Complete Analysis of Isogeny on Hessian Curve

Akash Rathor, Manoj Kumar, R. K. Mishra and Shivender Goswami

***Abstract*- Isogeny-based cryptography holds significant promise in the realm of post-quantum cryptography, primarily due to its perceived resilience against attacks from quantum computers, based on our current understanding of the underlying mathematical problems. However, it's essential to acknowledge the dynamic nature of post-quantum cryptography, where ongoing research may yield profound insights or lead to the development of novel cryptographic approaches. This study delves into the analysis of Hessian form curves within the framework of isogeny-based cryptography (IBC). We specifically investigate the computational costs associated with deriving Hessian form curves for constructing sections, particularly when employing compression functions. The square-root Velu method is utilized for handling Hessian form curves, and we introduce a novel formula for calculating the curve's coefficient at a specific point on a Hessian curve. Our results indicate that the operational costs of the Hessian form and the Montgomery curve are comparable. Furthermore, we propose the Hessian-Edwards hybrid model, optimizing Hessian-CSIDH and determining the coefficient for the codomain's curve using Edwards curves. According to our findings, various Isogeny-based cryptosystems can be implemented by leveraging Hessian curves.**

***Keywords*- Hessian curves, post-quantum cryptography, Isogeny, SIDH, CSIDH, Montgomery curves, square-root Velu formula.**

I. INTRODUCTION

The emergence of quantum computers proficient in executing Shor's algorithm has sparked a necessity for post-quantum cryptography (PQC) approaches, which could potentially replace current public-key cryptography systems. Isogeny-based cryptography (IBC) is among the PQC primitives actively under exploration and is considered a leading candidate due to its capacity to yield shorter key lengths compared to other PQC primitives. The CRS method originated from Stolbunov's rediscovery of Couveignes' initial use of isogenies for quantum-resistant cryptography [1]-[2]. However, the vulnerability of the CRS technique to quantum sub-exponential attacks [3]-[4] renders it unsuitable for practical applications. In response to these limitations, De Feo introduced the supersingular isogeny-based scheme.

Diffie-Hellman (SIDH), later refined by Jao [5], within the context of IBC for enhanced security and utility. The non-commutative nature of the endomorphism rings of supersingular curves provides protection against attacks outlined in [3]-[6]. Additionally, the quantum-exponential complexity of establishing an isogeny between two isogenous elliptic curves on a finite field underscores the reliability of SIDH, in contrast to the vulnerability of discrete logarithm problems to Shor's A The SIDH-based key encapsulation technique, Supersingular Isogeny Key Encapsulation (SIKE), has been proposed as an alternative for NIST PQC standardization [5]-[7], emerging as a substitute option for Round 3. Research by De Feo, Kieffer, and Smith [9] provides updated parameter selections and effective techniques for determining the CRS group action. Addressing parameter selection concerns in CRS, Castryck et al. introduced commutative SIDH (CSIDH) in [10]-[11]. While CSIDH's full key transfer is slower than SIDH, taking around 80 ms at a 128-bit traditional security level, it proves effective for digital signatures, as demonstrated by CSI-FiSh [12]-[13], signing messages in 390 ms. Although SIDH and CSIDH offer unique benefits, their common drawback is slower performance compared to other quantum-resistant algorithms. The implementation of IBC involves elliptic curve arithmetic and isogeny actions, with the isogeny's degree determined by the prime used in the scheme. The choice of prime q for cryptographic techniques relies on SIDH, which takes the form $q = f_A^{i_A} f_B^{i_B} l \pm 1$, where f_A and f_B are coprime and denotes the degrees of the isogenies used in the method. As determining isogenies becomes more challenging with increasing degrees, implementations often focus on isogenies of degrees 3 and 4. The selection of prime q for algorithms is influenced by CSIDH, represented as $q = 4 f_1 f_2 .... f_n - 1$, where f_i are odd primes. The introduction of the CSIDH system has led to a growing demand for efficient odd-degree isogeny methods. In [14]-[15], Costello and Hisil proposed a practical method for generating random isogenies of odd degree on Montgomery curves. Traditional approaches for computing f-isogeny involve significant fieldwork. The square-root Velu formula, which enables the computation of f-isogeny in $\tilde{O}(f)$ field operations, was first presented by Bernstein et al. in [16].

Manuscript received July 31, 2023; revised May 18, 2024.

Akash Rathor is a Ph.D. candidate of the Deparment of Mathematics and Statistics, Gurukula Kangri (Deemed to be University), Haridwar, Uttarakhand, India, 249404. (E-mail: akashrathor9760@gmail.com).

Manoj Kumar is an Associate Professor of the Deparment of Mathematics and Statistics, Gurukula Kangri (Deemed to be University), Haridwar, Uttarakhand, India, 249404. (Corresponding author to provide phone: +91 8755386009; E-mail: sdmkg1@gmail.com).

R.K. Mishra is a Professor of the Department of Applied Science and Humanities, G.L. Bajaj Institute of Technology and Management, Greater Noida, U.P., India, 201306. (E-mail: rkmsit@rediffmail.com).

Shivender Goswami is a Ph.D. candidate of the Deparment of Mathematics and Statistics, Gurukula Kangri (Deemed to be University), Haridwar, Uttarakhand, India, 249404. (E-mail: shivendrgoswami@gmail.com).

Our work focuses on optimizing the computation of larger odd-degree isogenies, which is crucial for implementing B-SIDH [17]-[18] and CSIDH. This is particularly pertinent given the current quantum analyses emphasizing the need for large odd-degree isogenies to ensure robust security. The choice of an elliptic curve shape is pivotal for facilitating efficient arithmetic operations. Montgomery curves, celebrated for their rapid isogeny estimation and efficient elliptic curve arithmetic, are extensively utilized in isogeny-based cryptography applications. They constitute the foundation of state-of-the-art implementations [19]-[20]. Twisted-Edwards curves, particularly when utilizing projective coordinates, are favored for their capability to map points from one curve to another, owing to their birational equivalence to Montgomery curves. Meyer et al. pioneered the use of Montgomery curves for isogeny determination and Twisted-Edwards curves for elliptic curve arithmetic, laying the groundwork in [21]-[22], which marked the initial application of Edwards curves in this context. This methodology was further developed in [23]-[24], where Montgomery curves were employed for elliptic curve mathematical computations, while Edwards curves were utilized for isogeny computation. However, as pointed out in [24] and [25], relying exclusively on Edwards curves for SIDH-based approaches proves to be less effective compared to the use of Montgomery curves alone. The utility of employing Edwards curves became evident with the implementation of CSIDH, harnessing larger odd-degree isogenies. Moreover, a valid formula enables the reconstruction of a curve parameter's image on Montgomery curves. Montgomery curves excel in efficiently estimating isogenies of any odd degree [14], while Edwards curves offer a valuable method for computing the image curve's coefficient. Consequently, in [26]-[27], CSIDH was developed using Twisted-Edwards curves to derive the image curve's coefficient and Montgomery curves for isogeny determination.

An enhanced formula for odd-degree isogenies was introduced in [28]-[29] using Edwards curves' -coordinates. Achieving a more efficient implementation of Edwards-only CSIDH compared to Montgomery-CSIDH [10]-[28] or Hybrid-CSIDH [24] involves modifying the formula introduced in [26]. This study emphasizes the potential performance enhancements achievable by employing different elliptic curve types in various isogeny-based methods. Therefore, it's crucial to evaluate implementation outcomes across a range of elliptic curves.

Isogeny, a surjective homomorphism between elliptic curves with finite-length kernels, is not confined to specific curve models. However, distinct curve models may entail different processing costs for isogeny calculations. Elliptic curves can be represented in various forms, including Weierstrass, Edwards, Hessian, and others. Currently, schemes like SIDH [31]-[32] and CSIDH [33]-[34] leverage Montgomery curves for rapid isogeny evaluation, contributing to the ongoing focus of IBC research. Edwards curves have also been explored for their potential in efficient isogeny computations [35]. Kim et al. utilized a system dependent on ϖ -coordinates [35].

Farashahi et al. [36] proposed ϖ -coordinates in 2010 for efficient field mathematics formulae on generalized-Hessian curves across a field with characteristics $q = 2$. Bernstein et al. [37] investigated group computation on Twisted Hessian curves. Zheng Tao et al. [38] introduced quicker isogeny estimation using a Twisted-Hessian form curve.

This research employs various methodologies, including the compression function, square root Velu's formula, and an Edwards curve-based formula, to facilitate the creation of the Hessian-Edwards hybrid model and curve coefficient. Our findings reveal that the Hessian model exhibits significant similarity to the Weierstrass curve, outperforming the Edwards curve, and the operational cost of the Hessian form aligns comparably with that of the Montgomery curve.

The paper is structured as follows: Section II provides a thorough review of Twisted Hessian curves, the SIDH-CSIDH scheme, and square root Velu's formula to establish background context. In Section III, we introduce the relationship between Montgomery and Twisted Hessian curves, followed by the presentation of the proposed scheme in Section IV. Section V includes figures, table comparisons, and bar chart analyses that contrast the Short Weierstrass, Edwards, and Hessian curves. Finally, Section VI concludes the paper and outlines future prospects.

II. PRELIMINARIES

This section provides the foundational information and background necessary for the paper. We begin by introducing two fundamental IBC protocols: the Supersingular Isogeny Diffie-Hellman (SIDH) protocol and the Commutative Supersingular Isogeny Diffie-Hellman (CSIDH) protocol. Additionally, we delve into the various variations of Hessian curves and their arithmetic properties.

A. SIDH Protocol

Assume coprime numbers f_A and f_B. Let i_A and i_B be positive integers satisfying $f_A^{i_A} \approx f_B^{i_B}$, and let $q = f_A^{i_A} f_B^{i_B} l \pm 1$ be a prime of the form with a specific integer cofactor l. Construct a supersingular curve E with $\mathbb{F}_{q^2}$ such that its order is $(f_A^{i_A} f_B^{i_B} l)^2$. When $f \in \{f_A, f_B\}$ and $i \in \{i_A, i_B\}$ are chosen, we can determine the complete f^i -torsion subgroup with order E over $\mathbb{F}_{q^2}$. Then, for the $f_A^{i_A}$ -torsion and $f_B^{i_B}$ -torsion subgroups, respectively, we select the basis $\{J_A, K_A\}$ and $\{J_B, K_B\}$.

Let's consider Alice and Bob's process for transferring a secret key. Let $\{J_A, K_A\}$ and $\{J_B, K_B\}$ be the bases for Bob and Alice, respectively. Alice randomly selects elements x_A and y_A from the ring $\mathbb{Z} / f_A^{i_A}\mathbb{Z}$, which is not divisible by x_A or y_A, to generate the key. Using Velu's formula, Alice computes the subgroup $\langle L_A \rangle = \langle [x_A]J_A + [y_A]K_A \rangle$ and a curve $E_A = E / \langle L_A \rangle$, followed by an isogeny $\phi_A : E \to E_A$ of degree

$f_A^{i_A}$, where $Ker\phi_A = \langle L_A \rangle$ is the chosen prime. She then sends $(E_A, \phi_A(J_B), \phi_A(K_B))$ to Bob. Bob performs the same operation as Alice, obtaining $(E_B, \phi_B(J_A), \phi_B(K_A))$. Alice determines the subgroup $\langle L_A^{'} \rangle = \langle [x_A]\phi_B(J_A) + [y_A]\phi_B(K_A) \rangle$ in order to determine the key. Alice creates curve $E_{AB} = E_B / \langle L_A^{'} \rangle$ employing Velu's formula. Using this subgroup, Bob creates a curve $E_{BA} = E_A / \langle L_B^{'} \rangle$ employing Velu's formula. The j-invariant of E_{AB}, denoted $j(E_{AB}) = j(E_{BA})$, serves as the shared secret between Alice and Bob.

B. CSIDH Protocol

A commutative group action over supersingular elliptic curves described over a finite field $\mathbb{F}_q$ forms the foundational concept of CSIDH. Let $\varepsilon ll_q(\ddot{O})$ denote the set of elliptic curves over $\mathbb{F}_q$ via the endomorphism ring $\ddot{O}$, where $\ddot{O}$ is an imaginary quadratic order. The class group $Cl(\ddot{O})$ interacts with $\varepsilon ll_q(\ddot{O})$ freely and transitively, as widely recognized. This interaction, known as the CM-action, involves the action of an ideal class $[p] \in Cl(\ddot{O})$ on an elliptic curve $E \in \varepsilon ll_q(\ddot{O})$ by $[p]E$.

Let's consider $f_1, ... f_n$ as a smaller separate odd prime, taking the form $p = 4 f_1 f_2 .... f_n - 1$. Assume E is a supersingular elliptic curve over $\mathbb{F}_q$, where $\mathbb{Z}[\pi]$, a commutative subring of the quaternion order End(E), serves as the endomorphism ring of E under $\mathbb{F}_q$. Given the trace value of Frobenius is zero, it follows that $E(\mathbb{F}_q) = q + 1$. Moreover, due to $\pi^2 - 1 = 0 \bmod f_i$, the ideal f_iO splits into $f_i \mathrm{O} = I_i \bar{I}_i$, where $I_i = (f_i, \pi - 1)$ and $\bar{I}_i = (f_i, \pi + 1)$ are prime ideals. The group action $[I_i]E$ and $[\bar{I}_i]E$ can be computed using Velu's formulas through the isogenies ϕ_{I_i} (and $\phi_{\bar{I}_i}$) over $\mathbb{F}_q$ (and $\mathbb{F}_q$). Let's suppose Alice and Bob aim to exchange a secret key. Alice selects a vector $(i_1, ..., i_n) \in \mathbb{Z}^n$, where each component lies within the interval $[-x, x]$ for some positive integer x. This vector represents an isogeny corresponding to the group action by class $[p] = [I_1^{i_1}, ..., I_n^{i_n}]$, where $I_i = (f_i, \pi - 1)$ is an ideal in the endomorphism ring. Alice then computes her public key $E_A := [p]E$ and sends it to Bob.

Similarly, Bob utilizes his private ideal and transmits his public key $E_B := [q]E$ to Alice. Upon receiving Bob's public key, Alice calculates determines $[p]E_B$. $[p]E_B$ and $[q]E_A$, and Bob computes 1. Alice and Bob can deduce a shared secret value from the elliptic curves due to commutativity, rendering $[p]E_B$ and $[q]E_A$ isomorphic.

C. Twisted Hessian Curve and their Arithmetic

The Twisted Hessian curve over the finite field K is represented as follows:

$H_{p,m} : pR^3 + S^3 + T^3 = mRST$

It is defined over the projective space $P^2(K)$, where elements p, m belong to and must satisfy $p(27p - m^3) \neq 0$. The affine form equation of $H_{p,m}$ is defined by $H_{p,m} : p + s^3 + t^3 = mst$, where $s = \frac{S}{R}$ and $t = \frac{T}{R}$. When $p = 1$, the curve becomes a Hessian curve. The j-invariant of the Hessian curve [38] of $H_{p,m}$ is $j(H_{p,m}) = \frac{m^3(m^3 + 216p)^3}{p(m^3 - 27p)^3}$.

Note that the point at infinity, denoted $\mathrm{O} = (0 : -1 : 1)$, serves as the neutral element. Therefore, the elements of $H_{p,m}(K)$ form an additive group with the identity O, and the inverse element of a point $J = (R : S : T)$ in $H_{p,m}(K)$ is $-J = (R : T : S)$.

For the $H_{p,m}$, two addition formulas are described in [36]-[39]. Let $(R_1 : S_1 : T_1)$ and $(R_2 : S_2 : T_2)$ be points of $H_{p,m}$. The standard addition formula (also known as the Sylvester Formula) is given by $R_3 = R_1^2 S_2 T_2 - R_2^2 S_1 T_1$, $S_3 = T_1^2 R_2 S_2 - T_2^2 S_1 R_1$, and $T_3 = S_1^2 R_2 R_2 - S_2^2 R_1 T_1$. The rotated formula is given by $S_3^{'} = S_2^2 S_1 T_1 - P R_1^2 R_2 T_2$ and $T_3^{'} = p R_2^2 R_1 S_1 - T_1^2 T_2 S_2$.

Both formulas are considered $R_3^{'} = T_2^2 R_1 T_1 - S_1^2 R_2 S_2$ complete because the resulting points $(R_3 : S_3 : T_3)$ and $(R_3^{'} : S_3^{'} : T_3^{'})$ are distinct [39].

D. Isomorphism

The Weierstrass curve and the Twisted Hessian curve $H_{p,m}$ defined over the same field undergo a transformation. The following substitutions will convert a Twisted Hessian curve $H_{p,m}$ specified by the formula:

$pR^3 + S^3 + T^3 = mRST$

into a Weierstrass curve by the substitutions $r = (m - p)\frac{S}{T}$ $s = (m - 2p)\frac{R}{T}$. By applying these changes, an equation for the Twisted Hessian curve $H_{p,m}$ becomes $s^2 = r^3 - 432\frac{p^2}{(m-p)^2} r$, which represents a Weierstrass curve E with coefficients coefficients $p = -432\frac{p^2}{(m-p)^2}$ and $q = 0$ emerged. The following substitutions, however, can be employed to transform a Weierstrass curve E specified by the equation $s^2 = r^3 + pr + q$ into a Twisted Hessian curve. The equation for E becomes

$S^2T = R^3 + pRT^2 + qT^3$ with the substitutions $R = \frac{r}{T}, S = \frac{y}{T}, p = -\frac{4}{3}p, m = -\frac{4}{27}q$, resulting in a Twisted Hessian curve $H_{p,m}$ of coefficients p and m.

The Twisted Hessian curve and Montgomery curve are isomorphic, although this relationship holds true only for fields where the characteristic is not equal to 2 or 3. The following substitutions can be utilized to transform a Twisted Hessian curve $H_{p,m}$ (specified by equation $pR^3 + S^3 + T^3 = mRST$) into a Montgomery curve $M: u = \frac{S}{R}, v = \frac{T}{R}$. The expression for $H_{p,m}$ changes to $v = u^3 + pu^2 + u$ as a result of these modifications, yielding a Montgomery curve M with coefficients $A = p$ and $B = 1$. The following replacements $R = r, S = s, T = B$, on the other hand, can be utilized to transform a Montgomery curve M, which is specified by the equation $Bs^2 = r^3 + Ar^2 + r$, into a Twisted Hessian curve. With these changes, the expression for the Montgomery curve M becomes $S^2T = R^3 + AR^2T + RT^2$, and we are left with a Twisted Hessian curve $H_{p,m}$ with coefficients $p = A$ and $m = 1$.

E. ϖ -Coordinates on Twisted Hessian Curve

In their research, Farashahi et al. [35] introduced a ϖ -coordinate system tailored for efficient and streamlined field arithmetic operations on generalized Hessian-form curves, defined over fields $\mathbb{F}$ where $q = 2$. Expanding upon their work, we extend the ϖ -coordinate scheme to Twisted Hessian-form curves. Specifically, we delineate the rational map on points of the curve $J \in H_{p,m}$ as $\varpi(J) = st$, which can be computed efficiently and satisfies $\varpi(J) = \varpi(-J)$.

Theorem [37]: Consider two points J_1 and J_2 in $H_{p,m}$, with ϖ_i denoting the ϖ -coordinate for points. Specifically, we have $\varpi_1 = \varpi(J_1)$ and $\varpi_2 = \varpi(J_2)$. Let $\varpi_0 = \varpi(J_1 - J_2), \varpi_3 = \varpi(J_1 + J_2)$ and $\varpi_4 = \varpi(2J_1)$

Consequently, we derived these differential addition methods:

$$\varpi_3\varpi_0 = \frac{\varpi_1^2\varpi_2^2 - mp\varpi_1\varpi_2 + p^2\varpi_1 + p^2\varpi_2}{(\varpi_1 - \varpi_2)^2}$$

$$\varpi_4 = \frac{\varpi_1(pm\varpi_1 - \varpi_1^3 - 2p^2)}{(m\varpi_1 - p)^2 - 4\varpi_1^3}.$$

Theorem [37]: Consider a Twisted Hessian curve $H_{p,m}$ defined over field K, and let $\varpi(s,t)$ denote the value of the ϖ -coordinate at any point (s,t) on the curve. We can then derive the following equations:

(i) The 3-isogeny ϕ_3 maps $H_{p,m}$ to $H_{\hat{p},\hat{m}}$. We have

$$\varpi(\phi_3) = \frac{-3(\varpi^3 + pm\varpi - p^2)}{\varpi^2} + m^2,$$

where the image curve coefficients are $\hat{p} = m^3 - 27p$ and $\hat{m} = 3m$.

(ii) The 3-isogeny ϕ_3 maps $H_{p,m}$ to $H_{\hat{p},\hat{m}}$ is

$$\varpi(\phi_3) = \frac{(\varpi^3 + pm\varpi - p^2)}{\varpi^2} + 3c^2 + cm,$$

where the image curve coefficients are $\hat{p} = m^2c + 3mc^2 + 9p$ and $\hat{m} = m + 6c$.

Theorem [37]:

Let $G = \mathrm{O} \cup \{\pm(\mu_1, \eta_1), ..., \pm(\mu_s, \eta_s)\}$ be a subgroup of $H_{p,m}$ with a size of $f = 2s + 1 \neq 0 \pmod 3$.

Suppose ϕ_l is the f -isogeny from $H_{p,m}$ to $H_{\hat{p},\hat{m}}$. with finite length kernel G, and $H_{\hat{p},\hat{m}}$ be the value of the ϖ -coordinate at any point (s,t) on $H_{p,m}$. Let $\varpi_i = \varpi(\mu_i, \eta_i)$ for $i = 1, ..., s$. We can derive the following equation:

$$\varpi(\phi_l) = \varpi\prod_{i=1}^{s}\frac{\varpi_i^2\varpi^2 + p^2\varpi + p^2\varpi_i - pm\varpi\varpi_i}{(\varpi - \varpi_i)^2}$$

the image curve coefficients are $\hat{p} = p^f$ and $\hat{m} = (1+2s)m\prod_{i=1}^{s}\varpi_i - 6p\sum_{i=1}^{s}\prod_{j\neq i}\varpi_j$.

III. RELATIONSHIP WITH MONTGOMERY CURVES

This section explores the relationship between the ϖ -coordinate on a Hessian curve and the corresponding r -coordinate on a Montgomery curve. The conversion between Montgomery curves and Hessian curves is straightforward. To transform a point $(R_M : T_M)$ into corresponding Hessian WZ -coordinates, we can utilize the ϖ -function as a compression method defined as: $(R_M : T_M) \to (\widehat{W} : T) = (T_M : R_M)$

The utilization of Edwards curves to compute the image curve's coefficient can enhance efficiency when applying arbitrary isogenies of odd-degree on Montgomery curves, as seen in CSIDH. Edwards curves can also improve the efficiency of Hessian isogenies due to the conversion between the Montgomery curve. Switching between Hessian curves and Montgomery curves requires a straightforward transformation of coordinates, while the conversion between Montgomery curves and Edwards curves incurs minimal cost.

IV. THE PROPOSED HESSIAN CURVE ISOGENY CALCULATIONS

A recent efficient approach for determining f -isogeny utilizing field operations $O(\sqrt{f})$ was developed by Bernstein et al. [19]. This approach alleviates the computational burden imposed by traditional Vélu formulas for determining f -isogenies. The traditional Vélu formula requires extensive field operations.

The calculation of polynomials whose roots are determined by a function via a cyclic group can be conceptualized as utilizing the Vélu formula. A polynomial is defined using G, a cyclic group with the generator J, with a finite subset S_1 in $\mathbb{Z}$

$$h_{S_1}(R) = \prod_{s_1 \in S_1} (R - f([s_1]J))$$

Given a curve $E(K)$ and a point $J \in E(K)$, where $G = \langle J \rangle$ is a kernel of an f-isogeny $\phi : E \to E'$, and $\phi([s_1]J)$ the r-coordinate of $[s_1]J$, which is obtained through scalar multiplication of $[s_1]J$.

Let M_p denote a Montgomery curve, and let $J \in M_p$ be a point of prime order f (not equal to 2). The isogeny $\phi : M_p \to M_{p'}$ with kernel $\langle J \rangle$ can be expressed using equation (3) as follows:

$$\phi(R) = \frac{R^f h_{S_1}(1/R)^2}{h_{S_1}(R)^2}$$

$$(R - r(J+K))(R - R(J-K)) = R^2 + \frac{N_1(r(J), r(K))}{F_0(r(J), r(K))} R + \frac{N_2(r(J), r(K))}{N_0(r(J), r(K))}$$

holds for all $J, K \in E$ such that $O \notin \{J, K, J+K, J-K\}$.

Here, $r(J)$ denotes the r-coordinate of the point J.

When E is described in affine Montgomery equation form $Bs^2 = r^3 + Ar^2 + r$, the polynomials N_0, N_1 and N_2 are defined as follows [19]:

$$N_0(R_1, R_2) = (R_1 - R_2)^2$$

$$N_1(R_1, R_2) = -2((R_1 R_2 + 1)(R_1 + R_2) + 2AR_1 R_2)$$

$$N_2(R_1, R_2) = (R_1 R_2 - 1)^2$$

We formulated the following biquadratic polynomials specifically tailored expressly for the form of Hessian curves H_m to facilitate the implementation of the square-root form. Likewise, the relationship between a point's ϖ-coordinate J, K, $J+K$ and $J-K$ on a Hessian curve can be expressed as follows:

$$(\bar{W} - \varpi(J+K))(\bar{W} - \varpi(J-K)) = \bar{W}^2 + \frac{H_1(\varpi(J), \varpi(K))}{H_0(\varpi(J), \varpi(K))} \bar{W} + \frac{H_2(\varpi(J), \varpi(K))}{H_0(\varpi(J), \varpi(K))}$$

For the curve H_m utilizing the ϖ-function, the polynomials H_0, H_1 and H_2 are defined in the following manner:

$$H_0(\bar{W}_1, \bar{W}_2) = (\bar{W}_1 \bar{W}_2 - 1)^2$$

$$H_1(\bar{W}_1, \bar{W}_2) = 2((\bar{W}_1 \bar{W}_2 + 1)(\bar{W}_1 + \bar{W}_2) + 2\bar{C}\bar{W}_1 \bar{W}_2 + 4\bar{W}_1 \bar{W}_2)$$

$$H_2(\bar{W}_1, \bar{W}_2) = (\bar{W}_1 - \bar{W}_2)^2 \quad \text{Where } \bar{C} = c + \frac{1}{c} - 2.$$

Proposition 1 (Square-root formula on Hessian curves):

Consider a point J on the Hessian form curve H_m over a finite field ($\mathbb{F}_q$), which is of prime order. Additionally, let a compression function ϖ be applied on the points of H_m. This function is defined such that if $K \in H_m$ is compressed then $\varpi(K) = \bar{W}$. An isogeny ϕ is defined with finite length kernel $\langle J \rangle$, and it is defined as $\phi : H_m \to H_{m'}$; the calculation of $\varpi(\phi(K))$ proceeds as follows:

$$\varpi(\phi(\bar{W})) = \frac{\bar{W}^l h_{S_1}(\bar{W})^2}{h_{S_1}(1/\bar{W})^2}$$

Where $S_1 = \{1, 3, ..., f-2\}$ and $m' = m^l \cdot h_{S_1}(-m)^2 / h_{S_1}(-1/m)^2$.

A. Recovering the Curve Coefficient

We have outlined a methodology for recovering the coefficients of the Hessian form curve using the ϖ-coordinate of J, K and $J-K$ points on the Hessian curve. Additionally, we have employed biquadratic polynomials for this purpose. By leveraging the Montgomery ladder, $J_A - K_A$ and $J_B - K_B$ are regarded as pivotal for expedited kernel evaluation in the implementation of SIDH. $\phi_A(J_B - K_B)$ and $\phi_B(J_A - K_A)$ are computed and exchanged to evaluate the shared key potential, which may be viewed as enlarging the length of the public key. By utilizing the fact that the coefficient 'a' of the Montgomery curve relates with the r-coordinates of J, K and $J-K$ for $J, K \in M_p$. Therefore, $(\phi_B(J_A), \phi_B(K_A), \phi_B(J_A - K_A))$ and $(\phi_A(J_B), \phi_A(K_B), \phi_A(J_B - K_B))$ have been interchanged in the protocol during and after obtaining the public key, facilitating the recovery of the coefficient through this relationship. Similarly, for Hessian form curves, coefficients can also be generated through analogous relationships. Let H_m be a Hessian curve utilizing ϖ as a compression function. For J, K and $J-K$ in H_m, let $\varpi(J) = \varpi_J, \varpi(K) = \varpi_K$, $\varpi(J+K) = \varpi_p$ and $\varpi(J-K) = \varpi_s$. Then the following relations hold:

$$\varpi_p \varpi_s = \frac{\varpi_1^2 \varpi_2^2 - m\varpi_1 \varpi_2 + \varpi_1 + \varpi_2}{(\varpi_1 - \varpi_2)^2}$$

$$-m\varpi_1 \varpi_2 + \varpi_1^2 + \varpi_2^2 + \varpi_1 + \varpi_2 = \varpi_p \varpi_s (\varpi_1 - \varpi_2)^2$$

$$m = \frac{(\varpi_1 - \varpi_2)^2 + \varpi_1^2 + \varpi_2^2 + \varpi_1 + \varpi_2}{\varpi_1 \varpi_2}$$

B. The Hessian-Edwards hybrid model and computational costs

It is crucial to highlight the advantageous link between Montgomery curves and Edwards curves by leveraging the square-root Velu formula for computing the image coefficient of Montgomery curves [16]. This approach not only accelerates computations but also facilitates the determination of the image curve coefficient through the formulation of the square-root Velu formula for isogeny calculation. Montgomery curves solely necessitate the computation of $h_{S_1}(1)$ and $h_{S_1}(-1)$, whereas Hessian curves mandate the computation of $h_{S_1}(-m)$ and $h_{S_1}(-1/m)$ for m in $\mathbb{F}_q$. Indeed, given the absence of cost associated with converting between Montgomery and Hessian

curves, the same concept can be seamlessly applied to Hessian curves as well. Furthermore, by incorporating related Edwards curves, it becomes feasible to optimize the computation of the curve image coefficient in Hessian-form curves further. This approach extends the efficiency gains achieved through leveraging interrelationships between different curve forms, enhancing the overall performance of cryptographic protocols.

Suppose J is a point on a Hessian curve H_m with prime order $f \neq 2$. Let ϖ be a compression function for points on H_m, and let $\hat{m} = \frac{1}{4}(m + \frac{1}{m} - 2)$, and $\phi : H_m \to H_{m'}$ be a quotient isogeny with kernel $\langle J \rangle$. Then, we can compute $\hat{m}' = c/(1-c)$ as $\hat{m}' = \frac{1}{4}(m' + \frac{1}{m'} - 2)$, where $c = \left(\frac{\hat{m}+1}{\hat{m}}\right)^{l} \left(\frac{h_{S_1}(1)}{h_{S_1}(-1)}\right)^{8}$. It is crucial to remember that utilizing Edwards curves to reconstruct curve image coefficients in the Hessian-Edwards hybrid model not only enhances efficiency by estimating $h_{S_1}(1)$ and $h_{S_1}(-1)$ rather than $h_{S_1}(-m)$ and $h_{S_1}(-1/m)$, but also eliminates the need for coefficient translation. The image curve's percent value is calculated using the isogeny formula for Hessian curves in terms of the domain curve H_m. Thus, since $\hat{m}' = \frac{1}{4}(m' + \frac{1}{m'} - 2)$ is utilized for elliptic curve computations

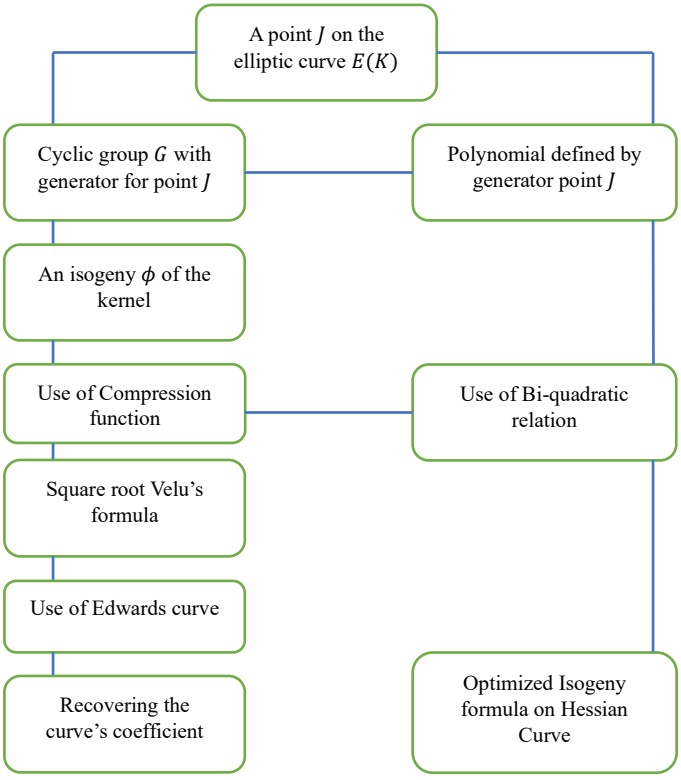

Fig 1 Graphical representation of proposed scheme

and m' must be preserved to proceed with isogeny evaluations, coefficient transition becomes necessary. However, when employing Edwards curves to recover the Hessian curve coefficient, the formula involves $\hat{m}$ and $\hat{m}'$, eliminating the need for coefficient transition and the necessity to maintain m for further isogeny evaluation. A graphical depiction of the proposed scheme is provided.

V. RESULTS

This section delves into the efficiency of Isogeny-Based Cryptography (IBC). We assessed the techniques by implementing them in Python and measuring their runtimes on a system running Windows 10 with a single core of an Intel Core i5-1035 CPU. The BIOS version used was X415A from American Megatrends Inc., with an SMBIOS version of 3.2 maintained throughout the compilation process. Finally, we will compare the computational costs of biquadratic polynomials for Montgomery and Hessian curves. Regarding the computation of f-isogenies, two phases are involved: isogeny evaluation and obtaining the image curve's coefficient. In Table I, the term "eval" denotes isogeny evaluation, while "coeff" pertains to the determination of the image curve's coefficient. Moreover, in Table I, "Hessian-Edwards" indicates the scenario where Hessian curves are utilized for isogeny evaluation and Edwards curves are utilized for coefficient determination. Table II illustrates the computational expenses of various components of isogeny-based cryptography using Montgomery and Hessian curves. The term "f-isogeny eval" denotes the evaluation of a f-isogeny, whereas "f-isogeny coeff" refers to the computation of the coefficient of the f-isogenous image curve. "CoeffTrans" specifies the cost of translating coefficients for effective elliptic curve computations, which is relevant only for Hessian curves. The term "Mont" pertains to the Montgomery curve, while "Mont-Edwards Hybrid" refers to a hybrid approach described in [39, 40], where Montgomery curves are utilized for elliptic curve computations and isogeny evaluation, while Edwards curves are employed for determining the image curve's coefficient. In the table, "m_1" indicates field multiplication, "s_1" denotes field squaring, and "c_1" signifies field coefficient multiplication.

The Hessian-Edwards Hybrid method combines Hessian curves for elliptic curve arithmetic and isogeny evaluation with Edwards curves for computing the coefficient of the image curve. Fig 2 depicts a pie chart comparing the performance of Montgomery, Hessian, and Hessian-Edwards Curves. Fig 3 presents the comparison among short Weierstrass, Edwards, and Hessian forms of elliptic curves over a finite field. For this comparison, set the coefficients of each of those curves equal to 1. Subsequently, we computed the runtime of the code to determine the 2, 3-torsion points for different field characteristics q of the form $q = 2^l 3^m - 1$, where l, m are integers. Table III presents the runtime for the Weierstrass, Edwards, and Hessian curves with fixed values of the coefficient $a = 1$ and $b = 1$ on different prime characteristics.

Fig 4 illustrates the comparison between short Weierstrass, Edwards, and Hessian forms of elliptic curves with constant fields with characteristic $2^6 3^1 - 1$. For this comparison, we utilized Python programming and set the value of the coefficient 'a' of each of those curves equal to 0. Subsequently, we computed the code's runtime to find the 2, and 3-torsion points for different values of the coefficient 'b'. Table IV showcases the runtime for the Weierstrass, Edwards, and Hessian curves with a fixed constant field of characteristic $2^6 3^1 - 1$ and $a = 0$ on different values of coefficient b. Fig 5 illustrates the comparison between short Weierstrass, Edwards, and Hessian forms of elliptic curves with constant fields with characteristic $2^6 3^1 - 1$. For this comparison, we used Python programming and set the value of the coefficient 'b' of each of those curves equal to 0. Subsequently, we computed the code's runtime to find the 2 and 3-torsion points for different values of the coefficient 'a'. Table V showcases the runtime for the Weierstrass, Edwards, and Hessian curves with a fixed constant field of characteristic $2^6 3^1 - 1$ and $b = 0$ on different values of coefficient a.

VI. CONCLUSION

In the examination of computational costs involved in isogeny on Hessian-form curves for constructing sections, we utilize compression functions. The square-root Velu method is employed for handling Hessian-form curves, introducing an innovative formula for calculating the curve's coefficient at a specific point on the Hessian curve. Referring to Table I, the computation costs for f-eval and f-coeff for Montgomery, Hessian, and Hessian-Edwards curves are determined as 14, 20, and 14 operations, respectively. Fig 3, Fig 4, and Fig 5 clearly illustrate the resemblance of the Hessian model to the Weierstrass curve and its superior performance over the Edwards curve. Our results indicate that the operational costs of the Hessian form and the Montgomery curve are comparable Furthermore, we introduced the Hessian-Edwards hybrid model by optimizing Hessian-CSIDH and computing the coefficient for the image's curve using Edwards curves. According to our findings, implementing IBC using Hessian curves is feasible.

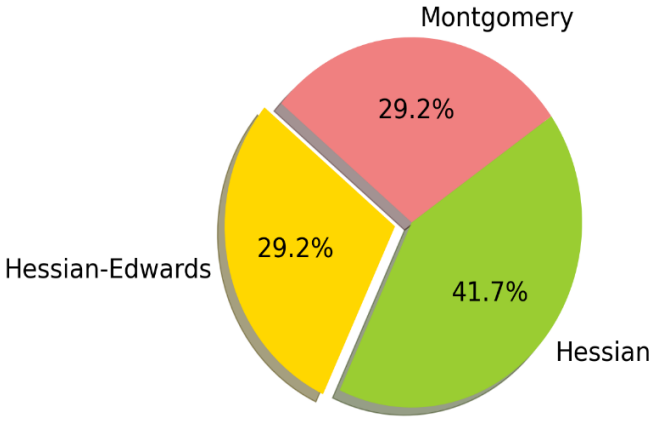

Fig 2 Pie-Chart comparing the performance of Montogomery, Hessian, and Hessian-Edwards Curves

Table I The expenses associated with computing biquadratic polynomials

$l-$ eval	Curves	Montgomery	Hessian	Hessian-Edwards
	Computation	$h_S(Q), h_S(1/Q)$	$h_S(Q), h_S(-1/Q)$	$h_S(Q), h_S(-1/Q)$
	Cost	$7m_1 + 2s_1$	$7m_1 + 3s_1$	$7m_1 + 2s_1$
$l-$ coeff	Computation	$h_S(1), h_S(-1)$	$h_S(c), h_S(-1/c)$	$h_S(1), h_S(-1)$
	Cost	$3m_1 + 2s_1$	$7m_1 + 3s_1$	$3m_1 + 2s_1$

Table II Computational cost of building-blocks of isogeny-based cryptography on Hessian curves and Montgomery curves

	Mont [10]-[16]	Mont-Edwards hybrid model	w -compression function	Hessian-Edwards Hybrid model
$2-$isogeny	$4m_1 + 2s_1$	$4m_1 + 2s_1$	$7m_1 + 2s_1 + 4c_1$	$7m_1 + 2s_1 + 4c_1$
$3-$isogeny	$6m_1 + 5s_1$	-	$5m_1 + 4s_1 + 5c_1$	-
$l-$isogeny eval	$4s'm_1 + 2s_1$	$4s'm_1 + 2s_1$	$7m_1 + 3s_1$	$7m_1 + 3s_1$
$l-$isogeny coeff	$(6s' - 2)m_1 + 3s_1$	$2s'm_1 + 6s_1 + 2w(l)$	$7m_1 + 3s_1$	$2s'm_1 + 6s_1 + 2w(l)$

Table III Runtime (in second) of building-blocks of isogeny-based cryptography on Weierstrass, Edwards and Hessian curves (Value of the coefficient $a = 1$ and $b = 1$)

	Runtime (in second)		
Prime Number	Weierstrass Curve	Edwards Curve	Hessian Curve
$2^0 3^1$ - 1	1.20837	0.00618	0.00755
$2^2 3^0$ - 1	0.01054	0.00000	0.00666
$2^3 3^0$ - 1	0.01718	0.01616	0.02071
$2^5 3^0$ - 1	0.18353	0.40964	0.21954
$2^7 3^0$ - 1	2.37102	7.99440	4.79775
$2^1 3^1$ - 1	0.01130	0.00735	0.00656
$2^2 3^1$ - 1	0.01040	0.04411	0.03323
$2^1 3^2$ - 1	0.03771	0.07858	0.06862
$2^3 3^2$ - 1	0.70893	2.23435	1.43595
$2^6 3^1$ - 1	5.47615	21.46011	12.22071

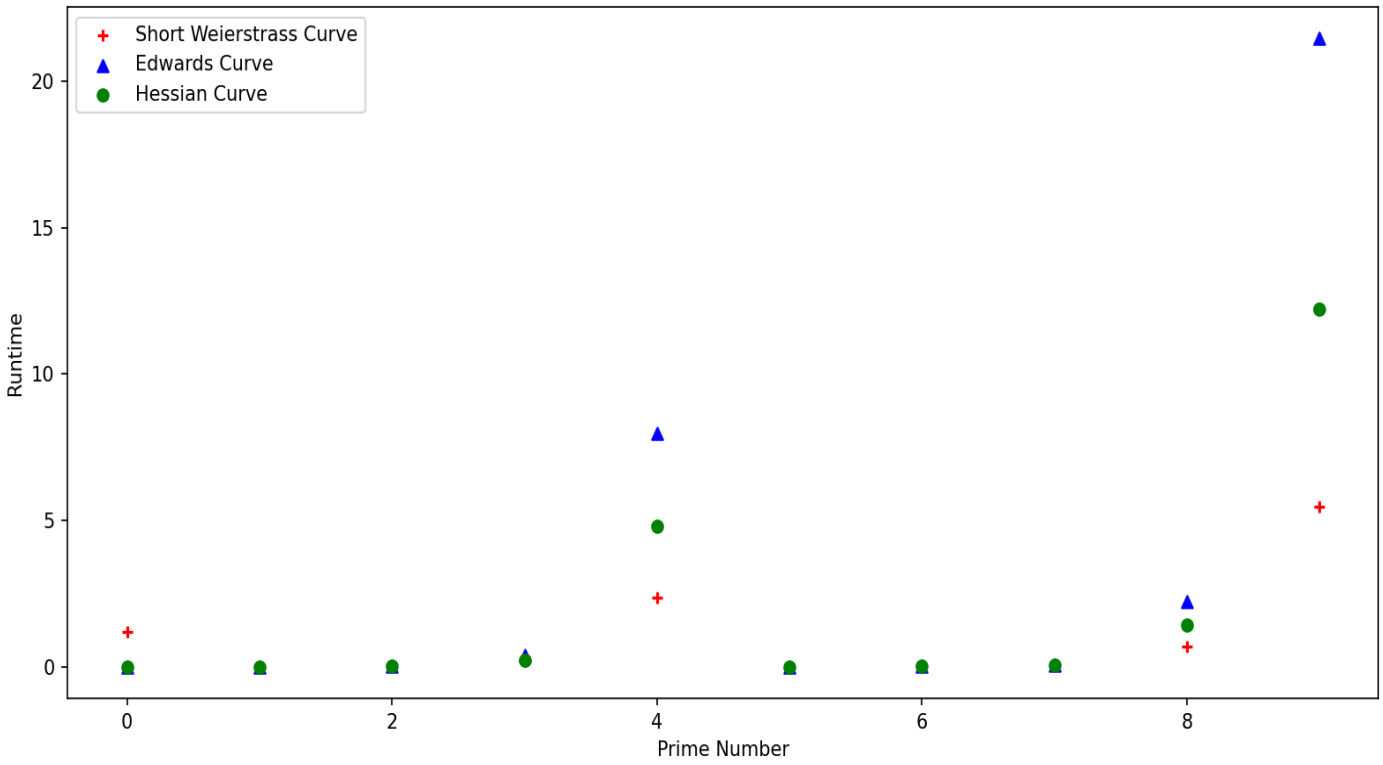

Fig 3 Graph of comparison between Short Weierstrass, Edwards and Hessian Curve

Table IV Runtime (in second) of building-blocks of isogeny-based cryptography on Weierstrass, Edwards and Hessian curves (Constant field with characteristic $2^6 3^1 - 1$ and $a = 0$)

	Runtime (in second)		
Values of b	Weierstrass Curve	Edwards Curve	Hessian Curve
0	8.74824	5.52195	9.04185
1	9.49638	13.63152	15.61121
2	8.95949	12.86678	9.31599
3	9.28498	13.65476	9.32817
4	10.61281	13.94777	9.01942
5	9.10469	13.21486	9.20848
6	8.90128	14.12703	8.63578
7	8.84155	13.98525	8.94770
8	9.76750	13.15157	9.49855
9	9.36304	15.78667	8.54008

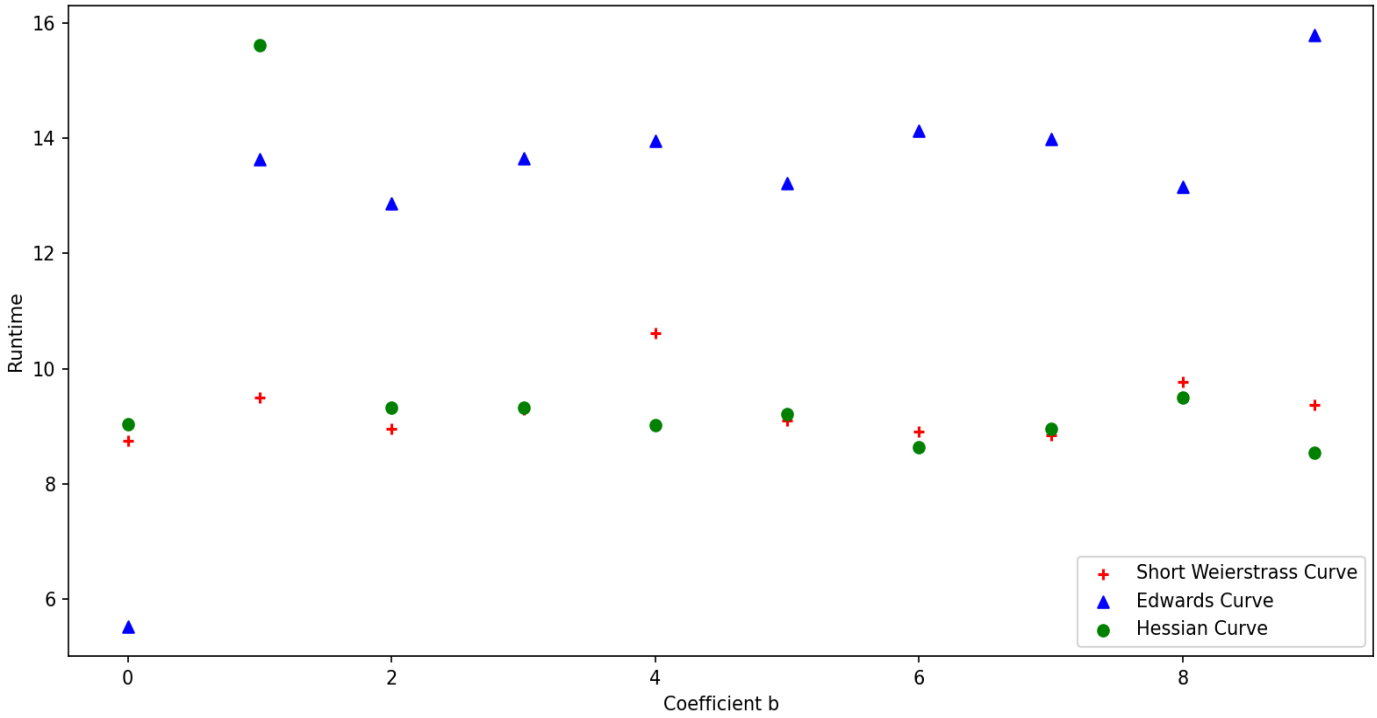

Fig 4 Graph of the comparison between Short Weierstrass, Edwards and Hessian Curve over finite field (const.)

Table V Runtime (in second) of building-blocks of isogeny-based cryptography on Weierstrass, Edwards and Hessian curves (Constant field with characteristic $2^6 3^1 - 1$ and $b = 0$)

	Runtime (in second)		
Values of a	Weierstrass Curve	Edwards Curve	Hessian Curve
0	8.07339	4.67914	8.24526
1	7.52350	15.32732	17.70590
2	7.56521	14.79981	17.67328
3	7.56031	16.57560	17.50555
4	7.59240	16.55787	17.60001
5	8.55207	16.51754	17.66080
6	7.59787	16.47829	18.00116
7	7.56619	16.78322	19.57871
8	7.62691	16.72845	17.64307
9	7.49035	16.56309	17.66482

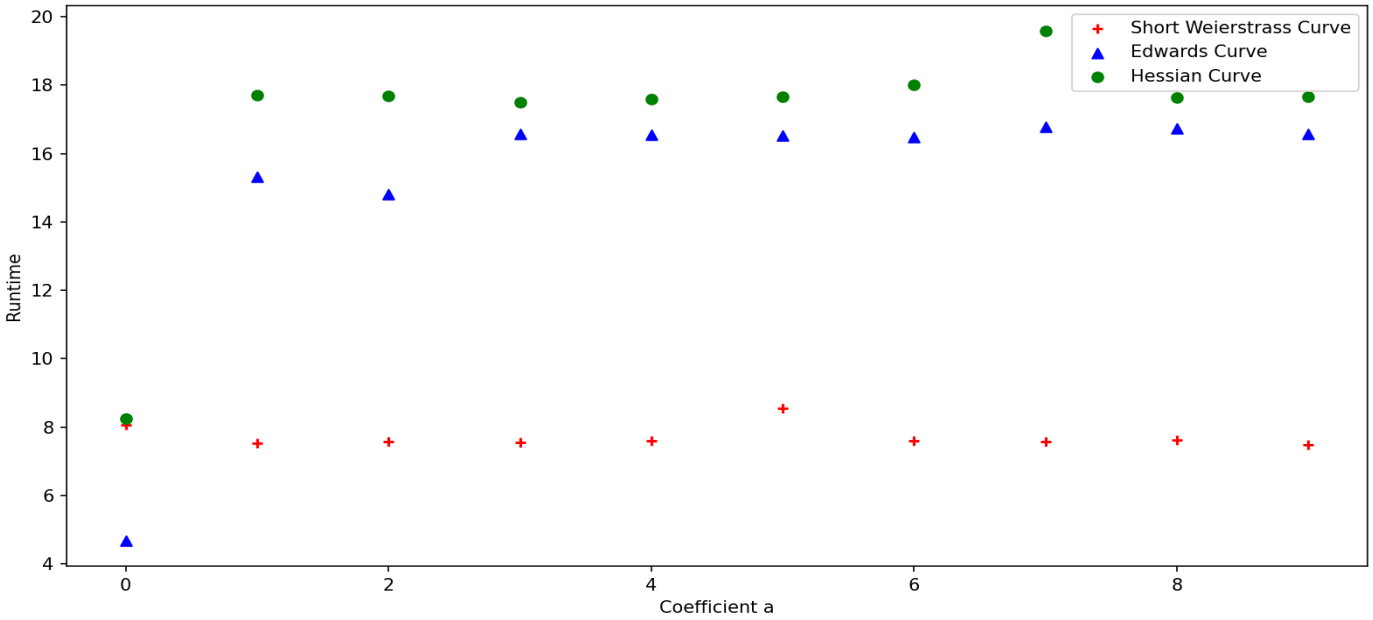

Fig 5 Graph of the comparison between Short Weierstrass, Edwards and Hessian Curve finite field (Const.)

REFERENCES

[1] J.M. Couveignes, "Hard homogeneous spaces," *Cryptology ePrint Archive*, 2006. Available: https://eprint.iacr.org/2006/291.pdf

[2] A. Stolbunov, "Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves," *Adv. Math. Commun.*, vol. 4, no. 2, pp. 215-235, 2010.

[3] A. Childs, D. Jao, and V. Soukharev, "Constructing elliptic curve isogenies in quantum subexponential time," *Journal of Mathematical Cryptology*, vol. 8, no. 1, 2014.

[4] W. Castryck, M. Houben, S.P. Merz, M. Mula, S. van Buuren, and F. Vercauteren, "Weak instances of class group action-based cryptography via self-pairings," *Cryptology ePrint Archive*, 2023. Available: https://eprint.iacr.org/2023/549.pdf

[5] D. Jao, and L. De Feo, "Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies," *In Post-Quantum Cryptography: 4th International Workshop*, pp. 19-34, 2011.

[6] W. Beullens, L. De Feo, S.D. Galbraith, and C. Petit, "Proving knowledge of isogenies: a survey," *Designs, Codes and Cryptography*, pp. 1-32, 2023. Available: https://eprint.iacr.org/2023/671.pdf

[7] G. Adj, J.J. Chi-Domínguez, and F. Rodríguez-Henríquez, "Karatsuba-based square-root Vélu's formulas applied to two isogeny-based protocols," *Journal of Cryptographic Engineering*, vol. 13, no. 1, pp. 89-106, 2023.

[8] D. Jao, R. Azarderakhsh, M. Campagna, C. Costello, L. De Feo, B. Hess, A. Jalali, B. Koziel, B. LaMacchia, P. Longa, and M. Naehrig, " Super-singular isogeny key encapsulation," *Submission to the NIST Post-Quantum Standardization Project*, 2017.

[9] L. De Feo, J. Kieffer, and B. Smith, "Towards practical key exchange from ordinary isogeny graphs," *In Advances in Cryptology–ASIACRYPT 2018: 24th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 365-394, 2018.

[10] W. Castryck, T. Lange, C. Martindale, L. Panny, and J. Renes, "CSIDH: an efficient post-quantum commutative group action" *In Advances in Cryptology–ASIACRYPT 2018: 24th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 395-427, 2018.

[11] Ren Guo-Xi, "Research on a Convolutional Neural Network Method for Modulation Waveform Classification," IAENG International Journal of Computer Science, vol. 50, no.3, pp875-882, 2023

[12] W. Beullens, T. Kleinjung, and F. Vercauteren, "CSI-FiSh: efficient isogeny-based signatures through class group computations," *In Advances in Cryptology–ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 227-247, 2019.

[13] Mohammad Ivan Azis, "A BEM for Transient Anisotropic Diffusion Convection Equation of Variable Coefficients," IAENG International Journal of Applied Mathematics, vol. 53, no.3, pp1107-1113, 2023

[14] C. Costello, and H. Hisil, "A simple and compact algorithm for SIDH with arbitrary degree isogenies," *In Advances in Cryptology–ASIACRYPT 2017: 23rd International Conference on the Theory and Applications of Cryptology and Information Security*, pp. 303-329, 2017.

[15] D. Jacquemin, A. Mukherjee, S.S. ROY, and P. Kutas, "Towards a constant-time implementation of isogeny-based signature", *Cryptology ePrint Archive*, 2023. Available: https://eprint.iacr.org/2023/807.pdf

[16] D.J. Bernstein, L. De Feo, A. Leroux, and B. Smith, "Faster computation of isogenies of large prime degree," *Open Book Series*, vol. 4, no. 1, pp. 39-55, 2020.

[17] C. Costello, "B-SIDH: supersingular isogeny Diffie-Hellman using Twisted torsion," *IACR Cryptol, ePrint Arch*, 2019. Available: https://eprint.iacr.org/2019/1145.pdf

[18] S. Mayo, "Automorphisms of the Super-singular Isogeny Graph," *arXiv preprint arXiv:2305.08158*, 2023. Available: https://arxiv.org/pdf/2305.08158.pdf

[19] P. Longa, "A note on post-quantum authenticated key exchange from supersingular isogenies," *Cryptology ePrint Archive*, 2018. Available: https://eprint.iacr.org/2018/267.pdf

[20] C. Costello, P. Longa, and M. Naehrig, "Efficient algorithms for supersingular isogeny Diffie-Hellman," *In Advances in Cryptology–CRYPTO 2016: 36th Annual International Cryptology Conference*, pp. 572-601, 2016.

[21] M. Meyer, S. Reith, and F. Campos, "On hybrid SIDH schemes using Edwards and Montgomery curve arithmetic," *Cryptology ePrint Archive*, 2017. Available: https://eprint.iacr.org/2017/1213.pdf

[22] W. Ghantous, F. Pintore, and M. Veroni, "Efficiency of SIDH-based signatures," *Cryptology ePrint Archive*, 2023. Available: https://eprint.iacr.org/2023/433.pdf

[23] Yung-Ning Cheng, and Kou-Huang Chen, "Study for Contradictory Pairwise Comparison Matrices," IAENG International Journal of Applied Mathematics, vol. 53, no.3, pp1138-1147, 2023

[24] S. Kim, K. Yoon, J. Kwon, Y.H. Park, and S. Hong, "New hybrid method for isogeny-based cryptosystems using Edwards curves," *IEEE Transactions on Information Theory*, vol. 66, no. 3, pp. 1934-1943, 2019.

[25] J.W. Bos, and S.J. Friedberger, "Arithmetic considerations for isogeny-based cryptography," *IEEE Transactions on Computers*, vol. 68, no. 7, pp. 979-990, 2018.

[26] M. Meyer, and S. Reith, "A faster way to the CSIDH," *In Progress in Cryptology–INDOCRYPT 2018: 19th International Conference on Cryptology in India*, pp. 137-152, 2018.

[27] Zhongfeng Li, Yingxin Wei, and Lidong Wang, "Active Event-Triggered Fault-Tolerant Control Design for Switched Pure-Feedback Nonlinear Systems," Engineering Letters, vol. 31, no.3, pp896-905, 2023

[28] S. Kim, K. Yoon, Y.H. Park, and S. Hong, "Optimized method for computing odd-degree isogenies on Edwards curves," *In Advances in Cryptology–ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 273-292, 2019.

[29] T. Moriya, H. Onuki, Y. Aikawa, and T. Takagi, "The generalized montgomery coordinate: A new computational tool for isogeny-based cryptography," Cryptology ePrint Archive, 2022. Available: https://eprint.iacr.org/2022/150.pdf

[30] B. Wesolowski, "The supersingular isogeny path and endomorphism ring problems are equivalent," *In 2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, IEEE, 2022. Available: https://eprint.iacr.org/2021/919.pdf, pp. 1100-1111

[31] D. Jao, and L. De Feo, "Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies," *In Post-Quantum Cryptography: 4th International Workshop*, pp. 19-34, 2011.

[32] Septia Devi Prihastuti Yasmirullah, Bambang Widjanarko Otok, Jerry Dwi Trijoyo Purnomo, and Dedy Dwi Prastyo, "Parameter Estimation of Spatial Error Model -Multivariate Adaptive Generalized Poisson Regression Spline," Engineering Letters, vol. 31, no.3, pp1265-1272, 2023

[33] W. Castryck, T. Lange, C. Martindale, L. Panny, and J. Renes, "CSIDH: an efficient post-quantum commutative group action," *In Advances in Cryptology–ASIACRYPT 2018: 24th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 395-427, 2018.

[34] M. Khairudin, R. Mahaputra, M. Luthfi Hakim, Asri Widowati, B Rahmatullah, and A. A. M. Faudzi, "Choosing the Quality of Two Dimension Objects by Comparing Edge Detection Methods and Error Analysis," IAENG International Journal of Computer Science, vol. 50, no.3, pp960-969, 2023

[35] S. Kim, K. Yoon, Y.H. Park, and S. Hong, "Optimized method for computing odd-degree isogenies on Edwards curves," *In Advances in Cryptology–ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 273-292, 2019.

[36] R.R. Farashahi, and M. Joye, "Efficient arithmetic on Hessian curves," *In Public Key Cryptography–PKC 2010: 13th International Conference on Practice and Theory in Public Key Cryptography*, pp. 243-260, 2010.

[37] D.J. Bernstein, C. Chuengsatiansup, D. Kohel, and T. Lange, "Twisted hessian curves," *In Progress in Cryptology-LATINCRYPT 2015: 4th International Conference on Cryptology and Information Security in Latin America*, pp. 269-294, 2015.

[38] Z. Tao, Z. Hu, and Z. Zhou, "Faster isogeny computation on Twisted Hessian curves," *Applied Mathematics and Computation*, vol. 444, p.127823, 2023.

[39] H. Hısıl, "Elliptic curves, group law, and efficient computation," PhD thesis, Queensland University of Technology, 2010. Available: http://eprints. qut. edu. au/33233/. 9

[40] F.L. Perez Broon, T. Dang, E. Fouotsa, and D. Moody, "Isogenies on Twisted Hessian curves," *Journal of Mathematical Cryptology*, vol. 15, no. 1, pp. 345-358, 2021.