Supersingular Hashing using Lattès Maps

Daniel Larsson

Abstract

In this note we propose a variant (with four sub-variants) of the Charles–Goren–Lauter (CGL) hash function using Lattès maps over finite fields. These maps define dynamical systems on the projective line. The underlying idea is that these maps "hide" the j-invariants in each step in the isogeny chain, similar to the Merkle–Damgård construction. This *might* circumvent the problem concerning the knowledge of the starting (or ending) curve's endomorphism ring, which is known to create collisions in the CGL hash function.

Let us, already in the abstract, preface this note by remarking that we have not done any explicit computer experiments and benchmarks (apart from a small test on the speed of computing the orbits), nor do we make any security claims. Part of the reason for this is the author's lack of competence in complexity theory and evaluation of security claims. Instead this note is only meant as a presentation of the main idea, the hope being that someone more competent will find it interesting enough to pursue further.

1 Introduction

In 2006 Charles and Lauter (published in 2009 together with Goren) proposed a hash function (colloquially referred to as the CGL hash function) based on walks in supersingular isogeny graphs. These graphs are known to be Ramanujan graphs by [Piz98] and as a consequence have good mixing property with, for all practical purposes, uniform distribution of isogenies.

However, recently proposed attacks (see for instance [EHL+18, EHL+20, PL17], using the endomorphism rings of the starting (or ending) curve have somewhat lowered the confidence that the CGL-function is pre-image and collision resistant. On the other hand, computing the endomorphism ring for a supersingular curve is believed to be a hard problem that is exponential in complexity for a "random" curve, so, under this hardness assumption, only knowing the endomorphism ring from the start is a problem.

In this note we propose a variant (with four sub-variants) of the CGL hash function that uses the dynamics of so called Lattès maps on the projective line $\mathbb{P}$ to "hide" the information of the isogeny walk. This is done by using an analogue to the Merkle–Damgård construction by viewing the Lattès map as a "compression" function. This could possibly avoid the problems with the endomorphism rings and thereby avoiding the pre-image and collision attacks.

However, as indicated in the abstract, it is important to emphasise that there might be attacks on our Lattès map approach (possibly easily spotted by

e-mail: daniel.larsson@usn.no

an expert) that will render our proposal, at least in the present form, useless. The security and complexity of the suggestions made are not discussed in this note, thereby questioning the viability of the overall idea. Nevertheless, we have chosen to present the idea with this shortcoming, to hopefully inspire some more knowledgeable people to take up the thread.

The organisation of the paper is as follows. In section 2 we recall the necessary notions pertinent to elliptic curves. However, we assume that the reader is already familiar with basics. This section is mostly concerned with fixing notation and making a few recollections. The section then proceeds by presenting the construction of Lattès maps. We use a recent result in proven in [BCC+22] to show that the Lattès maps are permutations when the curves are supersingular. As a consequence all points are periodic (i.e., there are no purely pre-periodic points).

Next, for completeness, we present in section 3 a short description of the CGL hash function, and then in section 4 the proposals for the family of hash functions using Lattès maps is presented. Included here is also a (very) brief discussion on some heuristic concerning the proposals and some ideas for future research (besides the security and complexity issues).

2 Elliptic curves and Lattès maps

2.1 Elliptic curves and supersingular isogeny graphs

We will assume the reader is familiar with the basics of elliptic curves over finite fields as presented in [Sil09], but for the reader's convenience we recall some of the necessary notions. Let ∞ denote the unit element on all curves.

Let p be a prime and $\mathscr{E}_{/\mathbb{F}_{p^n}}$ an elliptic curve defined over an extension $\mathbb{F}_{p^n}$ of $\mathbb{F}_p$. We will normally work with short Weierstrass models

$$y^2 = x^3 + ax + b, \quad a, b \in \mathbb{F}_{p^n}.$$

Let K be an arbitrary extension of $\mathbb{F}_p$. Then the *K-rational points* (i.e., the points whith coordinates in K) is denoted $\mathscr{E}(K)$ as usual.

The *N-torsion points* are the points

$$\mathscr{E}[N] := \{\mathtt{P} \in \mathscr{E}(\mathbb{F}_p^{\mathrm{al}}) \mid N \cdot \mathtt{P} = \infty\}.$$

Recall that $\mathscr{E}[N] \simeq \mathbb{Z}/N \times \mathbb{Z}/N$, when $\gcd(p, N) = 1$ and $\mathscr{E}[p^k] \simeq \mathbb{Z}/p^k$ if and only if $\mathscr{E}$ is *ordinary* and $\mathscr{E}[p^k] \simeq \{\infty\}$ if and only if $\mathscr{E}$ is *supersingular.*

An *isogeny* between two elliptic curves $\mathscr{E}$ and $\mathscr{E}'$ is a scheme-theoretic map that is also a morphism of abelian groups. The basic example is $N : \mathscr{E} \to \mathscr{E}$, where we notice that $\ker N = \mathscr{E}[N]$. The degree of a separable[1] isogeny ϕ is $\#\ker\phi$. The group of all isogenies $\mathscr{E} \to \mathscr{E}$ defined over $\mathbb{F}_p^{\mathrm{al}}$ is denoted $\mathrm{End}(\mathscr{E}) = \mathrm{End}_{\mathbb{F}_p^{\mathrm{al}}}(\mathscr{E})$. A curve is supersingular if and only if $\mathrm{End}(\mathscr{E})$ is an order in the quaternion algebra $\mathbf{B}_{p,\infty}$, ramified only at p and ∞.

Let ℓ be a prime different from p. We denote by $\underline{\mathrm{Iso}}_\ell$ the graph where the nodes are the $\mathbb{F}_p^{\mathrm{al}}$-isomorphism classes of supersingular curves over p and where the edges are isogenies of degree ℓ (up to conjugation). The number of nodes is roughly $p/12$. Recall that the j-invariant of a supersingular curve is an element

[1] We won't define this here. All isogenies appearing in this note are separable.

of $\mathbb{F}_{p^2}$. The nodes in $\underline{\mathrm{Iso}}_\ell$ are therefore normally enumerated by the j-invariants. It is well-known that $\underline{\mathrm{Iso}}_\ell$ is a connected Ramanujan graph and there are $\ell+1$ edges emanating from each node, corresponding to cyclic subgroups of $\mathscr{E}_0[\ell]$. The fact that $\underline{\mathrm{Iso}}_\ell$ is Ramanujan has as a consequence that, given two curves, it is presumably hard to find a path of length ℓ^n (for some "large" n) between these curves in $\underline{\mathrm{Iso}}_\ell$. This presumption is the underlying hardness assumption for all cryptographic primitives using isogeny graphs.

2.2 Lattès maps and their dynamics

We begin this section by remarking that the definition of Lattès maps is independent of the ground field. However, we will continue assuming that everything is defined over an extension of $\mathbb{F}_p$ for simplicity. For the basics of Lattès maps (of arbitrary fields) we refer to [Sil07, Chapter 6].

Let $\psi : \mathscr{E} \to \mathscr{E}$ be a rational map (not necessarily an isogeny) and let $\pi : \mathscr{E} \to \mathbb{P}$ be a morphism of varieties. Then a *Lattès map* associated with (ψ, π) is a rational map $\phi : \mathbb{P} \to \mathbb{P}$ making the diagram

$$\begin{tikzcd}
\mathscr{E} \arrow[r, "\psi"] \arrow[d, "\pi"] & \mathscr{E} \arrow[d, "\pi"] \\
\mathbb{P} \arrow[r, "\phi"'] & \mathbb{P}
\end{tikzcd}$$

commutative. The Lattès map is *flexible* if $\psi = N$ for some $N \in \mathbb{Z}$ and the degree of π is 2. The definition given in [Sil07] is slightly more general. The degree of ϕ is N^2 (see [Sil07, Prop. 6.51a]).

Let S be a set, $\alpha : S \to S$ a map and $x \in S$. Then the *orbit* of x under α is the set

$$\int x \cdot \alpha = \{s \in S \mid s = \alpha^n(x), \text{ for some } n \in \mathbb{Z}\}.$$

The set of all orbits is denoted

$$\int_S x \cdot \alpha$$

and is the *dynamical system* associated with (S, α). The reason for the unconventional notation is to convey the idea that computing orbits under a map α, is akin to integrating the elements in the set under the "measure" α.

A point $s \in S$ is called *periodic* (of *period* n) if $\alpha^n(s) = s$ and $\alpha^m(s) \neq s$ for all $m < n$. The point is *pre-periodic* if $\alpha^m(s)$ is periodic for some $m \geq 0$. Notice that every periodic point is pre-periodic.

If ϕ is a Lattès map, Proposition 6.44 in [Sil07] shows that $\mathrm{PrePer}(\phi) = \pi(\mathscr{E}_{\mathrm{tor}})$, where, of course, $\mathrm{PrePer}(\phi)$ is the set of pre-periodic points of $\mathbb{P}$ under ϕ. Clearly, over a finite field, every point of $\mathscr{E}$ is a torsion point so $\mathrm{PrePer}(\phi) = \mathbb{P}$.

A natural question is what the density is of the periodic points is in $\mathbb{P}$ under ϕ. This question is difficult to answer in general, but fortunately for the case of interest to us, this can be answered.

From now on we assume that we have a Lattès map defined by the diagram

$$\begin{tikzcd}
\mathscr{E}_0 \arrow[r, "{[\ell]}"] \arrow[d, "x"] & \mathscr{E}_0 \arrow[d, "x"] \\
\mathbb{P} \arrow[r, "\phi"'] & \mathbb{P},
\end{tikzcd}$$

where ℓ is a prime distinct from p. Notice that, given ℓ, the map ϕ is uniquely determined[2] by $\phi(a) = x(\ell \cdot a)$, $a \in \mathbb{P}$.

The number of $\mathbb{F}_{p^n}$-rational points on $\mathscr{E}$ is $\#\mathscr{E}(\mathbb{F}_{p^n}) = p^n + 1 - t$, where t is the trace of the p^n-power Frobenius. Then Corollary 2.6 of [BCC+22] shows that ϕ is a permutation on $\mathbb{P}(\mathbb{F}_{p^n})$ if and only if $(p^n + 1)^2 - t$ is coprime to p. In particular, $\mathrm{Per}(\phi) = \mathbb{P}(\mathbb{F}_{p^n})$.

A curve $\mathscr{E}_{/\mathbb{F}_{p^n}}$ is supersingular if and only if $t \equiv 0$ modulo p. Hence, in this case $(p^n + 1)^2 - t$ is certainly coprime to p and therefore $\phi : \mathbb{P}(\mathbb{F}_{p^n}) \to \mathbb{P}(\mathbb{F}_{p^n})$ is a permutation.

We say that the Lattès map is *supersingular* if $\mathscr{E}$ is supersingular.

3 The Charles–Goren–Lauter hash function

In [CLG09] a hash function based on walks in supersingular isogeny graphs was introduced. Their idea was the following.

Let $M = (m_0, m_2, \ldots, m_k)$ be a message where each $m_i \in \{0, 1, \ldots, \ell - 1\}$. Choose a "good" (avoiding the j-invariants 0 and 1728) supersingular curve $\mathscr{E}_0$ over $\mathbb{F}_{p^2}$ and a basis $\{\mathtt{P}, \mathtt{Q}\}$ for the ℓ-torsion group $\mathscr{E}_0[\ell]$. We choose $\mathtt{P}$ and $\mathtt{Q}$ with the smallest x-coordinates. As mentioned above there are $\ell + 1$ outgoing cyclic ℓ-isogenies corresponding to cyclic subgroups of $\mathscr{E}_0[\ell]$.

We order the the cyclic subgroups as

$$G_0 := \langle \mathtt{Q} \rangle, \quad G_i := \langle \mathtt{P} + (i-1) \cdot \mathtt{Q} \rangle, \quad 1 \leq i \leq \ell.$$

Now, starting at $\mathscr{E}_0$, we choose the subgroup G_{m_0} and the associated isogeny $\psi : \mathscr{E}_0 \to \mathscr{E}_1 := \mathscr{E}_0 / G_{m_0}$ is computed. In the next step we choose the group G_{m_1}, taking G_{m_1+1} (say) if G_{m_1} corresponds to the dual isogeny $\hat{\psi} : \mathscr{E}_1 \to \mathscr{E}_0$. On $\mathscr{E}_2 := \mathscr{E}_1 / G_{m_1}$, we take G_{m_2}, or G_{m_2+1} if G_{m_2} corresponds to the dual isogeny. We continue like this up to m_k, and the j-invariant of the last curve is the hash of M.

Obviously, the basis elements $\mathtt{P}$, $\mathtt{Q}$ changes in every step. The backtracking issue can be avoided if $p \equiv 1 \pmod{12}$ since then the curves with $j = 0, 1728$ are ordinary and backtracking is only possible when there are non-trivial automorphisms.

It is quite clear that the CGL hash function is pre-image resistant if and only if it is hard to compute an isogeny between two given (supersingular) curves. In [EHL+18, Prop. 7 and Prop. 8] it is proved that the hash function is pre-image and collision resistant if and only if it is hard to compute the endomorphism ring of the starting curve (or the ending curve).

[2] This claim is independent on ℓ being a prime.

It should be noted that there are polynomial collision attacks on the CGL hash function for some special curves corresponding to specific maximal orders in $\mathbf{B}_{p,\infty}$ (see [PL17]).

4 Hashing using supersingular Lattès maps

Choose a "good" supersingular curve[3] $\mathscr{E}_0$ over $\mathbb{F}_{p^2}$ and a basis $\{\mathtt{P}_0, \mathtt{Q}_0\}$ for the ℓ-torsion group $\mathscr{E}_0[\ell]$, where ℓ is a prime (typically 2 or 3). Put $\mathscr{E}[\ell] = \langle \mathtt{P}_0, \mathtt{Q}_0 \rangle$ and choose a cyclic subgroup $G_0 := \langle \mathtt{P}_0 + [N]\mathtt{Q}_0 \rangle$, for some N (this choice will be discussed below).

We consider the Lattès map fitting into the diagram

$$\begin{tikzcd}
\mathscr{E}_0 \arrow[r, "{[\ell]}"] \arrow[d, "x"] & \mathscr{E}_0 \arrow[d, "x"] \\
\mathbb{P} \arrow[r, "\phi_0"'] & \mathbb{P}
\end{tikzcd}$$

We now use G_0 to construct the isogeny $\psi_0 : \mathscr{E}_0 \to \mathscr{E}_0/G_0$ and extend the diagram as (put $\mathscr{E}_1 := \mathscr{E}_0/G_0$)

$$\begin{tikzcd}
\mathscr{E}_0 \arrow[r, "{[\ell]}"] \arrow[d, "x"] & \mathscr{E}_0 \arrow[r, "\psi_0"] \arrow[d, "x"] & \mathscr{E}_1 \arrow[r, "{[\ell]}"] \arrow[d, "x"] & \mathscr{E}_1 \arrow[d, "x"] \\
\mathbb{P} \arrow[r, "\phi_0"'] & \mathbb{P} \arrow[r, equal] & \mathbb{P} \arrow[r, "\phi_1"'] & \mathbb{P}
\end{tikzcd}$$

We now continue like this:

$$\begin{tikzcd}
\mathscr{E}_0 \arrow[r, "{[\ell]}"] \arrow[d, "x"] & \mathscr{E}_0 \arrow[r, "\psi_0"] \arrow[d, "x"] & \mathscr{E}_1 \arrow[r, "{[\ell]}"] \arrow[d, "x"] & \mathscr{E}_1 \arrow[r, "\psi_1"] \arrow[d, "x"] & \cdots \arrow[r, "{[\ell]}"] & \mathscr{E}_{k-1} \arrow[r, "\psi_{k-1}"] \arrow[d, "x"] & \mathscr{E}_k \arrow[d, "x"] \\
\mathbb{P} \arrow[r, "\phi_0"'] & \mathbb{P} \arrow[r, equal] & \mathbb{P} \arrow[r, "\phi_1"'] & \mathbb{P} \arrow[r, equal] & \cdots \arrow[r, "\phi_{k-1}"'] & \mathbb{P} \arrow[r, equal] & \mathbb{P}.
\end{tikzcd}$$

Now, how can the chain in the top row constructed? We use the construction described in the next paragraph. Other constructions, including the method in CGL described above, are obviously possible. The choice made here is only for the sake of illustration.

Let $\mathbf{v}$ be the value to be hashed, expressed in binary as $\mathbf{v} := v_{k-1} \cdots v_1 v_0$, $v_i \in \{0, 1\}$. Let G_0 be the cyclic group $G_0 := \langle \mathtt{P}_0 + [\ell - 1 + v_0]\mathtt{Q}_0 \rangle$. Then we

[3] It should be noted that this is not as easy as it sounds. Since the number of supersingular curves in the moduli space of elliptic curves over finite fields is roughly $p/12$, the density is essentially zero. Combing through all elliptic curves for a supersingular curve when p is a prime of 1024 bits or more is extremely difficult. Furthermore, finding a supersingular curve with an endomorphism ring that is hard to compute obviously adds to the difficulty. This problem is discussed in [BCC+23], [MMP22] and [Wes22]. We refer to these for more details.

have the diagram

$$\begin{tikzcd}
 & & \mathscr{E}_1 \arrow[d, equal] & \mathscr{E}_1 \arrow[d, equal] \\
\mathscr{E}_0 \arrow[r, "{[\ell]}"] \arrow[rr, bend left, "\psi_0"] \arrow[d, "x"] & \mathscr{E}_0 \arrow[d, "x"] & \mathscr{E}_0/G_0 \arrow[r, "{[\ell]}"] \arrow[d, "x"] & \mathscr{E}_0/G_0 \arrow[d, "x"] \\
\mathbb{P} \arrow[r, "\phi_0"'] & \mathbb{P} \arrow[r, equal] & \mathbb{P} \arrow[r, "\phi_1"'] & \mathbb{P}.
\end{tikzcd}$$

Then in $\mathscr{E}_1$ choose $G_1 := \langle \mathtt{P}_1 + [\ell - 1 + v_1]\mathtt{Q}_1 \rangle$ and extend the above diagram as

$$\begin{tikzcd}
 & & & & \mathscr{E}_2 \arrow[d, equal] & \mathscr{E}_2 \arrow[d, equal] \\
\mathscr{E}_0 \arrow[r, "{[\ell]}"] \arrow[rr, bend left, "\psi_0"] \arrow[d, "x"] & \mathscr{E}_0 \arrow[d, "x"] & \mathscr{E}_1 \arrow[r, "{[\ell]}"] \arrow[rr, bend left, "\psi_1"] \arrow[d, "x"] & \mathscr{E}_1 & \mathscr{E}_1/G_1 \arrow[r, "{[\ell]}"] \arrow[d, "x"] & \mathscr{E}_1/G_1 \arrow[d, "x"] \\
\mathbb{P} \arrow[r, "\phi_0"'] & \mathbb{P} \arrow[r, equal] & \mathbb{P} \arrow[r, "\phi_1"'] & \mathbb{P} \arrow[r, equal] & \mathbb{P} \arrow[r, "\phi_2"'] & \mathbb{P}.
\end{tikzcd}$$

We continue like this, choosing in each step the group $G_i := \langle \mathtt{P}_i + [\ell - 1 + v_i]\mathtt{Q}_i \rangle$, to get the desired isogeny chain.

Notice that it is important to choose G_i such that ψ_{i-1} is not equal to the dual $\hat{\psi_i}$ in order to avoid backtracking. As remarked above this can be achieved by choosing p such that $p \equiv 1 \pmod{12}$. Also, in each step it is important to have a canonical way to pick the torsion basis (to ensure the uniqueness of the hash value). This can be done as in the CGL hash function, by taking the ones with the smallest x-coordinates.

Remark 4.1. It is a possibility to change the value of ℓ during the construction of the chain. For instance, alternating between $\ell = 2$ and $\ell = 3$.

4.1 Non-keyed hashing

Put $j_i := j(\mathscr{E}_i)$. We compute the following $\mathbf{z}_0 := \phi_0(j_0)$, and recursively

$$\mathbf{z}_i := \phi_i(\mathbf{z}_{i-1} + j_i), \quad 1 \leq i \leq k.$$

The hash value is then $\mathbf{z}_k$. Notice the vague similarity to the Merkle–Damgård construction.

4.2 Keyed hashing

Let $\boldsymbol{\kappa} := \kappa_{n-1} \cdots \kappa_1 \kappa_0$ be a key with each $\kappa_i \in \{0, 1\}$ and $n \geq k$. Then we can use the groups $G_i := \langle \mathtt{P}_i + [\ell - 1 + v_i \oplus \kappa_i]\mathtt{Q}_i \rangle$ and the proceed as before.

4.3 Dynamic hashing

Let, as in the previous section, $\boldsymbol{\kappa} := \kappa_{n-1} \cdots \kappa_1 \kappa_0$ be a key with each $\kappa_i \in \{0, 1\}$ and $n \geq k$. Split $\boldsymbol{\kappa}$ into blocks, not necessarily of the same size, $\boldsymbol{\kappa} =$

$\tilde{\kappa}_{t-1} \cdots \tilde{\kappa}_1 \tilde{\kappa}_0$, with $t \geq 1$. Let σ_i be the integer representation of the binary number $\tilde{\kappa}_i$.

Now, we define $\mathbf{z}_0 := \phi_0^{\sigma_0}(j_0)$ (i.e., ϕ_0 is iterated σ_0 times), and recursively

$$\mathbf{z}_i := \phi_i^{\sigma_i}(\mathbf{z}_{i-1} + j_i), \quad 0 \leq i \leq k,$$

and when (or if) i exceeds k, we start again from σ_0 and proceed cyclically.

4.4 Dynamic hashing, again

As a final proposal, we present two other dynamical hash functions. The set-up is as in the previous set up but now we only use the first Lattès map, ϕ_0 and propose the recursions defined by $\mathbf{z}_0 := \phi_0(j_0)$,

$$\mathbf{z}_i := \phi_0^{i}(\mathbf{z}_{i-1} + j_i), \quad 1 \leq i \leq k,$$

and

$$\mathbf{z}_i := \phi_0^{\sigma_i}(\mathbf{z}_{i-1} + j_i), \quad 0 \leq i \leq k.$$

Notice the difference between these two: the first is non-keyed while the second is keyed.

Due to the number of iterations needed, these proposals seems unlikely to be of any practical use in the present form.

4.5 Some simple heuristics and computer experiments

First notice that the construction of the isogeny chain follows the same heuristics as the CGL-function [CLG09].

Next, we observe that the computation of ϕ_i is already done in the computation of $[\ell]$ and the chosen subgroups, so no extra work constructing these are required. Therefore, the cost of the non-keyed hash function is essentially the same as the cost of the CGL-hash function. The same applies to the keyed-hash function.

The possibly expensive version are thus the dynamic hash functions due to the computation of the orbits in each step. Clearly this is dependent on the block sizes in the key $\boldsymbol{\kappa}$. We did some initial computer experiments with Sage [The21] with $p = 2^{372}3^{239} - 1$, $\ell = 2$, and curve, given in short Weierstrass form as

$$\begin{aligned} y^2 = x^3 &+ 343984983749872389674928342342435345345342422352x \\ &+ 68743598734509720983928402983428798398798728798273984795. \end{aligned}$$

We note that this curve is not a supersingular curve, but this is inconsequential for the illustration. Also, the 2-torsion is only defined over $\mathbb{F}_{p^2}$.

If the block size (i.e., the number of iterations in the orbits) is 10000 and the (randomly chosen) starting value is

$$\begin{aligned} x = 8758987875985456789854567890234567876543456567890987654\backslash \\ 321234567893298765432345634567898790234568765434565678\backslash \\ 9098765432123456985456789023456787654345656789098765432\backslash \\ 1234567893298765432345634567898778932987654323456345678\backslash \\ 98765432123456789876587654789, \end{aligned}$$

the Lattès map is iterated in roughly 0.2s. Reducing the block size to 100 the computation takes about 5μs. On the other hand, taking 100000 iterations yields a computing time of around 1.7s. In fact, the size of the curve coefficients and starting value doesn't seem to have much impact on the speed. We did the computations on a MacBook Pro with an M1 processor.

Obviously this has to be scaled up o allow for computation of all steps in the chain. On the other hand, the implementation is obviously far from optimised and should ideally be implemented in C/C++ or assembler. In addition, the choice of block size is certainly not obvious. Furthermore, using a Montgomery curve and the associated addition formulas should reduce the computation time even further.

4.6 Future work

As mentioned repeatedly, a more thorough study of the security and complexity of our proposals is necessary for determining the viability of the presented families of hash functions.

The initial idea in the use of Lattès maps to hashing was to use the Deuring lifting theorem [Lan87, Thm. 13.12] to construct Lattès maps over number fields. However this presented a lot of problems, primarily related to uniqueness. To give a short version of that idea, consider an isogeny walk as in the CGL hash function. Using the Deuring lifting theorem in each step we can lift the isogeny to an isogeny over a number field between two CM-curves. Now, in a sense, Lattès maps over number fields has a richer theory, in particular the Julia set can be non-empty in which case there is "chaotic" behaviour in the dynamics (see [Sil07]). Then one could use these Lattès maps over the number fields to study the orbits of the j-invariants of the curves in the CGL-walk, thereby possibly constructing a more "chaotic" hash function.

A more thorough study along those lines is perhaps a worthy effort in the future.

Acknowledgements

I would like to thank Joe Silverman for directing me to using Lattès maps over finite fields directly, allowing me to avoid certain problematic issues related to lifting curves from characteristic p to characteristic zero. As a by-product I found the very useful paper [BCC+22]. As mentioned above, my first idea was to use Deuring's lifting theorem and Lattès maps in characteristic zero to dynamically iterate the j-invariants as in the proposals above.

References

[BCC+22] Zoë Bell, Jasmine Camero, Karina Cho, Trevor Hyde, Chieh-Mi Lu, Bianca Thompson, and Eric Zhu. Density of periodic points for Lattès maps over finite field. *J. Number Theory*, 238:951–966, 2022.

[BCC+23] Andrea Basso, Giulio Codogni, Deirdre Connolly, Luca De Feo, Tako Boris Fouotsa, Guido Maria Lido, Travis Morrison, Lorenz Panny, Sikhar Patranabis, and Benjamin Wesolowski. Supersingular curves you can trust. In *Advances in cryptology—EUROCRYPT*

2023. Part II, volume 14005 of *Lecture Notes in Comput. Sci.*, pages 405–437. Springer, Cham, [2023] ©2023.

[CLG09] Denis X. Charles, Kristin E. Lauter, and Eyal Z. Goren. Cryptographic hash functions from expander graphs. *J. Cryptology*, 22(1):93–113, 2009.

[EHL$^+$18] Kirsten Eisenträger, Sean Hallgren, Kristin Lauter, Travis Morrison, and Christophe Petit. Supersingular isogeny graphs and endomorphism rings: reductions and solutions. In *Advances in cryptology—EUROCRYPT 2018. Part III*, volume 10822 of *Lecture Notes in Comput. Sci.*, pages 329–368. Springer, Cham, 2018.

[EHL$^+$20] Kirsten Eisenträger, Sean Hallgren, Chris Leonardi, Travis Morrison, and Jennifer Park. Computing endomorphism rings of supersingular elliptic curves and connections to path-finding in isogeny graphs. In *ANTS XIV—Proceedings of the Fourteenth Algorithmic Number Theory Symposium*, volume 4 of *Open Book Ser.*, pages 215–232. Math. Sci. Publ., Berkeley, CA, 2020.

[Lan87] Serge Lang. *Elliptic functions*, volume 112 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, second edition, 1987. With an appendix by J. Tate.

[MMP22] Marzio Mula, Nadir Murru, and Federico Pintore. On random sampling of supersingular elliptic curves. Cryptology ePrint Archive, Paper 2022/528, 2022. https://eprint.iacr.org/2022/528.

[Piz98] Arnold K. Pizer. Ramanujan graphs. In *Computational perspectives on number theory (Chicago, IL, 1995)*, volume 7 of *AMS/IP Stud. Adv. Math.*, pages 159–178. Amer. Math. Soc., Providence, RI, 1998.

[PL17] Christophe Petit and Kristin Lauter. Hard and easy problems for supersingular isogeny graphs. Cryptology ePrint Archive, Paper 2017/962, 2017. https://eprint.iacr.org/2017/962.

[Sil07] Joseph H. Silverman. *The arithmetic of dynamical systems*, volume 241 of *Graduate Texts in Mathematics*. Springer, New York, 2007.

[Sil09] Joseph H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer, Dordrecht, second edition, 2009.

[The21] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 9.4)*, 2021. https://www.sagemath.org.

[Wes22] B. Wesolowski. The supersingular isogeny path and endomorphism ring problems are equivalent. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1100–1111, Los Alamitos, CA, USA, feb 2022. IEEE Computer Society.