi(P, Q)$ on the Kummer surface $J/_{\pm 1}$. We therefore have a sign ambiguity to lift, requiring additional information.

2.4 Gluing isogenies

The computation of gluing isogenies $\Phi : E_1 \times E_2 \to J_1$, as detailed in [11, Section 4] and [18, Section 3.2], is decomposed into 3 steps.

Step 1: Computing the theta structure This step consist in efficiently computing a theta structure over $E_1 \times E_2$ such that the isogeny Φ that we desire to compute is consistent with our theta structure, meaning that $\ker(\Phi) = \langle T_1, T_2 \rangle = [2]K_2$, where $K_1 \oplus K_2$ is the symplectic basis inducing our theta structure. To do so, we compute the change of coordinates matrix $\mathbf{M} \in \mathrm{GL}_4(\mathbb{F}_q)$ that satisfies the relation

$$\theta^{E_1 \times E_2}(P) = \mathbf{M} \cdot (P_1 \otimes P_2) = \mathbf{M} \cdot \begin{pmatrix} x_1 x_2 \\ x_1 z_2 \\ z_1 x_2 \\ z_1 z_2 \end{pmatrix}$$

for any point $P = (P_1, P_2) \in E_1 \times E_2$, where $\theta^{E_1 \times E_2} : E_1 \times E_2 \to \mathbb{P}^3$ is a level-2 theta structure induced by the symplectic basis

$$\langle S_1', S_2' \rangle \oplus \langle T_1', T_2' \rangle = \langle (0, T_{2,2}'), (T_{1,1}', 0) \rangle \oplus \langle (T_{1,1}', T_{1,2}'), (T_{2,1}', T_{2,2}') \rangle.$$

with $\langle T_{1,1}, T_{2,1} \rangle = E_1[4]$ and $\langle T_{1,2}, T_{2,2} \rangle = E_2[4]$ and such that $\ker(\Phi) = \langle T_1, T_2 \rangle = \big\langle (T_{1,1}, T_{1,2}), (T_{2,1}, T_{2,2}) \big\rangle$, with $T_i = [2]T_i'$, and $S_i = [2]S_i'$.

As detailed in [11, Section 2.3], the matrix $\mathbf{M}$ is efficiently derived from the *symmetric elements* of the generators of the symplectic basis underlying our theta structure. Specifically, $\mathbf{M}$ is given by

$$\begin{array}{l|l} \mathbf{M}_{0,*} = \Big[\Big(\mathfrak{g}_0 + \mathfrak{g}_{T_1'} \Big) \Big(\mathfrak{g}_0 + \mathfrak{g}_{T_2'} \Big) \Big]_{0,*} & \mathbf{M}_{2,*} = \mathbf{M}_{0,*} \cdot \mathfrak{g}_{S_2'} \\ \mathbf{M}_{1,*} = \mathbf{M}_{0,*} \cdot \mathfrak{g}_{S_1'} & \mathbf{M}_{3,*} = \mathbf{M}_{1,*} \cdot \mathfrak{g}_{S_2'} \end{array}$$

where $\mathfrak{g}_P = \mathfrak{g}_{P_1} \otimes \mathfrak{g}_{P_2}$. Computing $\mathbf{M}$ thus only requires computing $\mathfrak{g}_{T'_{1,1}}$, $\mathfrak{g}_{T'_{1,2}}$, $\mathfrak{g}_{T'_{2,1}}$, and $\mathfrak{g}_{T'_{2,2}}$.

Remark 1. The matrix $\mathbf{M}$ is not a change of coordinates matrix in the sense of section 2.2, as $P_1 \otimes P_2$ is not a product theta structure. Instead, it corresponds to the following sequence of operations: first constructing separate theta structures over E_1 and E_2, forming a product theta structure, and finally applying the right change of coordinates matrix using [8, Theorem 12].

Since we are working with a theta structure over a product of elliptic curves, it follows from [13, Proposition 6.5] that there exist $(x, \widehat{x}) \in \mathbb{Z}_2^2 \times \widehat{\mathbb{Z}_2^2}$ with $x \cdot \widehat{x} = 0$ such that

$$\sum_{i \in \mathbb{Z}_2^2} (-1)^{\widehat{x} \cdot i} \theta_i^{E_1 \times E_2}(0) \theta_{i+x}^{E_1 \times E_2}(0) = 0.$$

Due to our choice of theta structure, this couple is actually $(x, \widehat{x}) = (00, 11)$.[7]

This leads to the key implication:

$$\widetilde{\theta}_{11}^{J_1}(0)^2 = \sum_{i \in \mathbb{Z}_2^2} (-1)^{11 \cdot i} \theta_i^{E_1 \times E_2}(0)^2 = 0.$$

Thus, the last components of the dual theta-null point of J_1, the codomain of Φ, is zero, which is the key property that renders gluing evaluation complex.

Step 2: Computing the codomain Once our theta structure is fixed, we compute the codomain (the theta null point) of our gluing isogeny Φ. To do so, we use 8-torsion isotropic points. Let T_1'' and T_2'' be such torsion points lying above K_2 and let $(\alpha : \beta : \gamma : 0) = \widetilde{\theta}^{J_1}(0)$. Using the duplication formula, we have that

$$\mathcal{H}\Big(\theta^{E_1 \times E_2}(T_i'') \odot \theta^{E_1 \times E_2}(T_i'')\Big) = \widetilde{\theta}^{J_1}(\Phi(T_i'')) \odot \widetilde{\theta}^{J_1}(0)$$

with $\Phi(T_i'') \in J_1[4]$. Using the matrix $\mathbf{M}$, the left-hand side of these equations is computable. For the right-hand side, as we choose the theta structure θ^{J_1} to be induced by the symplectic basis $\Phi(K_1) \oplus \Phi(K_2')$, we have that $\widetilde{\theta}_i^{J_1}(\Phi(T_1'')) = \widetilde{\theta}_{i+01}^{J_1}(\Phi(T_1''))$ and $\widetilde{\theta}_i^{J_1}(\Phi(T_2'')) = \widetilde{\theta}_{i+10}^{J_1}(\Phi(T_2''))$. The right-hand side of our duplication formulas are of the form

$$(\alpha x : \beta x : \gamma y : 0) \text{ for } i = 1, \quad (\alpha z : \beta w : \gamma z : 0) \text{ for } i = 2.$$

We can therefore retrieve projectively our dual theta null point as simply

$$(\alpha : \beta : \gamma : 0) = \big((\alpha x) \cdot (\alpha z) : (\beta x) \cdot (\alpha z) : (\gamma z) \cdot (\alpha x) : 0\big)$$

and retrieve $\theta^{J_1}(0)$ using the Hadamard transform.

[7] This follows from [13, Proposition 6.5], where it is shown that $(x, \widehat{x}) = (11, 11)$ when working with a product theta structure. Applying [8, Theorem 12] yields the desired result.

Step 3: Evaluation Now, to evaluate a point $P \in E_1 \times E_2$, we use the following Duplication Formula

$$\mathcal{H}\Big(\theta^{E_1\times E_2}(P) \odot \theta^{E_1\times E_2}(P)\Big) = \widetilde{\theta}^{J_1}(\Phi(P)) \odot \widetilde{\theta}^{J_1}(0)$$

Due to our theta structure on $E_1 \times E_2$, $\widetilde{\theta}^{J_1}_{11}(0)$ is zero, and we can therefore only recover 3 components of $\widetilde{\theta}^{J_1}(\Phi(P))$, by multiplying by the projective inverse $\alpha^{-1}, \beta^{-1}, \gamma^{-1}$. To retrieve the last component, we use the duplication formula a second time:

$$\mathcal{H}\Big(\theta^{E_1\times E_2}(P+T_1') \odot \theta^{E_1\times E_2}(P+T_1')\Big) = \widetilde{\theta}^{J_1}(\Phi(P+T_1')) \odot \widetilde{\theta}^{J_1}(0)$$

with T_1' the 4-torsion above T_1. we have that

$$\widetilde{\theta}^{J_1}_i(\Phi(P+T_1')) = \widetilde{\theta}^{J_1}_i(\Phi(P)+\Phi(T_1')) = \widetilde{\theta}^{J_1}_{i+01}(\Phi(P))$$

We can therefore recover all components of $\widetilde{\theta}^{J_1}(\Phi(P))$. Note that recovering the last component requires some scaling, as we are working with projective coordinates.

3 Projectively computing the change of basis matrix

We see that the modular inversion exclusively appears in eq. (1) as it requires computing the inverse of δ and z. As it is a costly operation, our goal in this section is therefore to develop an approach that avoids these inversions, improving computational efficiency.

3.1 Projective symmetric elements

The need for modular inversion in computing symmetric elements in fact arises from the fact that we take two projective points as input, which must be rescaled accordingly. However, we can compute symmetric elements $\mathfrak{g}_T$ more efficiently by changing the representation of 4-torsion points. Indeed, on the Kummer line, the structure of $E[4]/_{\pm 1}$ is described in fig. 1.

Using this representation, we have that for $T' = (a+b : a-b)$, $\delta_{T'} = 2ab(a-b)$ and

$$\mathfrak{g}_{T'} = \begin{pmatrix} \frac{(a^2+b^2)(a-b)}{2ab(a-b)} & -\frac{(a+b)(a^2+b^2)}{2ab(a-b)} + \frac{(a+b)}{(a-b)} \\ \frac{(a^2-b^2)(a-b)}{2ab(a-b)} & -\frac{(a^2+b^2)(a-b)}{2ab(a-b)} \end{pmatrix} = \frac{1}{2ab}\begin{pmatrix} (a^2+b^2) & -(a^2-b^2) \\ (a^2-b^2) & -(a^2+b^2) \end{pmatrix}$$

Which can be written using the coordinates $T' = (x : z)$ as:

$$\mathfrak{g}_{T'} = \frac{1}{(x^2-z^2)}\begin{pmatrix} (x^2+z^2) & -2xz \\ 2xz & -(x^2+z^2) \end{pmatrix}$$

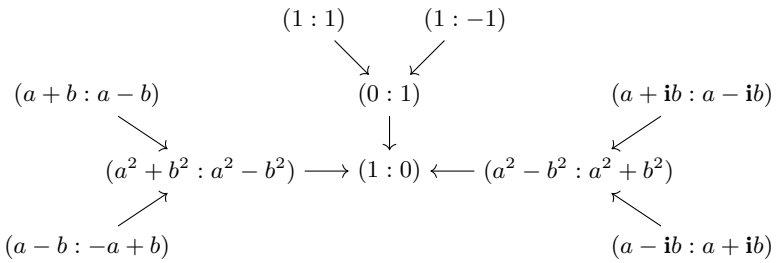

Fig. 1. Structure of $E[4]$ over the Kummer line. Arrows represent the doubling operation.

$$\mathfrak{g}_{(1:1)} = \tfrac{1}{2ab}\begin{pmatrix} 0 & 2ab \\ 2ab & 0 \end{pmatrix} \qquad \mathfrak{g}_{(1:-1)} = \tfrac{1}{2ab}\begin{pmatrix} 0 & -2ab \\ -2ab & 0 \end{pmatrix}$$

$$\mathfrak{g}_{(a+b:a-b)} = \tfrac{1}{2ab}\begin{pmatrix} (a^2+b^2) & -(a^2-b^2) \\ (a^2-b^2) & -(a^2+b^2) \end{pmatrix} \qquad \mathfrak{g}_{(a-b:a+b)} = \tfrac{1}{2ab}\begin{pmatrix} -(a^2+b^2) & (a^2-b^2) \\ -(a^2-b^2) & (a^2+b^2) \end{pmatrix}$$

$$\mathfrak{g}_{(a+\mathbf{i}b:a-\mathbf{i}b)} = \tfrac{1}{2ab}\begin{pmatrix} -\mathbf{i}(a^2-b^2) & \mathbf{i}(a^2+b^2) \\ -\mathbf{i}(a^2+b^2) & \mathbf{i}(a^2-b^2) \end{pmatrix} \qquad \mathfrak{g}_{(a-\mathbf{i}b:a+\mathbf{i}b)} = \tfrac{1}{2ab}\begin{pmatrix} \mathbf{i}(a^2-b^2) & -\mathbf{i}(a^2+b^2) \\ \mathbf{i}(a^2+b^2) & -\mathbf{i}(a^2-b^2) \end{pmatrix}$$

Table 1. Detailed list of the symmetric element of each point in $E[4]$.

With this notation, we can efficiently compute all the symmetric elements, as detailed in table 1. We in fact see that symmetric elements over elliptic curves are derived from the X, Y, Z Pauli matrices as they can be written as

$$\begin{aligned} \mathfrak{g}_{(1:\pm 1)} &= \pm X \\ \mathfrak{g}_{(a\pm b:a\mp b)} &= \pm\frac{1}{2ab}\big((a^2+b^2)Z - \mathbf{i}(a^2-b^2)Y\big) \\ \mathfrak{g}_{(a\pm \mathbf{i}b:a\mp \mathbf{i}b)} &= \mp\frac{1}{2ab}\big(\mathbf{i}(a^2-b^2)Z + (a^2+b^2)Y\big) \end{aligned}$$

and inherit their structure, namely for $P' \neq Q' \in E[4]$ such that $[2]P' \neq [2]Q'$, we have that

$$\mathfrak{g}_{P'} \cdot \mathfrak{g}_{Q'} = -\mathfrak{g}_{Q'} \cdot \mathfrak{g}_{P'}, \text{ and } \mathfrak{g}_{P'} \cdot \mathfrak{g}_{Q'} = \pm\mathbf{i}\mathfrak{g}_{P'+Q'}$$

The formulae of table 1 for symmetric element are practical to compute, as the only required computations are $2ab$ and $\pm(a^2 \pm b^2)$, which involve just two squarings and one multiplication. The remaining operations – addition, sign flips, and multiplication by $\mathbf{i}$ – are relatively inexpensive. Moreover, this approach eliminates the need for computing z^{-1}. Furthermore, the remaining factor is fixed for all points and is induced by the curve they are defined over. This motivates the following definition:

Definition 5 (Projective Symmetric Element). *Given $T \in E[4]$, we define a* projective symmetric element *as a matrix $\widehat{\mathfrak{g}_T}$ such that $\widehat{\mathfrak{g}_T} = \lambda\mathfrak{g}_T$ for some scalar $\lambda \in \mathbb{F}_q^*$. Additionally, given two points $T_1, T_2 \in E[4]$, we say that their respective projective symmetric elements $\widehat{\mathfrak{g}_{T_1}}, \widehat{\mathfrak{g}_{T_2}}$ are* coherent *if they share the same scaling factor.*

Using table 3, given T_1', T_2' a basis of $E[4]$ we can efficiently compute coherent projective symmetric elements $\widehat{\mathfrak{g}_{T_1'}}, \widehat{\mathfrak{g}_{T_2'}}$, and $\widehat{\mathfrak{g}_{T_{1+2}'}}$, all sharing the same scalar λ_E, where $\mathbf{i}\mathfrak{g}_{T_{1+2}'} = \mathfrak{g}_{T_1'} \cdot \mathfrak{g}_{T_2'}$. This computation is performed by algorithm 1.

Algorithm 1 CoherentSymmetricElement(T_1', T_2')

Require: $\langle T_1', T_2' \rangle$ a basis of $E[4]$ with $T_1' = (x_1 : z_1)$ and $T_2' = (x_2 : z_2)$

Ensure: $\widehat{\mathfrak{g}_{T_1}}, \widehat{\mathfrak{g}_{T_2}}$ and $\widehat{\mathfrak{g}_{T_{1+2}}}$ the coherent projective symmetric element with scaling factor λ.

Compute $\mathsf{Case} = \begin{cases} 1 & \text{if } x_1 = z_1 \\ 2 & \text{if } x_1 = -z_1 \\ 3 & \text{if } x_2 = z_2 \\ 4 & \text{if } x_2 = -z_2 \\ 5 & \text{if } \mathbf{i}(x_1 + z_1)(x_2 - z_2) = -(x_1 - z_1)(x_2 + z_2) \\ 6 & \text{if } \mathbf{i}(x_1 + z_1)(x_2 - z_2) = (x_1 - z_1)(x_2 + z_2) \end{cases}$

$\mathsf{Case} = 1$:
Set $\lambda = x_2^2 - z_2^2$ and **return**
$\begin{pmatrix} 0 & \lambda \\ \lambda & 0 \end{pmatrix}, \begin{pmatrix} (x_2^2 + z_2^2) & -2x_2z_2 \\ 2x_2z_2 & -(x_2^2 + z_2^2) \end{pmatrix}, \begin{pmatrix} \mathbf{i}2x_2z_2 & -\mathbf{i}(x_2^2 + z_2^2) \\ \mathbf{i}(x_2^2 + z_2^2) & -\mathbf{i}2x_2z_2 \end{pmatrix}, \lambda$

$\mathsf{Case} = 2$:
Set $\lambda = x_2^2 - z_2^2$ and **return**
$\begin{pmatrix} 0 & -\lambda \\ -\lambda & 0 \end{pmatrix}, \begin{pmatrix} (x_2^2 + z_2^2) & -2x_2z_2 \\ 2x_2z_2 & -(x_2^2 + z_2^2) \end{pmatrix}, \begin{pmatrix} -\mathbf{i}2x_2z_2 & \mathbf{i}(x_2^2 + z_2^2) \\ -\mathbf{i}(x_2^2 + z_2^2) & \mathbf{i}2x_2z_2 \end{pmatrix}, \lambda$

$\mathsf{Case} = 3$:
Set $\lambda = x_1^2 - z_1^2$ and **return**
$\begin{pmatrix} (x_1^2 + z_1^2) & -2x_1z_1 \\ 2x_1z_1 & -(x_1^2 + z_1^2) \end{pmatrix}, \begin{pmatrix} 0 & \lambda \\ \lambda & 0 \end{pmatrix}, \begin{pmatrix} -\mathbf{i}2x_1z_1 & \mathbf{i}(x_1^2 + z_1^2) \\ -\mathbf{i}(x_1^2 + z_1^2) & \mathbf{i}2x_1z_1 \end{pmatrix}, \lambda$

$\mathsf{Case} = 4$:
Set $\lambda = x_1^2 - z_1^2$ and **return**
$\begin{pmatrix} (x_1^2 + z_1^2) & -2x_1z_1 \\ 2x_1z_1 & -(x_1^2 + z_1^2) \end{pmatrix}, \begin{pmatrix} 0 & -\lambda \\ -\lambda & 0 \end{pmatrix}, \begin{pmatrix} \mathbf{i}2x_1z_1 & -\mathbf{i}(x_1^2 + z_1^2) \\ \mathbf{i}(x_1^2 + z_1^2) & -\mathbf{i}2x_1z_1 \end{pmatrix}, \lambda$

$\mathsf{Case} = 5$:
Set $\lambda = x_1^2 - z_1^2$ and **return**
$\begin{pmatrix} (x_1^2 + z_1^2) & -2x_1z_1 \\ 2x_1z_1 & -(x_1^2 + z_1^2) \end{pmatrix}, \begin{pmatrix} -\mathbf{i}2x_1z_1 & \mathbf{i}(x_1^2 + z_1^2) \\ -\mathbf{i}(x_1^2 + z_1^2) & \mathbf{i}2x_1z_1 \end{pmatrix} \begin{pmatrix} 0 & -\lambda \\ -\lambda & 0 \end{pmatrix}, \lambda$

$\mathsf{Case} = 6$:
Set $\lambda = x_1^2 - z_1^2$ and **return**
$\begin{pmatrix} (x_1^2 + z_1^2) & -2x_1z_1 \\ 2x_1z_1 & -(x_1^2 + z_1^2) \end{pmatrix}, \begin{pmatrix} \mathbf{i}2x_1z_1 & -\mathbf{i}(x_1^2 + z_1^2) \\ \mathbf{i}(x_1^2 + z_1^2) & -\mathbf{i}2x_1z_1 \end{pmatrix}, \begin{pmatrix} 0 & \lambda \\ \lambda & 0 \end{pmatrix}, \lambda$

▷ Total cost: $4\mathbf{M} + 2\mathbf{S} + 7\mathbf{a} + 5\mathbf{i}$

3.2 Retrieving theta structure from coherent projective symmetric elements

Using algorithm 1 on $\langle T'_{1,1}, T'_{2,1}\rangle = E_1[4]$ and $\langle T'_{1,2}, T'_{2,2}\rangle = E_2[4]$, we efficiently retrieve

$$\widehat{\mathfrak{g}_{T'_{1,1}}},\quad \widehat{\mathfrak{g}_{T'_{2,1}}},\quad \widehat{\mathfrak{g}_{T'_{1+2,1}}},\quad \widehat{\mathfrak{g}_{T'_{1,2}}},\quad \widehat{\mathfrak{g}_{T'_{2,2}}},\quad \widehat{\mathfrak{g}_{T'_{1+2,2}}}$$

along with their respective scaling factors λ_{E_1} and λ_{E_2}. The key insight that allows us to recover, using our coherent projective symmetric elements, the transformation matrix $\mathbf{M}$ is the fact that $\mathbf{M}$ fundamentally belongs to $\mathrm{PGL}_4(\mathbb{F}_q)$, i.e. it can be computed up to a scalar factor. Using our coherent projective symmetric element, we compute the matrix $\widehat{\mathbf{M}}$:

$$\begin{aligned}
\widehat{\mathbf{M}}_{0,*} &= \left[\lambda_{E_1}\lambda_{E_2}\mathfrak{g}_0 + \widehat{\mathfrak{g}_{T'_1}} + \widehat{\mathfrak{g}_{T'_2}} - \widehat{\mathfrak{g}_{T'_{1+2}}}\right]_{0,-} \\
&= \left[\lambda_{E_1}\lambda_{E_2}\mathfrak{g}_0 + (\widehat{\mathfrak{g}_{T'_{1,1}}} \otimes \widehat{\mathfrak{g}_{T'_{1,2}}}) + (\widehat{\mathfrak{g}_{T'_{2,1}}} \otimes \widehat{\mathfrak{g}_{T'_{2,2}}}) - (\widehat{\mathfrak{g}_{T'_{1+2,1}}} \otimes \widehat{\mathfrak{g}_{T'_{1+2,2}}})\right]_{0,-} \\
\widehat{\mathbf{M}}_{1,*} &= \widehat{\mathbf{M}}_{0,*} \cdot \widehat{\mathfrak{g}_{S'_1}} = \widehat{\mathbf{M}}_{0,*} \cdot (\mathfrak{g}_0 \otimes \widehat{\mathfrak{g}_{T'_{2,2}}}) \\
\widehat{\mathbf{M}}_{2,*} &= \widehat{\mathbf{M}}_{0,*} \cdot \widehat{\mathfrak{g}_{S'_2}} = \widehat{\mathbf{M}}_{0,*} \cdot (\widehat{\mathfrak{g}_{T'_{1,1}}} \otimes \mathfrak{g}_0) \\
\widehat{\mathbf{M}}_{3,*} &= \widehat{\mathbf{M}}_{0,*} \cdot \widehat{\mathfrak{g}_{S'_{1+2}}} = \widehat{\mathbf{M}}_{1,*} \cdot (\widehat{\mathfrak{g}_{T'_{1,1}}} \otimes \mathfrak{g}_0)
\end{aligned}$$

We then compute the matrix $\mathbf{M}$ (in $\mathrm{PGL}_4(\mathbb{F}_q)$) as

$$\mathbf{M} = \begin{pmatrix} \lambda_{E_1}\lambda_{E_2} & 0 & 0 & 0 \\ 0 & \lambda_{E_1} & 0 & 0 \\ 0 & 0 & \lambda_{E_2} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \widehat{\mathbf{M}}$$

Used in such a way, this method allows us to compute the change-of-basis matrix from the symmetric elements projectively, without requiring modular inversion.

4 Improving the duplication formula for gluing isogenies

We now turn our attention to improving the computation of the duplication formula:

$$\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q) \odot \theta^{E_1\times E_2}(P-Q)\Big) = \widetilde{\theta}^{J_1}(\Phi(P)) \odot \widetilde{\theta}^{J_1}(\Phi(Q)). \tag{2}$$

A key observation is that, provided $\widetilde{\theta}^{J_1}(\Phi(Q))$ has no zero coefficients, one can use this equation to compute $\widetilde{\theta}^{J_1}(\Phi(P))$ without needing two separate computation of the duplication formula, as done in section 2.4, which require projective realignment. An optimal choice for such a point Q is T''_1, which is already computed as part of the codomain construction and is guaranteed to have no zero coefficients.

However, a naive evaluation of eq. (2) remains less efficient than the method described in section 2.4. To close this performance gap, we must introduce additional tools and refinements.

4.1 Barycentric coordinate representation of elliptic points

To efficiently work the combined sum and difference of points, we define the notion of *barycentric coordinates*.

Definition 6 (Barycentric coordinates). *Let $P \neq \pm Q$ be points of E. The* barycentric coordinates *of the couple P, Q is a point $(u : v : w) \in \mathbb{P}^2$ such that, in the Kummer line, $P + Q = (x_\oplus : z_\oplus)$ and $P - Q = (x_\ominus : z_\ominus)$ are of the form*

$$\begin{cases} x_\oplus = u - v \\ x_\ominus = u + v \\ z_\oplus = z_\ominus = w \end{cases}$$

We denote the barycentric coordinates of P, Q as $P \pm Q$.

As it is defined in the Kummer line, we have that $P \pm Q = Q \pm P$ and that we can represent any point $P \in E$ in barycentric coordinates by simply setting $Q = 0$, in which case $P \pm 0 = (x_P : 0 : z_P)$. There are different methods to compute the barycentric coordinates $P \pm Q$. Namely:

- Given the (x, z)-coordinates of $P + Q$ and $P - Q$, we can compute $P \pm Q$ as

$$u = x_\ominus z_\oplus + x_\oplus z_\ominus \quad v = x_\ominus z_\oplus - x_\oplus z_\ominus \quad w = 2 z_\ominus z_\oplus$$

- Using the differential addition formulae [7], given P, Q and $P - Q$, we can recover $P \pm Q$ as

$$u = (\mu x_\ominus)^2 + (\lambda z_\ominus)^2 \quad v = (\mu x_\ominus)^2 - (\lambda z_\ominus)^2 \quad w = 2 z_\ominus x_\ominus \mu^2$$

  with $\lambda = (x_P - z_P)(x_Q + z_Q) + (x_P + z_P)(x_Q - z_Q)$ and $\mu = (x_P - z_P)(x_Q + z_Q) - (x_P + z_P)(x_Q - z_Q)$. This operation can be efficiently performed in $6\mathbf{M}, 2\mathbf{S}$ and $9\mathbf{a}$.
- Finally, those components can be efficiently retrieved from the full $(x : y : z)$-projective Montgomery group law.

Lemma 1. *Let $P \neq \pm Q \in E$ be points, with $P = (x_P : y_P : z_P), Q = (x_Q : y_Q : z_Q)$ defined over $E(\mathbb{F}_q)$. We can compute $P \pm Q$ as*

$$\begin{cases} u = \left(z_P^2 y_Q^2 + y_P^2 z_Q^2\right) z_P z_Q - (A z_P z_Q + x_P z_Q + z_P x_Q) \Gamma^2 \\ v = 2 y_P y_Q z_P^2 z_Q^2 \\ w = \Gamma^2 z_P z_Q \end{cases}$$

with $\Gamma = (x_P z_Q - z_P x_Q)$.

This lemma enables to efficiently compute, given any P, Q the barycentric components $P \pm Q$. This is performed by algorithm 2.

Algorithm 2 Barycentric-coordinates(P, Q)

Require: Distinct points $P = (x_P : y_P : z_P)$, $Q = (x_Q : y_Q : z_Q)$ of the montgomery curve E_A.
Ensure: $(u : v : w) = P \pm Q$ the barycentric coordinates of P and Q

$t_1 \leftarrow z_P \cdot z_Q$
$t_2 \leftarrow x_P \cdot z_Q$
$t_3 \leftarrow z_P \cdot x_Q$
$t_4 \leftarrow y_P \cdot z_Q$
$t_5 \leftarrow z_P \cdot y_Q$
$t_6 \leftarrow t_4 \cdot t_5$
$t_6 \leftarrow t_6 \cdot t_1$
$v \leftarrow t_6 + t_6$
$t_7 \leftarrow A \cdot t_1$
$t_2 \leftarrow t_2 + t_3$
$t_7 \leftarrow t_7 + t_2$
$t_3 \leftarrow t_3 + t_3$
$t_2 \leftarrow t_2 - t_3$
$t_2 \leftarrow t_2^2$
$t_7 \leftarrow t_7 \cdot t_2$
$t_4 \leftarrow t_4^2$
$t_5 \leftarrow t_5^2$
$t_4 \leftarrow t_4 + t_5$
$t_4 \leftarrow t_4 \cdot t_1$
$u \leftarrow t_4 - t_7$
$w \leftarrow t_2 \cdot t_1$
return u, v, w ▷ Total cost: $11\mathbf{M} + 3\mathbf{S} + 7\mathbf{a}$

4.2 From barycentric coordinates to theta structure

Barycentric coordinates are nicely compatible with alternating product in the theta structure.

Lemma 2. *Let $P, Q \in E_1 \times E_2$ with $P = (P_1, P_2)$ and $Q = (Q_1, Q_2)$. Let $P_1 \pm Q_1 = (u_1 : v_1 : w_1)$ and $P_2 \pm Q_2 = (u_2 : v_2 : w_2)$ be barycentric coordinates. Let $\mathbf{M} \in GL_4(\mathbb{F}_q)$ such that $\theta^{E_1 \times E_2}(X) = \mathbf{M} \cdot (X_1 \otimes X_2)$.*

$$\theta^{E_1 \times E_2}(P + Q) \odot \theta^{E_1 \times E_2}(P - Q) = \left(\mathbf{M}\boldsymbol{u} + \mathbf{M}\boldsymbol{w}\right)^{\odot 2} - \left(\mathbf{M}\boldsymbol{v_1} + \mathbf{M}\boldsymbol{v_2}\right)^{\odot 2}$$

with

$$\boldsymbol{u} = \begin{pmatrix} u_1u_2 \\ u_1w_2 \\ w_1u_2 \\ w_1w_2 \end{pmatrix} \quad \boldsymbol{w} = \begin{pmatrix} v_1v_2 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \boldsymbol{v_1} = \begin{pmatrix} v_1u_2 \\ v_1w_2 \\ 0 \\ 0 \end{pmatrix} \quad \boldsymbol{v_2} = \begin{pmatrix} u_1v_2 \\ 0 \\ w_1v_2 \\ 0 \end{pmatrix}$$

Proof. We first examine the Kronecker product $(P_1 + Q_1) \otimes (P_2 + Q_2)$. Using the addition components, it can be written as:

$$(P_1 + Q_1) \otimes (P_2 + Q_2) = (u_1 - v_1, w_1) \otimes (u_2 - v_2, w_2)$$

$$\begin{aligned}&= \big((u_1, w_1) - (v_1, 0)\big) \otimes \big((u_2, w_2) - (v_2, 0)\big)\\ &= \boldsymbol{u} + \boldsymbol{w} - \boldsymbol{v_1} - \boldsymbol{v_2}\end{aligned}$$

Applying the $\mathbf{M}$ matrix, we get that

$$\theta^{E_1 \times E_2}(P + Q) = \mathbf{M}\boldsymbol{u} + \mathbf{M}\boldsymbol{w} - \mathbf{M}\boldsymbol{v_1} - \mathbf{M}\boldsymbol{v_2}$$

As $P - Q$ just flips the sign of v_1 and v_2, we get that

$$\theta^{E_1 \times E_2}(P - Q) = \mathbf{M}\boldsymbol{u} + \mathbf{M}\boldsymbol{w} + \mathbf{M}\boldsymbol{v_1} + \mathbf{M}\boldsymbol{v_2}$$

Applying the Hadamard product finishes the proof. □

By adding together $\boldsymbol{u} + \boldsymbol{w}$ and $\boldsymbol{v_1} + \boldsymbol{v_2}$, we already retrieve more efficient formulae than those described in section 2.4 and which have been successfully implemented for the gluing isogenies detailed in [1, Section 8]. Nevertheless, we can do better ! We now rewrite lemma 2 as follows.

Corollary 1. *Using the same notation as lemma 2, we have that*

$$\begin{aligned}\theta^{E_1 \times E_2}(P + Q) \odot \theta^{E_1 \times E_2}(P - Q) = {} & [\mathbf{M}_0\mathbf{M}_0](u_1^2 - v_1^2)(u_2^2 - v_2^2) + [\mathbf{M}_1\mathbf{M}_1]w_2^2(u_1^2 - v_1^2)\\ & + [\mathbf{M}_2\mathbf{M}_2]w_1^2(u_2^2 - v_2^2) + [\mathbf{M}_3\mathbf{M}_3]w_1^2w_2^2\\ & + 2[\mathbf{M}_0\mathbf{M}_1]u_2w_2(u_1^2 - v_1^2) + 2[\mathbf{M}_2\mathbf{M}_3]u_2w_2w_1^2\\ & + 2[\mathbf{M}_0\mathbf{M}_2]u_1w_1(u_2^2 - v_2^2) + 2[\mathbf{M}_1\mathbf{M}_3]u_1w_1w_2^2\\ & + 2[\mathbf{M}_0\mathbf{M}_3 + \mathbf{M}_1\mathbf{M}_2]u_1u_2w_1w_2\\ & + 2[\mathbf{M}_0\mathbf{M}_3 - \mathbf{M}_1\mathbf{M}_2]v_1v_2w_1w_2\end{aligned}$$

with $[\mathbf{M}_i\mathbf{M}_j]$ *the Hadamard product of the* i^{th} *and* j^{th} *columns of* $\mathbf{M}$.

Corollary 1 induces that we can express $\theta^{E_1 \times E_2}(P + Q) \odot \theta^{E_1 \times E_2}(P - Q)$ as a homogeneous quartic combination of the barycentric coordinates of $P_1 \pm Q_1$ and $P_2 \pm Q_2$, scaled by all couples of columns of $\mathbf{M}$. Using $C = (0 : 1)$ the canonical 2-torsion point defined on all Montgomery curves, we have that all couples of columns $\mathbf{M}_i\mathbf{M}_j$ correspond to the Hadamard product of all possible $\theta^{E_1 \times E_2}\big((X, Y)\big)$ with $X, Y \in \{0, C\}$. We therefore have, following theorem 1 that the Hadamard transform of all couples of columns $\widetilde{\mathbf{M}_i\mathbf{M}_j}$ correspond to points in the dual theta structure of the codomain, all detailed in table 2.

We see that the Hadamard transform of the equality described in corollary 1 is in fact induced by only 4 squares of dual theta points of at most 4-torsion, namely $(\widetilde{\theta}^{J_1}(\Phi(0,0)))^{\odot 2}$, $(\widetilde{\theta}^{J_1}(\Phi(0,C')))^{\odot 2}$, $(\widetilde{\theta}^{J_1}(\Phi(C',0)))^{\odot 2}$ and $(\widetilde{\theta}^{J_1}(\Phi(C',C')))^{\odot 2}$ with the rest being derivable from the action of the 2-torsion points $(0, C)$ and $(C, 0)$ on the theta structure, itself induced by their respective position in the symplectic basis, and thus their position in $\ker(\Phi)$. Table 3 describes in which *position* $(C, 0)$ and $(0, C)$ are in the symplectic basis induced by $\ker(\Phi)$.

columns	theta points		columns	dual theta points
$\mathbf{M_0M_0}$	$\theta^{E_1\times E_2}(0,0)\theta^{E_1\times E_2}(0,0)$	$\iff$	$\widetilde{\mathbf{M_0M_0}}$	$\widetilde{\theta}^{J_1}(\Phi(0,0))\widetilde{\theta}^{J_1}(\Phi(0,0))$
$\mathbf{M_1M_1}$	$\theta^{E_1\times E_2}(0,C)\theta^{E_1\times E_2}(0,C)$	$\iff$	$\widetilde{\mathbf{M_1M_1}}$	$\widetilde{\theta}^{J_1}(\Phi(0,0))\widetilde{\theta}^{J_1}(\Phi(0,C))$
$\mathbf{M_2M_2}$	$\theta^{E_1\times E_2}(C,0)\theta^{E_1\times E_2}(C,0)$	$\iff$	$\widetilde{\mathbf{M_2M_2}}$	$\widetilde{\theta}^{J_1}(\Phi(0,0))\widetilde{\theta}^{J_1}(\Phi(C,0))$
$\mathbf{M_3M_3}$	$\theta^{E_1\times E_2}(C,C)\theta^{E_1\times E_2}(C,C)$	$\iff$	$\widetilde{\mathbf{M_3M_3}}$	$\widetilde{\theta}^{J_1}(\Phi(0,0))\widetilde{\theta}^{J_1}(\Phi(C,C))$
$\mathbf{M_0M_1}$	$\theta^{E_1\times E_2}(0,0)\theta^{E_1\times E_2}(0,C)$	$\iff$	$\widetilde{\mathbf{M_0M_1}}$	$\widetilde{\theta}^{J_1}(\Phi(0,C'))\widetilde{\theta}^{J_1}(\Phi(0,C'))$
$\mathbf{M_2M_3}$	$\theta^{E_1\times E_2}(C,0)\theta^{E_1\times E_2}(C,C)$	$\iff$	$\widetilde{\mathbf{M_2M_3}}$	$\widetilde{\theta}^{J_1}(\Phi(0,C'))\widetilde{\theta}^{J_1}(\Phi(C,C'))$
$\mathbf{M_0M_2}$	$\theta^{E_1\times E_2}(0,0)\theta^{E_1\times E_2}(C,0)$	$\iff$	$\widetilde{\mathbf{M_0M_2}}$	$\widetilde{\theta}^{J_1}(\Phi(C',0))\widetilde{\theta}^{J_1}(\Phi(C',0))$
$\mathbf{M_1M_3}$	$\theta^{E_1\times E_2}(0,C)\theta^{E_1\times E_2}(C,C)$	$\iff$	$\widetilde{\mathbf{M_1M_3}}$	$\widetilde{\theta}^{J_1}(\Phi(C',0))\widetilde{\theta}^{J_1}(\Phi(C',C))$
$\mathbf{M_0M_3}$	$\theta^{E_1\times E_2}(0,0)\theta^{E_1\times E_2}(C,C)$	$\iff$	$\widetilde{\mathbf{M_0M_3}}$	$\widetilde{\theta}^{J_1}(\Phi(C',C'))\widetilde{\theta}^{J_1}(\Phi(C',C'))$
$\mathbf{M_1M_2}$	$\theta^{E_1\times E_2}(0,C)\theta^{E_1\times E_2}(C,0)$	$\iff$	$\widetilde{\mathbf{M_1M_2}}$	$\widetilde{\theta}^{J_1}(\Phi(C',C'))\widetilde{\theta}^{J_1}(\Phi(C',-C'))$

Table 2. Correspondence between change of basis matrix columns and theta points with $C = (0:1)$ and $C' = (1:\pm 1)$.

Position	**Type**	$\ker(\Phi)$	$(C,0)$	$(0,C)$	(C,C)
00	II	$\langle (C,C),(\alpha,\beta)\rangle$	S_2	S_2+T_1	T_1
01	I	$\langle (C,\beta),(\alpha,C)\rangle$	S_2	S_1	S_1+S_2
02	I	$\langle (C,\beta),(\alpha,\beta^{-1})\rangle$	S_2	$S_1+S_2+T_1$	S_1+T_1
10	I	$\langle (\alpha,C),(C,\beta)\rangle$	S_1+T_2	S_2+T_1	$S_1+S_2+T_1+T_2$
11	II	$\langle (\alpha,\beta),(C,C)\rangle$	S_1+T_2	S_1	T_2
12	I	$\langle (\alpha,\beta),(C,\beta^{-1})\rangle$	S_1+T_2	$S_1+S_2+T_1$	$S_2+T_1+T_2$
20	I	$\langle (\alpha,C),(\alpha^{-1},\beta)\rangle$	$S_1+S_2+T_2$	S_2+T_1	$S_1+T_1+T_2$
21	I	$\langle (\alpha,\beta),(\alpha^{-1},C)\rangle$	$S_1+S_2+T_2$	S_1	S_2+T_2
22	II	$\langle (\alpha,\beta),(\alpha^{-1},\beta^{-1})\rangle$	$S_1+S_2+T_2$	$S_1+S_2+T_1$	T_1+T_2

Table 3. Different positions of $(0:1)$ points in the symplectic basis depending on the basis of $\ker(\Phi)$ where α and β denote non-trivial 2 torsion points.

The reason the Hadamard transform in corollary 1 is particularly interesting compared to its counterpart lies in the sparsity of most of its theta points, as shown in lemma 3.

Lemma 3. *Let $\Phi : A \to B$ be a $(2,2)$-isogeny between (principally polarised) Abelian surface with kernel K_2. Let θ^A be a theta structure induced by the symplectic basis $K_1' \oplus K_2'$ and let θ^B be a theta structure induced by the symplectic basis $\Phi(K_1') \oplus \Phi(K_2'')$. Let $P \in A[4]$ such that $\pi_A(P) = (x', \widehat{x}')$, with $\pi_A([2]P) = (x, \widehat{x})$*[8]*.*

1. *$\pi_B(\Phi(P)) = (x', \widehat{x})$ and thus $\widetilde{\theta}^B_j(\Phi(P)) = \widetilde{\theta}^B_{j+\widehat{x}}(V_{x'})$ for $V_{x'} \in B[4]$ s.t. $\pi_B(V_{x'}) = (x', 0)$.*
2. *Additionally, if $x' \notin \mathbb{Z}_2^2$, then $\widetilde{\theta}^B_j(\Phi(P)) = 0$ for j such that $j \cdot x \neq \widehat{x} \cdot x$.*

Proof.

[8] Note that $x = x' \mod 2$ and $\widehat{x} = \widehat{x}' \mod 2$

1. Using the symplectic representation of $A[4]$, we write $P = \sum_{i=1,2}[x'_i]S'_i + [\widehat{x}'_i]T'_i$. We then have that $\Phi(P) = \sum_{i=1,2}[x'_i]\Phi(S'_i) + [\widehat{x}_i]\Phi(T'_i))$ as $\Phi(T'_i)$ are of order 2, giving us that $\pi_B(\Phi(B)) = (x', \widehat{x})$, with the following being direct of the definition of our theta structures.
2. This is a straightforward corollary of proposition 1 using the fact that duality permutes the symplectic basis.

□

Proposition 1. *Let A be a (principally polarised) Abelian surface with θ^A theta structure of symplectic basis $\langle S_1, S_2\rangle \oplus \langle T_1, T_2\rangle = A[2]$. Let P be a 4-torsion point in A, with $[2]P \in \langle T_1, T_2\rangle$, i.e. $\pi_A([2]P) = (0, \widehat{x})$.*

$$\theta^A_j(P) = 0 \text{ for } j \text{ s.t. } j \cdot \widehat{x} = 1, \text{ i.e. } \begin{cases} j = 01, 11 & \text{if } [2]P = T_1 \\ j = 10, 11 & \text{if } [2]P = T_2 \\ j = 01, 10 & \text{if } [2]P = T_1 + T_2 \end{cases}$$

Proof. Let $\widehat{x} \in \mathbb{Z}_2^2$ be such that $\pi([2]P) = (0, \widehat{x})$. We have that

$$\theta^A_j(P) = \theta^A_j(-P) = \theta^A_j(P + [2]P) = (-1)^{\widehat{x \cdot j}}\theta^A_j(P)$$

$$\text{Therefore, we have that } \theta^A_j(P) = \begin{cases} \theta^A_j(P) & \text{if } \widehat{x} \cdot j = 0 \\ -\theta^A_j(P) & \text{if } \widehat{x} \cdot j = 1 \end{cases}$$

□

Remark 2. Interestingly, both lemma 3 and proposition 1—the key properties underlying our fast formulae—are intrinsic to theta structures and do not rely on the gluing construction. As a result, the formulae presented in this paper can, *mutatis mutandis*, be generalized to arbitrary $(2, 2)$-isogenies.

We now explain how we derive fast formulae depending if the position of our kernel basis is of Type I or Type II.

4.3 Type I evaluation formulae

We first consider the Type I, in which we have that the point $(C, C) \notin \ker(\Phi)$. We therefore have that $\pi_{E_1\times E_2}((0, C')) = (x'_1, \widehat{x_1}')$ and $\pi_{E_1\times E_2}((C', 0)) = (x'_2, \widehat{x_2}')$ with $\langle x'_1, x'_2\rangle = \mathbb{Z}_4^2$. Let $(x'_3, \widehat{x_3}') = (x'_1 + x'_2, \widehat{x_1}' + \widehat{x_2}')$. We detail what occurs for all our squares of theta points.

1. As the point $\widetilde{\mathbf{M}_0\mathbf{M}_0} = (\widetilde{\theta}^{J_1}(0))^{\odot 2}$ is the square of the theta null point, we know, following our choice of theta structure, that $(\widetilde{\theta}^{J_1}_{11}(0))^{\odot 2} = 0$ which implies that $\mathbf{M}^2_{0,0} + \mathbf{M}^2_{3,0} = \mathbf{M}^2_{1,0} + \mathbf{M}^2_{2,0}$. We can therefore write $\widetilde{\mathbf{M}_0\mathbf{M}_0}$ as:

$$\widetilde{\mathbf{M}_0\mathbf{M}_0} = \left(\theta^{J_1}(0)\right)^{\odot 2} = \begin{pmatrix} \mathbf{M}^2_{1,0} + \mathbf{M}^2_{2,0} \\ \mathbf{M}^2_{0,0} - \mathbf{M}^2_{1,0} \\ \mathbf{M}^2_{0,0} - \mathbf{M}^2_{2,0} \\ 0 \end{pmatrix}$$

Additionally, following table 2 and table 3 we have that $[\widetilde{\mathbf{M}_i\mathbf{M}_i}]_j = \widetilde{\theta}^B_j(0) \cdot \widetilde{\theta}^B_j(\Phi(C^{i\cdot 10}, C^{i\cdot 01})) = (-1)^{x_i \cdot j}[\widetilde{\mathbf{M}_0\mathbf{M}_0}]_j$ for $i = 1, 2, 3$.

2. As $\widetilde{\mathbf{M}_0\mathbf{M}_1} = (\widetilde{\theta}^{J_1}(\Phi(0, C'))^{\odot 2}$, following lemma 3, it has two non-zero components, whenever $j \cdot x_1 = j \cdot \widehat{x_1}$. Additionally, following table 2, we have that

$$[\widetilde{\mathbf{M}_2\mathbf{M}_3}]_j = \widetilde{\theta}^B_{j+\widehat{x_1}}(V_{x_1'}) \cdot \widetilde{\theta}^B_{j+\widehat{x_1}}(V_{2x_2+x_1'}) = (-1)^{x_2 \cdot (j+\widehat{x_1})}[\widetilde{\mathbf{M}_0\mathbf{M}_1}]_j$$

inducing that $\widetilde{\mathbf{M}_2\mathbf{M}_3}$ shares the same zero coefficients as $\widetilde{\mathbf{M}_0\mathbf{M}_1}$ and that, among the remaining components, one is equal while the second is sign-flipped. This enables us to recover the 2 components of $\widetilde{\mathbf{M}_0\mathbf{M}_2}$ as

$$\mathbf{M}_{0,0}\mathbf{M}_{0,1} \pm \mathbf{M}_{0,2}\mathbf{M}_{0,3}$$

3. The computation of $\widetilde{\mathbf{M}_0\mathbf{M}_2}$ and $\widetilde{\mathbf{M}_1\mathbf{M}_3}$ is analog to the prior case. We recover the two components of $\widetilde{\mathbf{M}_0\mathbf{M}_2}$ as

$$\mathbf{M}_{0,0}\mathbf{M}_{0,2} \pm \mathbf{M}_{0,1}\mathbf{M}_{0,3}$$

4. Finally, in Type I, the computation of $\widetilde{\mathbf{M}_0\mathbf{M}_3}$ and $\widetilde{\mathbf{M}_1\mathbf{M}_2}$ is analog to the prior two cases. We thus recover the components of $\widetilde{\mathbf{M}_0\mathbf{M}_3}$ as

$$\mathbf{M}_{0,0}\mathbf{M}_{0,3} \pm \mathbf{M}_{0,1}\mathbf{M}_{0,2}$$

Putting everything together, we obtain new Type I formulae for the duplication operation, which we call *Superglue*. We present here the explicit formulae for position 01. The formulae for all other positions are detailed in appendix A.

Theorem 2 (Superglue formula for position 01). *Let $\theta^{E_1\times E_2} : E_1 \times E_2 \to \mathbb{P}^3$ be a theta structure such that the symplectic basis over $(E_1 \times E_2)[2]$ induces is of the form $\langle (0, C), (C, 0)\rangle \oplus \langle (C, \alpha), (\beta, C)\rangle$ with α, β non-canonical 2 torsion points and $\mathbf{M}$ its change of coordinate matrix. For any $P, Q \in E_1 \times E_2$ with barycentric components $P_1 \pm Q_1$ and $P_2 \pm Q_2$, we have that*

$$\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q) \odot \theta^{E_1\times E_2}(P-Q)\Big) =$$

$$\begin{pmatrix} \mathbf{M_{1,0}}^2 + \mathbf{M_{2,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{1,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{2,0}}^2 \\ 0 \end{pmatrix} \odot \begin{pmatrix} (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 - w_2^2) \\ (u_1^2 - v_1^2 - w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ 0 \end{pmatrix} + 2u_2w_2 \begin{pmatrix} \mathbf{M_{0,0}M_{0,1}} + \mathbf{M_{0,2}M_{0,3}} \\ 0 \\ \mathbf{M_{0,0}M_{0,1}} - \mathbf{M_{0,2}M_{0,3}} \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_1^2 - v_1^2 + w_1^2 \\ 0 \\ u_1^2 - v_1^2 - w_1^2 \\ 0 \end{pmatrix}$$

$$+ 2u_1w_1 \begin{pmatrix} \mathbf{M_{0,0}M_{0,2}} + \mathbf{M_{0,1}M_{0,3}} \\ \mathbf{M_{0,0}M_{0,2}} - \mathbf{M_{0,1}M_{0,3}} \\ 0 \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_2^2 - v_2^2 + w_2^2 \\ u_2^2 - v_2^2 - w_2^2 \\ 0 \\ 0 \end{pmatrix} + 4w_1w_2 \begin{pmatrix} \mathbf{M_{0,0}M_{0,3}} + \mathbf{M_{0,1}M_{0,3}} \\ 0 \\ 0 \\ \mathbf{M_{0,0}M_{0,3}} - \mathbf{M_{0,1}M_{0,3}} \end{pmatrix} \odot \begin{pmatrix} u_1u_2 \\ 0 \\ 0 \\ v_1v_2 \end{pmatrix}$$

Remark 3 (Technical Remark). To derive the superglue formulae, we are implicitly using the action of $(E_1 \times E_2)[2]$ not over $\mathbb{P}^3$ but over an affine lift $\mathbb{F}_q^4$ in which it is defined times a factor ω that makes the point consistent with the symmetric elements of the symplectic basis underlying our theta structure. In the dimension 2 case, we have that $\omega = 1$ for S_1, S_2, T_1, T_2 and $\omega = -1$ for $S_1 + S_2$ and $T_1 + T_2$.

4.4 Type II evaluation formulae

Type II occurs whenever $(C, C) \in \ker(\Phi)$. This induces that the point $\Phi(C', C')$ is in fact a 2-torsion point and is therefore a permutation of the dual theta null point. This additionally induces that $x_1 = x_2$ while $\widehat{x_1} \neq \widehat{x_2}$.

1. The computation of $\widetilde{\mathbf{M_0}\mathbf{M_0}}$ is the same as in Type I. This also holds for $\widetilde{\mathbf{M}_i\mathbf{M}_i}$ for $i = 1, 2, 3$ that are such that $\widetilde{\mathbf{M_1}\mathbf{M_1}} = \widetilde{\mathbf{M_2}\mathbf{M_2}}$ and $\widetilde{\mathbf{M_0}\mathbf{M_0}} = \widetilde{\mathbf{M_3}\mathbf{M_3}}$.
2. Similarly to the Type I, we have that $\widetilde{\mathbf{M_0}\mathbf{M_1}}$ and $\widetilde{\mathbf{M_2}\mathbf{M_3}}$ have two non-zero components *but* they are in fact equal. (This comes from the fact that the action of $\Phi(C, 0)$ sign-flips the zero components.). To recover the coordinates, we therefore need to use another line of the change of basis matrix
$$\mathbf{M}_{0,0}\mathbf{M}_{0,1} \pm \mathbf{M}_{i,0}\mathbf{M}_{i,1}$$
with $i = 1, 2$ depending on the position.
3. As $x_1' = x_2' \mod 2$ while $\widehat{x_1}' \neq \widehat{x_2}'$, we have that, *up to sign*, the non-zero components of $\widetilde{\mathbf{M_0}\mathbf{M_2}}$ are the permutation of those in $\widetilde{\mathbf{M_0}\mathbf{M_1}}$. This sign difference depends on whether the points C' above C in E_1 and E_2 that induce the theta structure over $E_1 \times E_2$ are the same of not. This ambiguity can be lifted by simply checking if $\mathbf{M}_{0,1} = -\mathbf{M}_{0,2}$. If it does, the sign is flipped.
4. Finally, in Type II, the computation of $\widetilde{\mathbf{M_0}\mathbf{M_3}}$ and $\widetilde{\mathbf{M_1}\mathbf{M_2}}$ are derived from the permutation of the square of the dual theta-null point $(\widetilde{\theta}^{J_1}(0))^{\odot 2}$ up to sign with the same reason as previously.

Putting everything together, we get the Type II superglue formula. We give here the case explicit formula for position 00. The others two formulae are detailed in appendix A.

Theorem 3 (Superglue formula for position 00). *Let $\theta : E_1 \times E_2 \to \mathbb{P}^3$ be a theta structure such that the symplectic basis it induces is of from $\langle (0, \beta), (C, 0) \rangle \oplus \langle (C, C), (\alpha, \beta) \rangle$ with α, β non-canonical 2 torsion points and with $\mathbf{M}$ its change of coordinate matrix. For any $P, Q \in E_1 \times E_2$ with barycentric coordinates $P_1 \pm Q_1$ and $P_2 \pm Q_2$, we have that*

$$\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q) \odot \theta^{E_1\times E_2}(P-Q)\Big) =$$

$$\begin{pmatrix} \mathbf{M_{1,0}}^2 + \mathbf{M_{2,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{1,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{2,0}}^2 \\ 0 \end{pmatrix} \odot \begin{pmatrix} (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ (u_1^2 - v_1^2 - w_1^2)(u_2^2 - v_2^2 - w_2^2) \\ 0 \end{pmatrix} + 2u_2w_2 \begin{pmatrix} \mathbf{M_{0,0}M_{0,1}} + \mathbf{M_{1,0}M_{1,1}} \\ \mathbf{M_{0,0}M_{0,1}} - \mathbf{M_{1,0}M_{1,1}} \\ 0 \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_1^2 - v_1^2 + w_1^2 \\ u_1^2 - v_1^2 + w_1^2 \\ 0 \\ 0 \end{pmatrix}$$

$$+ (-1)^{\mathbf{M}_{0,1}=-\mathbf{M}_{0,2}} \left(2u_1w_1 \begin{pmatrix} \mathbf{M_{0,0}M_{0,1}} - \mathbf{M_{1,0}M_{1,1}} \\ \mathbf{M_{0,0}M_{0,1}} + \mathbf{M_{1,0}M_{1,1}} \\ 0 \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_2^2 - v_2^2 + w_2^2 \\ u_2^2 - v_2^2 + w_2^2 \\ 0 \\ 0 \end{pmatrix} + 4w_1w_2 \begin{pmatrix} \mathbf{M_{0,0}}^2 - \mathbf{M_{1,0}}^2 \\ \mathbf{M_{1,0}}^2 + \mathbf{M_{2,0}}^2 \\ 0 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{2,0}}^2 \end{pmatrix} \odot \begin{pmatrix} u_1u_2 \\ u_1u_2 \\ 0 \\ v_1v_2 \end{pmatrix} \right)$$

5 Concrete algorithms for Superglue

5.1 Detailed algorithms

As demonstrated by theorems 2 and 3, using the superglue formulae does not require computing the full change of coefficient $\mathbf{M}$, as we only require 6 components[9], which are multiplied together to precompute the different coefficients used for the Superglue formulae. This is described in algorithm 3. The cost reported are maximal cost, as depending on the Type of our kernel, we do not have the same minimal cost. We can nevertheless compute faster branches suboptimally, such that we use the same number of operations on all branches. Those suboptimal computations are not reported in the pseudocode of algorithm 3 for readability's sake.

Once the components are computed, they can be reused to derive the duplication formulae. This process is described in algorithm 4 for position 01 and in algorithm 5 for position 00. The implementations for the other positions are comparatively straightforward. The overall cost of a generic gluing evaluation algorithm is thus upper-bounded by the evaluation cost of a Type 2 kernel, and hence by that of algorithm 5. To achieve constant-time execution, one must suboptimally compute algorithm 4 as well.

Finally, we note that whenever $Q = 0$, algorithm 4/5 directly compute $\mathcal{H}(\theta^{E_1 \times E_2}(P)^{\odot 2})$. In this case, lines 3 and 8/9 of algorithm 4/5 can be skipped, saving $4\mathbf{M}$ and $2\mathbf{a}$. Moreover, line 1 can be carried out in $4\mathbf{S} + 4\mathbf{a}$.

5.2 Efficiency comparison

Putting everything together, the use of Superglue formulas leads to a significant improvement in the efficiency of gluing across all aspects. In table 4, we compare our algorithm against the previous state of the art, as reported in [11,1].

To ensure a fair comparison, our cost estimates are aligned with the subroutines defined in [1, Chapter 8.5], namely: ThetaChangeOfBasis corresponds to Step 1 of section 2.4 and to algorithm 3; GluingCodomain covers Steps 1 and 2 of section 2.4; GluingEval together with GluingEvalSpecial correspond to Step 3, with GluingEvalSpecial handling the special case $Q = 0$. For the evaluation routines, we exclude the cost of shifting by a 4-torsion point and the computation of barycentric coordinates, as these depend on the specific representations chosen for the underlying curves. We provide in appendix C detailed pseudocode of all algorithms using Superglue formulae.

We also implemented the Superglue formulae in both the SageMath and Rust implementations of $(2^n, 2^n)$-isogenies from [11]. The code is available at the following GitHub repository: https://github.com/MaxDuparc/Superglue.

Using our Rust implementation as a baseline, we observe that the Superglue formulae are approximately twice as fast as those of [11]. When evaluating the

[9] Though it seems at first sight theorem 3 we need to compute 7 components in Type II, it occurs that $\mathbf{M}_{0,3} = (-1)^{\mathbf{M}_{0,1}=-\mathbf{M}_{0,2}}\mathbf{M}_{0,0}$.

Algorithm 3 $\mathsf{SuperglueCoefficients}(T'_1, T'_2)$

Require: (T'_1, T'_2) an isotropic basis of $(E_1 \times E_2)[4]$ with gluing $\ker(\Phi) = \langle [2](T'_1, T'_2) \rangle$
Ensure: pos the position of our gluing isogeny Φ together and its superglue coefficients

$T'_{1,1}, T'_{1,2} \leftarrow T'_1$
$T'_{2,1}, T'_{2,2} \leftarrow T'_2$
Find pos from T'_1, T'_2 ▷ Find which points are $(1 : \pm 1)$ in $\langle T'_1, T'_2 \rangle$.
$\delta_1, G^1, G^2, G^3 \leftarrow \mathsf{CoherentSymetricElement}(T'_{1,1}, T'_{2,1})$ ▷ $4\mathbf{M} + 2\mathbf{S} + 7\mathbf{a} + 5\mathbf{i}$
$\delta_2, H^1, H^2, H^3 \leftarrow \mathsf{CoherentSymetricElement}(T'_{1,2}, T'_{2,2})$ ▷ $4\mathbf{M} + 2\mathbf{S} + 7\mathbf{a} + 5\mathbf{i}$
$\mathbf{M}_{0,0} \leftarrow \delta_1\delta_2 + G^1_{0,0} \cdot H^1_{0,0} + G^2_{0,0} \cdot H^2_{0,0} - G^3_{0,0} \cdot H^3_{0,0}$
$\mathbf{M}_{0,1} \leftarrow G^1_{0,0} \cdot H^1_{0,1} + G^2_{0,0} \cdot H^2_{0,1} - G^3_{0,0} \cdot H^3_{0,1}$
$\mathbf{M}_{0,2} \leftarrow G^1_{0,1} \cdot H^1_{0,0} + G^2_{0,1} \cdot H^2_{0,0} - G^3_{0,1} \cdot H^3_{0,0}$
$\mathbf{M}_{0,3} \leftarrow G^1_{0,1} \cdot H^1_{0,1} + G^2_{0,1} \cdot H^2_{0,1} - G^3_{0,1} \cdot H^3_{0,1}$
$\mathbf{M}_{1,0} \leftarrow H^2_{0,0} \cdot \mathbf{M}_{0,0} + H^2_{1,0} \cdot \mathbf{M}_{0,1}$
$\mathbf{M}_{2,0} \leftarrow G^1_{0,0} \cdot \mathbf{M}_{0,0} + G^1_{1,0} \cdot \mathbf{M}_{0,2}$
if pos $= 00$ or 22 **then**
    $\mathbf{M}_{1,1} \leftarrow \delta_1 \left(H^2_{0,1} \cdot \mathbf{M}_{0,2} + H^2_{1,1} \cdot \mathbf{M}_{0,3} \right)$
end if
if pos $= 11$ **then**
    $\mathbf{M}_{2,1} \leftarrow \delta_2 \left(G^2_{0,1} \cdot \mathbf{M}_{0,1} + G^2_{1,1} \cdot \mathbf{M}_{0,3} \right)$
end if
$\mathbf{M}_{0,i} \leftarrow \delta_1\delta_2 \cdot \mathbf{M}_{0,i}$ for $i = 0, \cdots, 3$
$\mathbf{M}_{1,0} \leftarrow \delta_1 \cdot \mathbf{M}_{1,0}$
$\mathbf{M}_{2,0} \leftarrow \delta_2 \cdot \mathbf{M}_{2,0}$.
$\boldsymbol{g_{00}} \leftarrow (\mathbf{M}^2_{1,0} + \mathbf{M}^2_{2,0}, \mathbf{M}^2_{0,0} - \mathbf{M}^2_{1,0}, \mathbf{M}^2_{0,0} - \mathbf{M}^2_{2,0})$ ▷ $3\mathbf{S} + 3\mathbf{a}$
if pos $= 00, 22$ **then**
    $\boldsymbol{g_{01}} \leftarrow (\mathbf{M}_{0,0}\mathbf{M}_{0,1} + \mathbf{M}_{1,0}\mathbf{M}_{1,1}, \mathbf{M}_{0,0}\mathbf{M}_{0,1} - \mathbf{M}_{1,0}\mathbf{M}_{1,1})$ ▷ $2\mathbf{M} + 2\mathbf{a}$
    $\mathsf{sgn\text{-}flip} \leftarrow \mathbf{M_{0,1}} - \mathbf{M_{0,2}} \overset{?}{\neq} \mathbf{0}$
else if pos $= 11$ **then**
    $\boldsymbol{g_{01}} \leftarrow (\mathbf{M}_{0,0}\mathbf{M}_{0,1} + \mathbf{M}_{2,0}\mathbf{M}_{2,1}, \mathbf{M}_{0,0}\mathbf{M}_{0,1} - \mathbf{M}_{2,1}\mathbf{M}_{2,1})$ ▷ $2\mathbf{M} + 2\mathbf{a}$
    $\mathsf{sgn\text{-}flip} \leftarrow \mathbf{M_{0,1}} - \mathbf{M_{0,2}} \overset{?}{\neq} \mathbf{0}$
else
    $\mathsf{sgn\text{-}flip} \leftarrow \mathsf{false}$
    $\boldsymbol{g_{01}} \leftarrow (\mathbf{M}_{0,0}\mathbf{M}_{0,1} + \mathbf{M}_{0,2}\mathbf{M}_{0,3}, \mathbf{M}_{0,0}\mathbf{M}_{0,1} - \mathbf{M}_{0,2}\mathbf{M}_{0,3})$ ▷ $2\mathbf{M} + 2\mathbf{a}$
    $\boldsymbol{g_{02}} \leftarrow (\mathbf{M}_{0,0}\mathbf{M}_{0,2} + \mathbf{M}_{0,1}\mathbf{M}_{0,3}, \mathbf{M}_{0,0}\mathbf{M}_{0,2} - \mathbf{M}_{0,1}\mathbf{M}_{0,3})$ ▷ $2\mathbf{M} + 2\mathbf{a}$
    $\boldsymbol{g_{03}} \leftarrow (\mathbf{M}_{0,0}\mathbf{M}_{0,3} + \mathbf{M}_{0,1}\mathbf{M}_{0,2}, \mathbf{M}_{0,0}\mathbf{M}_{0,3} - \mathbf{M}_{0,1}\mathbf{M}_{0,3})$ ▷ $2\mathbf{M} + 2\mathbf{a}$
end if
return pos, $\boldsymbol{g_{00}}, \boldsymbol{g_{01}}, \boldsymbol{g_{02}}, \boldsymbol{g_{02}}$, sgn-flip ▷ Total cost: $38\mathbf{M} + 7\mathbf{S} + 34\mathbf{a} + 10\mathbf{i}$

computational cost of a $(2^n, 2^n)$-isogeny chain, note that gluing occurs only at the beginning of the chain; hence, the relative speed-up decreases proportionally with the chain length. Nevertheless, we still observe clear improvements, as detailed in table 5.

Algorithm 4 EvaluateSuperglue$(P \pm Q, \boldsymbol{g_{00}}, \boldsymbol{g_{01}}, \boldsymbol{g_{02}}, \boldsymbol{g_{03}}, \mathsf{sgn\text{-}flip})$ (for $\mathsf{pos} = 01$)

Require: The addition components $P_1 \pm Q_1$ and $P_2 \pm Q_2$ and the precomputed superglue components $\boldsymbol{g_{00}}, \cdots, \boldsymbol{g_{03}}$.
Ensure: $\mathcal{H}\big(\theta^{E_1 \times E_2}(P+Q) \odot \theta^{E_1 \times E_2}(P-Q)\big)$.

Compute $u_1^2 - v_1^2 \pm w_1^2, u_2^2 - v_2^2 \pm w_2^2$ ▷ $2\mathbf{M} + 2\mathbf{S} + 8\mathbf{a}$
Compute $2u_1w_1$, $2u_2w_2$ ▷ $2\mathbf{M} + 2\mathbf{a}$
Compute $4v_1v_2w_1w_2$ ▷ $3\mathbf{M} + 2\mathbf{a}$
$\alpha \leftarrow (u_1^2 - v_1^2 + w_1^2) \cdot \big(\boldsymbol{g_{00}}[0] \cdot (u_2^2 - v_2^2 + w_2^2) + 2u_2w_2 \cdot \boldsymbol{g_{01}}[0]\big)$
$\alpha \leftarrow \quad \alpha \quad + 2u_1w_1 \cdot \big(\boldsymbol{g_{02}}[0] \cdot (u_2^2 - v_2^2 + w_2^2) + 2u_2w_2 \cdot \boldsymbol{g_{03}}[0]\big)$
$\beta \leftarrow (u_2^2 - v_2^2 - w_2^2) \cdot \big(\boldsymbol{g_{00}}[1] \cdot (u_1^2 - v_1^2 + w_1^2) + 2u_1w_1 \cdot \boldsymbol{g_{02}}[1]\big)$
$\gamma \leftarrow (u_1^2 - v_1^2 - w_1^2) \cdot \big(\boldsymbol{g_{00}}[2] \cdot (u_2^2 - v_2^2 + w_2^2) + 2u_2w_2 \cdot \boldsymbol{g_{01}}[1]\big)$
$\delta \leftarrow 4v_1v_2w_1w_2 \cdot \boldsymbol{g_{03}}[1]$.
return $(\alpha : \beta : \gamma : \delta)$ ▷ Total cost: $20\mathbf{M} + 2\mathbf{S} + 17\mathbf{a}$

Algorithm 5 EvaluateSuperglue$(P \pm Q, \boldsymbol{g_{00}}, \boldsymbol{g_{01}}, \boldsymbol{g_{02}}, \boldsymbol{g_{03}}, \mathsf{sgn\text{-}flip})$ (for $\mathsf{pos} = 00$)

Require: The addition components $P_1 \pm Q_1$ and $P_2 \pm Q_2$ and the precomputed superglue components $\boldsymbol{g_{00}}, \cdots, \boldsymbol{g_{03}}$.
Ensure: $\mathcal{H}\big(\theta^{E_1 \times E_2}(P+Q)\theta^{E_1 \times E_2}(P-Q)\big)$.

Compute $u_1^2 - v_1^2 \pm w_1^2, u_2^2 - v_2^2 \pm w_2^2$ ▷ $2\mathbf{M} + 2\mathbf{S} + 8\mathbf{a}$
Compute $2u_1w_1$, $2u_2w_2$ ▷ $2\mathbf{M} + 2\mathbf{a}$
Compute $4v_1v_2w_1w_2$ ▷ $3\mathbf{M} + 2\mathbf{a}$
$\alpha \leftarrow (u_1^2 - v_1^2 + w_1^2) \cdot \big(\boldsymbol{g_{00}}[0] \cdot (u_2^2 - v_2^2 + w_2^2) + 2u_2w_2 \cdot \boldsymbol{g_{01}}[0]\big)$
$\alpha \leftarrow \alpha \ \pm^{(1)} \ 2u_1w_1 \cdot \big(\boldsymbol{g_{01}}[1] \cdot (u_2^2 - v_2^2 + w_2^2) + 2u_2w_2 \cdot \boldsymbol{g_{00}}[1]\big)$
$\beta \leftarrow (u_1^2 - v_1^2 + w_1^2) \cdot \big(\boldsymbol{g_{00}}[1] \cdot (u_2^2 - v_2^2 + w_2^2) + 2u_2w_2 \cdot \boldsymbol{g_{01}}[1]\big)$
$\beta \leftarrow \beta \ \pm^{(1)} \ 2u_1w_1 \cdot \big(\boldsymbol{g_{01}}[0] \cdot (u_2^2 - v_2^2 + w_2^2) + 2u_2w_2 \cdot \boldsymbol{g_{00}}[0]\big)$
$\gamma \leftarrow (u_1^2 - v_1^2 - w_1^2) \cdot \boldsymbol{g_{00}}[2] \cdot (u_2^2 - v_2^2 - w_2^2)$
$\delta \leftarrow \pm^{(1)} 4v_1v_2w_1w_2 \cdot \boldsymbol{g_{00}}[2]$ ▷ $\pm^{(1)}$ depends on $\mathsf{sgn\text{-}flip}$.
return $(\alpha : \beta : \gamma : \delta)$ ▷ Total cost: $22\mathbf{M} + 2\mathbf{S} + 18\mathbf{a}$

Algorithms	[11]	[1, Chapter 8.5]	Our work
ThetaChangeOfBasis	$100\mathbf{M} + 8\mathbf{S} + 4\mathbf{I} + 49\mathbf{a}$	$113\mathbf{M} + 8\mathbf{S} + 1\mathbf{I} + 49\mathbf{a}$	$38\mathbf{M} + 7\mathbf{S} + 34\mathbf{a}$
GluingCodomain	$179\mathbf{M} + 24\mathbf{S} + 5\mathbf{I} + 113\mathbf{a}$	$175\mathbf{M} + 24\mathbf{S} + 1\mathbf{I} + 113\mathbf{a}$	$94\mathbf{M} + 23\mathbf{S} + 82\mathbf{a}$
GluingEval	$45\mathbf{M} + 8\mathbf{S} + 1\mathbf{I} + 48\mathbf{a}$	$40\mathbf{M} + 8\mathbf{S} + 44\mathbf{a}$	$26\mathbf{M} + 2\mathbf{S} + 26\mathbf{a}$
GluingEvalSpecial	–	$22\mathbf{M} + 4\mathbf{S} + 28\mathbf{a}$	$19\mathbf{M} + 4\mathbf{S} + 18\mathbf{a}$

Table 4. Cost comparison between gluing mechanisms using prior formulae or using Superglue

6 Superglue on the twist

Superglue formulae also naturally enable evaluation on quadratic twists. This comes from the fact that the barycentric representation naturally extend to the quadratic twist.

$\log p$	n	[11] Rust implementation	Our Rust implementation
254	126	861 μs	832 μs
381	208	3.08 ms	3.01 ms
1293	632	154.6 ms	153.43 ms

Table 5. Running times of computing a $(2^n, 2^n)$-isogeny between elliptic products over the base field $\mathbb{F}_{p^2}$. Times were recorded on an Apple M4-Pro chip.

Proposition 2. *Let $P \neq \pm Q \in (E \cup E^\top)$, there exists a barycentric representation of $P \pm Q = (u : v : w) \in \mathbb{P}^2$ such that:*

$$\begin{cases} x_\oplus = u - \delta_P \delta_Q v \\ x_\ominus = u + \delta_P \delta_Q v \\ z_\oplus = z_\ominus = w \end{cases}$$

using the indicator function $\delta_X = (\sqrt{B})^{\mathbf{1}_{X \in E^\top}}$, where B is the twisting parameter from definition 1.

For points not in the same curve, we can compute these coordinates using algorithm 2 by just changing line 17 and 18 ($t_2 \leftarrow \delta_P^2 \cdot t_2^2$ and $t_4 \leftarrow \delta_Q^2 \cdot t_4^2$). We can apply this representation and efficiently use the superglue formulae to compute gluing isogenies over the quadratic twist.

Example 2. In position 01, we have that:

$$\mathcal{H}\Big(\theta^{E_1 \times E_2}(P+Q) \odot \theta^{E_1 \times E_2}(P-Q)\Big) =$$

$$\begin{pmatrix} \mathbf{M_{1,0}}^2 + \mathbf{M_{2,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{1,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{2,0}}^2 \\ 0 \end{pmatrix} \odot \begin{pmatrix} (u_1^2 - \delta_1^2 v_1^2 + w_1^2)(u_2^2 - \delta_2^2 v_2^2 + w_2^2) \\ (u_1^2 - \delta_1^2 v_1^2 + w_1^2)(u_2^2 - \delta_2^2 v_2^2 - w_2^2) \\ (u_1^2 - \delta_1^2 v_1^2 - w_1^2)(u_2^2 - \delta_2^2 v_2^2 + w_2^2) \\ 0 \end{pmatrix} + 2u_2 w_2 \begin{pmatrix} \mathbf{M_{0,0}M_{0,1}} + \mathbf{M_{0,2}M_{0,3}} \\ 0 \\ \mathbf{M_{0,0}M_{0,1}} - \mathbf{M_{0,2}M_{0,3}} \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_1^2 - \delta_1^2 v_1^2 + w_1^2 \\ 0 \\ u_1^2 - \delta_1^2 v_1^2 - w_1^2 \\ 0 \end{pmatrix}$$

$$+ 2u_1 w_1 \begin{pmatrix} \mathbf{M_{0,0}M_{0,2}} + \mathbf{M_{0,1}M_{0,3}} \\ \mathbf{M_{0,0}M_{0,2}} - \mathbf{M_{0,1}M_{0,3}} \\ 0 \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_2^2 - \delta_2^2 v_2^2 + w_2^2 \\ u_2^2 - \delta_2^2 v_2^2 - w_2^2 \\ 0 \\ 0 \end{pmatrix} + 4w_1 w_2 \begin{pmatrix} \mathbf{M_{0,0}M_{0,3}} + \mathbf{M_{0,1}M_{0,3}} \\ 0 \\ 0 \\ \mathbf{M_{0,0}M_{0,3}} - \mathbf{M_{0,1}M_{0,3}} \end{pmatrix} \odot \begin{pmatrix} u_1 u_2 \\ 0 \\ 0 \\ \delta_1 \delta_2 v_1 v_2 \end{pmatrix}$$

with $\delta_1 = \delta_{P_1} \delta_{Q_1}$ and $\delta_2 = \delta_{P_2} \delta_{Q_2}$.

Provided $\delta_{P_1} \delta_{Q_1} = \delta_{P_2} \delta_{Q_2}$, we have that $\widetilde{\theta}^{J_1}(\Phi(P)) \odot \widetilde{\theta}^{J_1}(\Phi(Q))$ is well-defined over the field $\mathbb{F}_q$ and not over $\mathbb{F}_{q^2}$. If $T_1'' \in E_1 \times E_2$, then the above equality holds for all points $P \in (E_1 \times E_2) \cup (E_1^\top \times E_2^\top)$, which are all points such that $\widetilde{\theta}^{J_1}(\Phi(P))$ is defined over $\mathbb{F}_q$, meaning that superglue formulae are maximal in that aspect. This enables us to evaluate $(2^n, 2^n)$ on the quadratic twist, which is useful for isogeny based schemes such as [12,17,3].

7 Conclusion & Future work

In this paper, we introduced new techniques and formulae to efficiently compute $(2, 2)$-gluing isogenies. These formulae are both more general and more

efficient than previous constructions, opening up several promising directions for future research. One natural extension is to generalize our approach to higher-dimensional gluing. While this should be conceptually straightforward, as demonstrated by the dimension-one analog we derived in appendix B, it is likely to be computationally intensive due to the exponential growth in the number of possible configurations. Another question worth investigating is susceptible to side-channel attacks as, although we can make our implementation constant time, its behavior still depends on the structure of the computed isogeny kernel. Additionally, it would be worth investigating if the techniques we developed could be applied to generic $(2, 2)$-isogenies, or leveraged to accelerate differential additions in the context of theta structures.

References

1. Aardal, M.A., Adj, G., Aranha, D.F., Basso, A., Canales Martínez, I.A., Chávez-Saab, J., Corte-Real Santos, M., Dartois, P., De Feo, L., Duparc, M., Eriksen, J.K., Fouotsa, T.B., Gazzoni Filho, D.L., Hess, B., Kohel, D., Leroux, A., Longa, P., Maino, L., Meyer, M., Nakagawa, K., Onuki, H., Panny, L., Patranabis, S., Petit, C., Pope, G., Reijnders, K., Robert, D., Rodríguez-Henríquez, F., Schaeffler, S., Wesolowski, B.: SQIsign. Tech. rep., National Institute of Standards and Technology (2025), https://sqisign.org
2. Basso, A., Dartois, P., De Feo, L., Leroux, A., Maino, L., Pope, G., Robert, D., Wesolowski, B.: SQIsign2D-West - the fast, the small, and the safer. In: Chung, K.M., Sasaki, Y. (eds.) ASIACRYPT 2024, Part III. LNCS, vol. 15486, pp. 339–370. Springer, Singapore (Dec 2024). https://doi.org/10.1007/978-981-96-0891-1_11
3. Basso, A., Maino, L.: POKÉ: A compact and efficient PKE from higher-dimensional isogenies. In: Fehr, S., Fouque, P.A. (eds.) EUROCRYPT 2025, Part II. LNCS, vol. 15602, pp. 94–123. Springer, Cham (May 2025). https://doi.org/10.1007/978-3-031-91124-8_4
4. Cantor, D.G.: Computing in the jacobian of a hyperelliptic curve. Mathematics of computation **48**(177), 95–101 (1987)
5. Castryck, W., Decru, T.: An efficient key recovery attack on SIDH. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 423–447. Springer, Cham (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_15
6. Chen, M., Leroux, A., Panny, L.: SCALLOP-HD: Group action from 2-dimensional isogenies. In: Tang, Q., Teague, V. (eds.) PKC 2024, Part II. LNCS, vol. 14603, pp. 190–216. Springer, Cham (Apr 2024). https://doi.org/10.1007/978-3-031-57725-3_7
7. Costello, C., Smith, B.: Montgomery curves and their arithmetic - the case of large characteristic fields. Journal of Cryptographic Engineering **8**(3), 227–240 (Sep 2018). https://doi.org/10.1007/s13389-017-0157-6
8. Dartois, P.: Fast computation of 2-isogenies in dimension 4 and cryptographic applications. Cryptology ePrint Archive, Report 2024/1180 (2024), https://eprint.iacr.org/2024/1180
9. Dartois, P., Eriksen, J.K., Fouotsa, T.B., Merdy, A.H.L., Invernizzi, R., Robert, D., Rueger, R., Vercauteren, F., Wesolowski, B.: PEGASIS: Practical effective class group action using 4-dimensional isogenies. Cryptology ePrint Archive, Report 2025/401 (2025), https://eprint.iacr.org/2025/401

10. Dartois, P., Leroux, A., Robert, D., Wesolowski, B.: SQIsignHD: New dimensions in cryptography. In: Joye, M., Leander, G. (eds.) EUROCRYPT 2024, Part I. LNCS, vol. 14651, pp. 3–32. Springer, Cham (May 2024). https://doi.org/10.1007/978-3-031-58716-0_1
11. Dartois, P., Maino, L., Pope, G., Robert, D.: An algorithmic approach to (2, 2)-isogenies in the theta model and applications to isogeny-based cryptography. In: Chung, K.M., Sasaki, Y. (eds.) ASIACRYPT 2024, Part III. LNCS, vol. 15486, pp. 304–338. Springer, Singapore (Dec 2024). https://doi.org/10.1007/978-981-96-0891-1_10
12. Duparc, M., Fouotsa, T.B.: SQIPrime: A dimension 2 variant of SQISignHD with non-smooth challenge isogenies. In: Chung, K.M., Sasaki, Y. (eds.) ASIACRYPT 2024, Part III. LNCS, vol. 15486, pp. 396–429. Springer, Singapore (Dec 2024). https://doi.org/10.1007/978-981-96-0891-1_13
13. Dupont, R.: Moyenne arithmético-géométrique, suites de Borchardt et applications. Ph.D. thesis, PhD thesis, École polytechnique (2006)
14. Farashahi, R.R., Fadavi, M., Sabbaghian, S.: Faster complete addition laws for montgomery curves. IACR TCHES **2024**(4), 737–762 (2024). https://doi.org/10.46586/tches.v2024.i4.737-762
15. Gaudry, P.: Fast genus 2 arithmetic based on theta functions. Journal of Mathematical Cryptology **1**(3), 243–265 (2007). https://doi.org/doi:10.1515/JMC.2007.012, https://doi.org/10.1515/JMC.2007.012
16. Kani, E.: The number of curves of genus two with elliptic differentials. Journal für die reine und angewandte Mathematik **1997**(485), 93–122 (1997). https://doi.org/10.1515/1997.485.932
17. Leroux, A.: Verifiable random function from the deuring correspondence and higher dimensional isogenies. In: Fehr, S., Fouque, P.A. (eds.) EUROCRYPT 2025, Part VII. LNCS, vol. 15607, pp. 167–194. Springer, Cham (May 2025). https://doi.org/10.1007/978-3-031-91098-2_7
18. Lin, J., Wang, S., Zhao, C.A.: A note on (2,2)-isogenies via theta coordinates. Finite Fields and Their Applications **99**, 102496 (2024). https://doi.org/https://doi.org/10.1016/j.ffa.2024.102496
19. Lubicz, D., Robert, D.: Computing isogenies between abelian varieties. Compositio Mathematica **148**(5), 1483–1515 (2012). https://doi.org/10.1112/S0010437X12000243
20. Lubicz, D., Robert, D.: Computing separable isogenies in quasi-optimal time. LMS Journal of Computation and Mathematics **18**(1), 198–216 (2015). https://doi.org/10.1112/S146115701400045X
21. Lubicz, D., Robert, D.: Arithmetic on abelian and kummer varieties. Finite Fields and Their Applications **39**, 130–158 (2016). https://doi.org/https://doi.org/10.1016/j.ffa.2016.01.009
22. Lubicz, D., Robert, D.: Fast change of level and applications to isogenies. Research in Number Theory **18**(1), 7 (2022). https://doi.org/10.1007/s40993-022-00407-9
23. Maino, L., Martindale, C., Panny, L., Pope, G., Wesolowski, B.: A direct key recovery attack on SIDH. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 448–471. Springer, Cham (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_16
24. Moriya, T.: LIT-SiGamal: An efficient isogeny-based PKE based on a LIT diagram. Cryptology ePrint Archive, Report 2024/521 (2024), https://eprint.iacr.org/2024/521
25. Mumford, D.B.: On the equations defining abelian varieties. I. Inventiones mathematicae (1966). https://doi.org/10.1007/BF01389737

26. Mumford, D.B.: On the equations defining abelian varieties. II. Inventiones mathematicae (1966). https://doi.org/10.1007/BF01389741
27. Mumford, D.B.: On the equations defining abelian varieties. III. Inventiones mathematicae (1967). https://doi.org/10.1007/BF01425401
28. Nakagawa, K., Onuki, H.: QFESTA: Efficient algorithms and parameters for FESTA using quaternion algebras. In: Reyzin, L., Stebila, D. (eds.) CRYPTO 2024, Part V. LNCS, vol. 14924, pp. 75–106. Springer, Cham (Aug 2024). https://doi.org/10.1007/978-3-031-68388-6_4
29. Nakagawa, K., Onuki, H., Castryck, W., Chen, M., Invernizzi, R., Lorenzon, G., Vercauteren, F.: SQIsign2D-East: A new signature scheme using 2-dimensional isogenies. In: Chung, K.M., Sasaki, Y. (eds.) ASIACRYPT 2024, Part III. LNCS, vol. 15486, pp. 272–303. Springer, Singapore (Dec 2024). https://doi.org/10.1007/978-981-96-0891-1_9
30. Robert, D.: Fonctions thêta et applications à la cryptographie. Ph.D. thesis, Université Henri Poincaré-Nancy I (2010)
31. Robert, D.: Breaking SIDH in polynomial time. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part V. LNCS, vol. 14008, pp. 472–503. Springer, Cham (Apr 2023). https://doi.org/10.1007/978-3-031-30589-4_17
32. Rück, H.G.: Abelian surfaces and jacobian varieties over finite fields. Compositio Mathematica **76**(3), 351–366 (1990)
33. Silverman, J.H.: The arithmetic of elliptic curves, vol. 106. Springer (2009). https://doi.org/10.1007/978-0-387-09494-6
34. Smith, B.A., et al.: Explicit endomorphisms and correspondences. Ph.D. thesis, University of Sydney (2005)
35. Weil, A.: Sur le théorème de Torelli. In: Séminaire Bourbaki : années 1956/57 - 1957/58, exposés 137-168, pp. 207–211. No. 4 in Astérisque, Société mathématique de France (1958)
36. Yoshizumi, R., Onuki, H., Ohashi, R., Kudo, M., Nuida, K.: Efficient theta-based algorithms for computing (ℓ, ℓ)-isogenies on kummer surfaces for arbitrary odd ℓ. In: Niederhagen, R., Saarinen, M.J.O. (eds.) Post-Quantum Cryptography. pp. 3–37. Springer Nature Switzerland, Cham (2025). https://doi.org/10.1007/978-3-031-86602-9_1

A All superglue formulae

We list in this appendix the superglue formulae for all 9 possible positions in dimension 2.

– **Position 00**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q)\odot\theta^{E_1\times E_2}(P-Q)\Big) =$

$$\begin{pmatrix}\mathbf{M_{1,0}}^2+\mathbf{M_{2,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{1,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{2,0}}^2\\ 0\end{pmatrix}\odot\begin{pmatrix}(u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2+w_2^2)\\ (u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2+w_2^2)\\ (u_1^2-v_1^2-w_1^2)(u_2^2-v_2^2-w_2^2)\\ 0\end{pmatrix}+2u_2w_2\begin{pmatrix}\mathbf{M_{0,0}M_{0,1}}+\mathbf{M_{1,0}M_{1,1}}\\ \mathbf{M_{0,0}M_{0,1}}-\mathbf{M_{1,0}M_{1,1}}\\ 0\\ 0\end{pmatrix}\odot\begin{pmatrix}u_1^2-v_1^2+w_1^2\\ u_1^2-v_1^2+w_1^2\\ 0\\ 0\end{pmatrix}$$

$$+(-1)^{M_{0,1}=-M_{0,2}}\left(2u_1w_1\begin{pmatrix}\mathbf{M_{0,0}M_{0,1}}-\mathbf{M_{1,0}M_{1,1}}\\ \mathbf{M_{0,0}M_{0,1}}+\mathbf{M_{1,0}M_{1,1}}\\ 0\\ 0\end{pmatrix}\odot\begin{pmatrix}u_2^2-v_2^2+w_2^2\\ u_2^2-v_2^2+w_2^2\\ 0\\ 0\end{pmatrix}+4w_1w_2\begin{pmatrix}\mathbf{M_{0,0}}^2-\mathbf{M_{1,0}}^2\\ \mathbf{M_{1,0}}^2+\mathbf{M_{2,0}}^2\\ 0\\ \mathbf{M_{0,0}}^2-\mathbf{M_{2,0}}^2\end{pmatrix}\odot\begin{pmatrix}u_1u_2\\ u_1u_2\\ 0\\ v_1v_2\end{pmatrix}\right)$$

– **Position 01**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q)\odot\theta^{E_1\times E_2}(P-Q)\Big)=$

$$\begin{pmatrix}\mathbf{M_{1,0}}^2+\mathbf{M_{2,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{1,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{2,0}}^2\\ 0\end{pmatrix}\odot\begin{pmatrix}(u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2+w_2^2)\\ (u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2-w_2^2)\\ (u_1^2-v_1^2-w_1^2)(u_2^2-v_2^2+w_2^2)\\ 0\end{pmatrix}+2u_2w_2\begin{pmatrix}\mathbf{M_{0,0}M_{0,1}}+\mathbf{M_{0,2}M_{0,3}}\\ 0\\ \mathbf{M_{0,0}M_{0,1}}-\mathbf{M_{0,2}M_{0,3}}\\ 0\end{pmatrix}\odot\begin{pmatrix}u_1^2-v_1^2+w_1^2\\ 0\\ u_1^2-v_1^2-w_1^2\\ 0\end{pmatrix}$$
$$+2u_1w_1\begin{pmatrix}\mathbf{M_{0,0}M_{0,2}}+\mathbf{M_{0,1}M_{0,3}}\\ \mathbf{M_{0,0}M_{0,2}}-\mathbf{M_{0,1}M_{0,3}}\\ 0\\ 0\end{pmatrix}\odot\begin{pmatrix}u_2^2-v_2^2+w_2^2\\ u_2^2-v_2^2-w_2^2\\ 0\\ 0\end{pmatrix}+4w_1w_2\begin{pmatrix}\mathbf{M_{0,0}M_{0,3}}+\mathbf{M_{0,1}M_{0,3}}\\ 0\\ 0\\ \mathbf{M_{0,0}M_{0,3}}-\mathbf{M_{0,1}M_{0,3}}\end{pmatrix}\odot\begin{pmatrix}u_1u_2\\ 0\\ 0\\ v_1v_2\end{pmatrix}$$

– **Position 02**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q)\odot\theta^{E_1\times E_2}(P-Q)\Big)=$

$$\begin{pmatrix}\mathbf{M_{1,0}}^2+\mathbf{M_{2,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{1,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{2,0}}^2\\ 0\end{pmatrix}\odot\begin{pmatrix}(u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2-w_2^2)\\ (u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2+w_2^2)\\ (u_1^2-v_1^2-w_1^2)(u_2^2-v_2^2+w_2^2)\\ 0\end{pmatrix}+2u_2w_2\begin{pmatrix}0\\ \mathbf{M_{0,0}M_{0,1}}+\mathbf{M_{0,2}M_{0,3}}\\ \mathbf{M_{0,0}M_{0,1}}-\mathbf{M_{0,2}M_{0,3}}\\ 0\end{pmatrix}\odot\begin{pmatrix}0\\ u_1^2-v_1^2+w_1^2\\ u_1^2-v_1^2-w_1^2\\ 0\end{pmatrix}$$
$$+2u_1w_1\begin{pmatrix}\mathbf{M_{0,0}M_{0,2}}-\mathbf{M_{0,1}M_{0,3}}\\ \mathbf{M_{0,0}M_{0,2}}+\mathbf{M_{0,1}M_{0,3}}\\ 0\\ 0\end{pmatrix}\odot\begin{pmatrix}u_2^2-v_2^2-w_2^2\\ u_2^2-v_2^2+w_2^2\\ 0\\ 0\end{pmatrix}+4w_1w_2\begin{pmatrix}0\\ \mathbf{M_{0,0}M_{0,3}}+\mathbf{M_{0,1}M_{0,3}}\\ 0\\ \mathbf{M_{0,0}M_{0,3}}-\mathbf{M_{0,1}M_{0,3}}\end{pmatrix}\odot\begin{pmatrix}0\\ u_1u_2\\ 0\\ v_1v_2\end{pmatrix}$$

– **Position 10**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q)\odot\theta^{E_1\times E_2}(P-Q)\Big)=$

$$\begin{pmatrix}\mathbf{M_{1,0}}^2+\mathbf{M_{2,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{1,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{2,0}}^2\\ 0\end{pmatrix}\odot\begin{pmatrix}(u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2+w_2^2)\\ (u_1^2-v_1^2-w_1^2)(u_2^2-v_2^2+w_2^2)\\ (u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2-w_2^2)\\ 0\end{pmatrix}+2u_2w_2\begin{pmatrix}\mathbf{M_{0,0}M_{0,1}}+\mathbf{M_{0,2}M_{0,3}}\\ \mathbf{M_{0,0}M_{0,1}}-\mathbf{M_{0,2}M_{0,3}}\\ 0\\ 0\end{pmatrix}\odot\begin{pmatrix}u_1^2-v_1^2+w_1^2\\ u_1^2-v_1^2-w_1^2\\ 0\\ 0\end{pmatrix}$$
$$+2u_1w_1\begin{pmatrix}\mathbf{M_{0,0}M_{0,2}}+\mathbf{M_{0,1}M_{0,3}}\\ 0\\ \mathbf{M_{0,0}M_{0,2}}-\mathbf{M_{0,1}M_{0,3}}\\ 0\end{pmatrix}\odot\begin{pmatrix}u_2^2-v_2^2+w_2^2\\ 0\\ u_2^2-v_2^2-w_2^2\\ 0\end{pmatrix}+4w_1w_2\begin{pmatrix}\mathbf{M_{0,0}M_{0,3}}+\mathbf{M_{0,1}M_{0,3}}\\ 0\\ 0\\ \mathbf{M_{0,0}M_{0,3}}-\mathbf{M_{0,1}M_{0,3}}\end{pmatrix}\odot\begin{pmatrix}u_1u_2\\ 0\\ 0\\ v_1v_2\end{pmatrix}$$

– **Position 11**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q)\odot\theta^{E_1\times E_2}(P-Q)\Big)=$

$$\begin{pmatrix}\mathbf{M_{1,0}}^2+\mathbf{M_{2,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{1,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{2,0}}^2\\ 0\end{pmatrix}\odot\begin{pmatrix}(u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2+w_2^2)\\ (u_1^2-v_1^2-w_1^2)(u_2^2-v_2^2-w_2^2)\\ (u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2+w_2^2)\\ 0\end{pmatrix}+2u_2w_2\begin{pmatrix}\mathbf{M_{0,0}M_{0,1}}+\mathbf{M_{2,0}M_{2,1}}\\ 0\\ \mathbf{M_{0,0}M_{0,1}}-\mathbf{M_{2,0}M_{2,1}}\\ 0\end{pmatrix}\odot\begin{pmatrix}u_1^2-v_1^2+w_1^2\\ 0\\ u_1^2-v_1^2+w_1^2\\ 0\end{pmatrix}$$
$$+(-1)^{M_{0,1}=-M_{0,2}}\left(2u_1w_1\begin{pmatrix}\mathbf{M_{0,0}M_{0,1}}-\mathbf{M_{2,0}M_{2,1}}\\ 0\\ \mathbf{M_{0,0}M_{0,1}}+\mathbf{M_{2,0}M_{2,1}}\\ 0\end{pmatrix}\odot\begin{pmatrix}u_2^2-v_2^2+w_2^2\\ 0\\ u_2^2-v_2^2+w_2^2\\ 0\end{pmatrix}+4w_1w_2\begin{pmatrix}\mathbf{M_{0,0}}^2-\mathbf{M_{2,0}}^2\\ 0\\ \mathbf{M_{1,0}}^2+\mathbf{M_{2,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{1,0}}^2\end{pmatrix}\odot\begin{pmatrix}u_1u_2\\ 0\\ u_1u_2\\ v_1v_2\end{pmatrix}\right)$$

– **Position 12**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q)\odot\theta^{E_1\times E_2}(P-Q)\Big)=$

$$\begin{pmatrix}\mathbf{M_{1,0}}^2+\mathbf{M_{2,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{1,0}}^2\\ \mathbf{M_{0,0}}^2-\mathbf{M_{2,0}}^2\\ 0\end{pmatrix}\odot\begin{pmatrix}(u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2-w_2^2)\\ (u_1^2-v_1^2-w_1^2)(u_2^2-v_2^2+w_2^2)\\ (u_1^2-v_1^2+w_1^2)(u_2^2-v_2^2+w_2^2)\\ 0\end{pmatrix}+2u_2w_2\begin{pmatrix}0\\ \mathbf{M_{0,0}M_{0,1}}+\mathbf{M_{0,2}M_{0,3}}\\ \mathbf{M_{0,0}M_{0,1}}-\mathbf{M_{0,2}M_{0,3}}\\ 0\end{pmatrix}\odot\begin{pmatrix}0\\ u_1^2-v_1^2+w_1^2\\ u_1^2-v_1^2-w_1^2\\ 0\end{pmatrix}$$

$$+ 2u_1w_1 \begin{pmatrix} \mathbf{M_{0,0}M_{0,2} - M_{0,1}M_{0,3}} \\ 0 \\ \mathbf{M_{0,0}M_{0,2} + M_{0,1}M_{0,3}} \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_2^2 - v_2^2 - w_2^2 \\ 0 \\ u_2^2 - v_2^2 + w_2^2 \\ 0 \end{pmatrix} + 4w_1w_2 \begin{pmatrix} 0 \\ 0 \\ \mathbf{M_{0,0}M_{0,3} + M_{0,1}M_{0,3}} \\ \mathbf{M_{0,0}M_{0,3} - M_{0,1}M_{0,3}} \end{pmatrix} \odot \begin{pmatrix} 0 \\ 0 \\ u_1u_2 \\ v_1v_2 \end{pmatrix}$$

– **Position 20**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q) \odot \theta^{E_1\times E_2}(P-Q)\Big) =$

$$\begin{pmatrix} \mathbf{M_{1,0}}^2 + \mathbf{M_{2,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{1,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{2,0}}^2 \\ 0 \end{pmatrix} \odot \begin{pmatrix} (u_1^2 - v_1^2 - w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 - w_2^2) \\ 0 \end{pmatrix} + 2u_2w_2 \begin{pmatrix} \mathbf{M_{0,0}M_{0,1} - M_{0,2}M_{0,3}} \\ \mathbf{M_{0,0}M_{0,1} + M_{0,2}M_{0,3}} \\ 0 \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_1^2 - v_1^2 - w_1^2 \\ u_1^2 - v_1^2 + w_1^2 \\ 0 \\ 0 \end{pmatrix}$$

$$+ 2u_1w_1 \begin{pmatrix} 0 \\ \mathbf{M_{0,0}M_{0,2} + M_{0,1}M_{0,3}} \\ \mathbf{M_{0,0}M_{0,2} - M_{0,1}M_{0,3}} \\ 0 \end{pmatrix} \odot \begin{pmatrix} 0 \\ u_2^2 - v_2^2 + w_2^2 \\ u_2^2 - v_2^2 - w_2^2 \\ 0 \end{pmatrix} + 4w_1w_2 \begin{pmatrix} 0 \\ \mathbf{M_{0,0}M_{0,3} + M_{0,1}M_{0,3}} \\ 0 \\ \mathbf{M_{0,0}M_{0,3} - M_{0,1}M_{0,3}} \end{pmatrix} \odot \begin{pmatrix} 0 \\ u_1u_2 \\ 0 \\ v_1v_2 \end{pmatrix}$$

– **Position 21**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q) \odot \theta^{E_1\times E_2}(P-Q)\Big) =$

$$\begin{pmatrix} \mathbf{M_{1,0}}^2 + \mathbf{M_{2,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{1,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{2,0}}^2 \\ 0 \end{pmatrix} \odot \begin{pmatrix} (u_1^2 - v_1^2 - w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 - w_2^2) \\ (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ 0 \end{pmatrix} + 2u_2w_2 \begin{pmatrix} \mathbf{M_{0,0}M_{0,1} - M_{0,2}M_{0,3}} \\ 0 \\ \mathbf{M_{0,0}M_{0,1} + M_{0,2}M_{0,3}} \\ 0 \end{pmatrix} \odot \begin{pmatrix} u_1^2 - v_1^2 - w_1^2 \\ 0 \\ u_1^2 - v_1^2 + w_1^2 \\ 0 \end{pmatrix}$$

$$+ 2u_1w_1 \begin{pmatrix} 0 \\ \mathbf{M_{0,0}M_{0,2} - M_{0,1}M_{0,3}} \\ \mathbf{M_{0,0}M_{0,2} + M_{0,1}M_{0,3}} \\ 0 \end{pmatrix} \odot \begin{pmatrix} 0 \\ u_2^2 - v_2^2 - w_2^2 \\ u_2^2 - v_2^2 + w_2^2 \\ 0 \end{pmatrix} + 4w_1w_2 \begin{pmatrix} 0 \\ 0 \\ \mathbf{M_{0,0}M_{0,3} + M_{0,1}M_{0,3}} \\ \mathbf{M_{0,0}M_{0,3} - M_{0,1}M_{0,3}} \end{pmatrix} \odot \begin{pmatrix} 0 \\ 0 \\ u_1u_2 \\ v_1v_2 \end{pmatrix}$$

– **Position 22**: $\mathcal{H}\Big(\theta^{E_1\times E_2}(P+Q) \odot \theta^{E_1\times E_2}(P-Q)\Big) =$

$$\begin{pmatrix} \mathbf{M_{1,0}}^2 + \mathbf{M_{2,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{1,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{2,0}}^2 \\ 0 \end{pmatrix} \odot \begin{pmatrix} (u_1^2 - v_1^2 - w_1^2)(u_2^2 - v_2^2 - w_2^2) \\ (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ (u_1^2 - v_1^2 + w_1^2)(u_2^2 - v_2^2 + w_2^2) \\ 0 \end{pmatrix} + 2u_2w_2 \begin{pmatrix} 0 \\ \mathbf{M_{0,0}M_{0,1} + M_{1,0}M_{1,1}} \\ \mathbf{M_{0,0}M_{0,1} - M_{1,0}M_{1,1}} \\ 0 \end{pmatrix} \odot \begin{pmatrix} 0 \\ u_1^2 - v_1^2 + w_1^2 \\ u_1^2 - v_1^2 + w_1^2 \\ 0 \end{pmatrix}$$

$$+ (-1)^{M_{0,1}=-M_{0,2}} \left(2u_1w_1 \begin{pmatrix} 0 \\ \mathbf{M_{0,0}M_{0,1} - M_{1,0}M_{1,1}} \\ \mathbf{M_{0,0}M_{0,1} + M_{1,0}M_{1,1}} \\ 0 \end{pmatrix} \odot \begin{pmatrix} 0 \\ u_2^2 - v_2^2 + w_2^2 \\ u_2^2 - v_2^2 + w_2^2 \\ 0 \end{pmatrix} + 4w_1w_2 \begin{pmatrix} 0 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{2,0}}^2 \\ \mathbf{M_{0,0}}^2 - \mathbf{M_{1,0}}^2 \\ \mathbf{M_{1,0}}^2 + \mathbf{M_{2,0}}^2 \end{pmatrix} \odot \begin{pmatrix} 0 \\ u_1u_2 \\ u_1u_2 \\ v_1v_2 \end{pmatrix} \right)$$

B Superglue in dimension 1

In this appendix, we apply the ideas underlying the Superglue formulae in dimension 1 to compute 2-isogenies between theta structure of elliptic curves. Using the symmetric elements detailed in section 3, we can compute the theta structure induced by a choice of (trivially symplectic) basis over $E[4] = \langle S', T' \rangle$. They are detailed in table 6.

Similarly to lemma 2, we can express the duplication formula over elliptic curve as a quadratic homogeneous map. More specifically, given θ^{E_1} a theta

$$\begin{aligned}
\langle (a+b:a-b),(1:1)\rangle &\implies \mathbf{M} = \begin{pmatrix} b & b \\ a & -a \end{pmatrix} \\
\langle (a+b:a-b),(1:-1)\rangle &\implies \mathbf{M} = \begin{pmatrix} a & -a \\ b & b \end{pmatrix} \\
\langle (1:1),(a+b:a-b)\rangle &\implies \mathbf{M} = \begin{pmatrix} a+b & -(a-b) \\ -(a-b) & a+b \end{pmatrix} \\
\langle (1:-1),(a+b:a-b)\rangle &\implies \mathbf{M} = \begin{pmatrix} a+b & -(a-b) \\ (a-b) & -(a+b) \end{pmatrix} \\
\langle (a+b:a-b),(a+\mathbf{i}b:a-\mathbf{i}b)\rangle &\implies \mathbf{M} = \begin{pmatrix} a+b & -(a-b) \\ -\mathbf{i}(a-b) & \mathbf{i}(a+b) \end{pmatrix} \\
\langle (a+b:a-b),(a-\mathbf{i}b:a+\mathbf{i}b)\rangle &\implies \mathbf{M} = \begin{pmatrix} a+b & -(a-b) \\ \mathbf{i}(a-b) & -\mathbf{i}(a+b) \end{pmatrix}
\end{aligned}$$

Table 6. Detailed list of the change of coordinate matrix of the different theta structures depending on the basis of $E[4]$.

structure induced by the 4-basis $\langle S', T'\rangle$ and let $\phi : E_1 \to E_2$ be the isogeny whose kernel is $\{0, [2]T'^{\}}$. Given $P \pm Q$ the barycentric coordinates of P and Q. Then,

$$\theta^{E_1}(P+Q) \odot \theta^{E_1}(P-Q) = \left(\mathbf{M} \cdot \begin{pmatrix} u \\ w \end{pmatrix}\right)^{\odot 2} - \left(\mathbf{M} \cdot \begin{pmatrix} v \\ 0 \end{pmatrix}\right)^{\odot 2}$$

$$\theta^{E_1}(P+Q) \odot \theta^{E_1}(P-Q) = [\mathbf{M}_0\mathbf{M}_0](u^2 - v^2) + [\mathbf{M}_1\mathbf{M}_1]w^2 + 2uw[\mathbf{M}_0\mathbf{M}_1]$$

where, similarly to prior, the different product of columns of the change of basis correspond to the evaluation of theta points related to $C = (0:1)$ which can be S, T or $S+T$ in the basis of $E[2]$ (See table 7).

columns	theta points		columns	dual theta points
$\mathbf{M}_0\mathbf{M}_0$	$\theta^{E_1}(0)\theta^{E_1}(0)$	$\iff$	$\widetilde{\mathbf{M}_0\mathbf{M}_0}$	$\widetilde{\theta}^{E_2}(\Phi(0))\widetilde{\theta}^{E_2}(\Phi(0))$
$\mathbf{M}_1\mathbf{M}_1$	$\theta^{E_1}(C)\theta^{E_1}(C)$	$\iff$	$\widetilde{\mathbf{M}_1\mathbf{M}_1}$	$\widetilde{\theta}^{E_2}(\Phi(0))\widetilde{\theta}^{E_2}(\Phi(C))$
$\mathbf{M}_0\mathbf{M}_1$	$\theta^{E_1}(0)\theta^{E_1}(C)$	$\iff$	$\widetilde{\mathbf{M}_0\mathbf{M}_1}$	$\widetilde{\theta}^{E_2}(\Phi(C'))\widetilde{\theta}^{E_2}(\Phi(C'))$

Table 7. Correspondence between change of basis matrix columns and theta points with $C = (0:1)$ and $C' = (1:\pm 1)$.

We therefore have the following 3 set of equations depending on the position of C in the basis.

Position	C	$\mathcal{H}\Big(\theta^{E_1}(P+Q)\odot\theta^{E_1}(P-Q)\Big)=$
0	S	$\begin{pmatrix}\mathbf{M}_{0,0}^2+\mathbf{M}_{1,0}^2\\ \mathbf{M}_{0,0}^2-\mathbf{M}_{1,0}^2\end{pmatrix}\begin{pmatrix}u^2-v^2+w^2\\ u^2-v^2-w^2\end{pmatrix}+4uw\begin{pmatrix}\mathbf{M}_{0,0}\mathbf{M}_{0,1}\\ 0\end{pmatrix}$
1	T	$\begin{pmatrix}\mathbf{M}_{0,0}^2+\mathbf{M}_{1,0}^2\\ \mathbf{M}_{0,0}^2-\mathbf{M}_{1,0}^2\end{pmatrix}\begin{pmatrix}u^2-v^2+w^2\\ u^2-v^2+w^2\end{pmatrix}\pm 2uw\begin{pmatrix}\mathbf{M}_{0,0}^2-\mathbf{M}_{1,0}^2\\ \mathbf{M}_{0,0}^2+\mathbf{M}_{1,0}^2\end{pmatrix}$
2	$S+T$	$\begin{pmatrix}\mathbf{M}_{0,0}^2+\mathbf{M}_{1,0}^2\\ \mathbf{M}_{0,0}^2-\mathbf{M}_{1,0}^2\end{pmatrix}\begin{pmatrix}u^2-v^2-w^2\\ u^2-v^2+w^2\end{pmatrix}+4uw\begin{pmatrix}0\\ \mathbf{M}_{0,0}\mathbf{M}_{0,1}\end{pmatrix}$

Using the changes of basis described in table 7, we can rewrite the above equations as below. Whether they are of cryptographic interest is for future study.

Position	C	$\mathcal{H}\Big(\theta^{E_1}(P+Q)\odot\theta^{E_1}(P-Q)\Big)=$
0	S	$\begin{pmatrix}b^2\big((u\pm w)^2-v^2\big)+a^2\big((u\mp w)^2-v^2\big)\\ 2ab(u^2-v^2-w^2)\end{pmatrix}$
1	T	$\begin{pmatrix}b^2\big((u\pm w)^2-v^2\big)+a^2\big((u\mp w)^2-v^2\big)\\ b^2\big((u\pm w)^2-v^2\big)-a^2\big((u\mp w)^2-v^2\big)\end{pmatrix}$
2	$S+T$	$\begin{pmatrix}2ab(u^2-v^2-w^2)\\ b^2\big((u\pm w)^2-v^2\big)+a^2\big((u\mp w)^2-v^2\big)\end{pmatrix}$

C Explicit algorithms of Section 5.2

In this appendix, we provide detailed algorithms of the algorithm in section 5.2, using Superglue formulae.

Algorithm 7 $\mathsf{GluingEval}(P\pm T_1'', J, \mathsf{precomp})$

Require: $P\pm T_1''$ the barycentric coordinates $P_1\pm T_{1,1}''$ and $P_2\pm T_{1,2}''$, J the dual of the theta point $\Phi(T_1'')$ on A and $\mathsf{precomp}$ the Superglue precomputations outputted in algorithm 3.

Ensure: Theta coordinates of $\Phi(P)$.

1: $x, y \leftarrow J$
2: $X_\pm, Y_\pm, Z_\pm, W_\pm \leftarrow \mathsf{EvaluateSuperglue}(P\pm T_1'', \mathsf{precomp})$ ▷ $22\mathbf{M}+2\mathbf{S}+18\mathbf{a}$
3: $X_{\Phi(P)}, Y_{\Phi(P)}, Z_{\Phi(P)}, W_{\Phi(P)} \leftarrow X_\pm\cdot y, Y_\pm\cdot y, Z_\pm\cdot x, W_\pm\cdot x$
4: $(x_{\Phi(P)}, y_{\Phi(P)}, z_{\Phi(P)}, w_{\Phi(P)}) \leftarrow \mathcal{H}(X_{\Phi(P)}, Y_{\Phi(P)}, Z_{\Phi(P)}, W_{\Phi(P)})$ ▷ $8\mathbf{a}$
5: **return** $(x_{\Phi(P)} : y_{\Phi(P)} : z_{\Phi(P)} : w_{\Phi(P)})$ ▷ Total cost: $26\mathbf{M}+2\mathbf{S}+26\mathbf{a}$

Algorithm 6 GluingCodomain(T_1'', T_2'')

Require: 8-torsion points $T_1'', T_2'' \in E_1 \times E_2$, and $\ker(\Phi) = \langle [4]T_1'' \rangle \oplus \langle [4]T_2'' \rangle$.

Ensure: Dual isogeneous theta null point $\widetilde{0_A}$, the "inverse" of the dual isogeneous theta null point $\widetilde{0_A}^{-1}$, the theta null point 0_A on A, the dual isogeneous theta point $\Phi(T_1'')$ on A, and the change-of-basis matrix $\mathbf{N}$ (re-used for point evaluation).

1: $T_1' \leftarrow [2](T_1'')$ ▷ $8\mathbf{M} + 4\mathbf{S} + 8\mathbf{a}$ using xDBL
2: $T_2' \leftarrow [2](T_2'')$ ▷ $8\mathbf{M} + 4\mathbf{S} + 8\mathbf{a}$ using xDBL
3: precomp $\leftarrow$ SuperglueCoefficients(T_1', T_2') ▷ $38\mathbf{M} + 7\mathbf{S} + 34\mathbf{a}$
4: Let $P_1 = (x_{T'_{1,1}} : 0 : z_{T'_{1,1}})$, $P_2 = (x_{T'_{1,2}} : 0 : z_{T'_{1,2}})$
5: Let $Q_1 = (x_{T'_{2,1}}, 0, z_{T'_{2,1}})$, $Q_2 = (x_{T'_{2,2}}, 0, z_{T'_{2,2}})$
6: $X_1, Y_1, Z_1, 0 \leftarrow$ EvaluateSuperglue$((P_1, P_2)$, precomp$)$ ▷ $16\mathbf{M} + 4\mathbf{S} + 12\mathbf{a}$
7: $X_2, Y_2, Z_2, 0 \leftarrow$ EvaluateSuperglue$((Q_1, Q_2)$, precomp$)$ ▷ $16\mathbf{M} + 4\mathbf{S} + 12\mathbf{a}$
8: $\alpha \leftarrow X_1 \cdot X_2$
9: $\beta \leftarrow Y_1 \cdot X_2$
10: $\gamma \leftarrow X_1 \cdot Z_2$
11: $\alpha^{-1} \leftarrow Y_1 \cdot Z_2$
12: $\beta^{-1} \leftarrow \gamma$
13: $\gamma^{-1} \leftarrow \beta$
14: $x \leftarrow X_1 \cdot \alpha^{-1}$
15: $y \leftarrow Z_1 \cdot \gamma^{-1}$
16: $J \leftarrow (x : x : y : y)$
17: $(a_2, b_2, c_2, d_2) \leftarrow \mathcal{H}(\alpha, \beta, \gamma, 0)$ ▷ $8\mathbf{a}$
18: $0_A \leftarrow (a_2 : b_2 : c_2 : d_2)$
19: $\widetilde{0_A} \leftarrow (\alpha : \beta : \gamma : 0)$
20: $\widetilde{0_A}^{-1} \leftarrow (\alpha^{-1} : \beta^{-1}, \gamma^{-1} : 0)$
21: **return** $\widetilde{0_A}, \widetilde{0_A}^{-1}, 0_A, J$, precomp ▷ Total cost: $94\mathbf{M} + 23\mathbf{S} + 82\mathbf{a}$

Algorithm 8 GluingEvalSpecial$(P, \widetilde{0_A}^{-1}$, precomp$)$

Require: The point $P \in E_1 \times E_2$ of the form $(P_1, 0)$ or $(0, P_2)$, $\widetilde{0_A}^{-1}$ the "inverse" of the dual isogeneous theta null point on A and precomp the Superglue precomputations outputted in algorithm 3..

Ensure: Theta coordinates of $\Phi(P)$.

1: $\alpha^{-1}, \beta^{-1}, \gamma^{-1}, 0 \leftarrow (x_{P_1} : 0 : z_{P_1})$
2: Let $P_1 = (x_{P_1} : 0 : z_{P_1})$, $P_2 = (x_{P_2} : 0 : z_{P_2})$
3: $X_P, Y_P, Z_P, 0 \leftarrow$ EvaluateSuperglue$((P_1, P_2)$, precomp$)$ ▷ $16\mathbf{M} + 4\mathbf{S} + 12\mathbf{a}$
4: $(X_{\Phi(P)}, Y_{\Phi(P)}, Z_{\Phi(P)}) \leftarrow X_P \cdot \alpha^{-1}, Y_P \cdot \beta^{-1}, Z_P \cdot \gamma^{-1}$
5: $(x_{\Phi(P)}, y_{\Phi(P)}, z_{\Phi(P)}, w_{\Phi(P)}) \leftarrow \mathcal{H}(X_{\Phi(P)}, Y_{\Phi(P)}, Z_{\Phi(P)}, 0)$ ▷ $6\mathbf{a}$
6: **return** $(x_{\Phi(P)} : y_{\Phi(P)} : z_{\Phi(P)} : w_{\Phi(P)})$ ▷ Total cost: $19\mathbf{M} + 4\mathbf{S} + 18\mathbf{a}$