Efficient algorithms for abelian varieties and their moduli spaces

Damien Robert

Efficient algorithms for abelian varieties and their moduli spaces

Habilitation à diriger les recherches

Damien Robert

March 2021

[Current Version: Wednesday September 22, 2021 at 10:16]

To my wife Blandine

To my children: Ange†, Olympe, Alphonse

†04 September 2016

CONTENTS

THANKS

I would like to thank the reviewers: Bas Edixhoven, Pierrick Gaudry and John Voight who did an amazing job in reviewing this hdr. I apologize for the font size, apparently the text can be a bit hard to read! I would also like to thank the rest of the jury: Andreas Enge, David Kohel and Christophe Ritzenthaler for accepting to participate.

On the professional side, I think Andreas Enge and all the LFANT team deserve special thanks for the welcoming atmosphere created in Bordeaux. I enjoy our coffee/tea break very much. I would like to thank my coauthors, and especially David Lubicz whose influence on my scientific career has been considerable. I would also like to thank my students who taught me a lot (hopefully as much as I was able to teach them!) A particular thought for Guillaume Hanrot, supervising students made me appreciate even more his supervision for my PhD thesis. Last but not least, I would like to thank my colleagues, many of whom are friends, for the nice atmosphere in the conferences or our scientific discussions. There are too many to list them all, lest I forget some! This includes my colleagues from the agregation juries, this was a wonderful experience, even though I don't have time anymore to participate for now.

On the personal side, I would also like to thank my esteemed colleague and friend Gaetan Bisson for his invitations and for initiating me to the wonders of scuba diving!

Finally my wife Blandine deserves special thanks. I promised her this hdr would not take long to write, but I was very very wrong. I planned to finish way in advance the birth of our son Alphonse, alas I underestimated the task ahead and did not manage it. This is my excuse for the many typographical errors still remaining, after the birth of Alphonse my corrections were not as thorough as I would have liked. For her patience she deserves a lot of praise.

Finally this hdr is dedicated to my children, whom I love very much.

1 INTRODUCTION

CONTENTS

1.1 SECRET SHARING AND CRYPTOGRAPHY

When giving expository introductions to cryptography, I found that one of the best way to illustrate public key cryptography is to give the example of secret sharing between Alice and Bob across a public channel. Unsurprisingly, this was the first example of public key cryptography by Diffie and Hellman [DH76]. In my expositions, I use examples coming from paths in "commutative graphs" rather than group exponentiation, this is more visual.

Formalising this a bit, Alice has a secret s_A and publishes some information p_A, Bob has a secret s_B and publishes p_B, and there exists a function DH such that $DH(s_A, p_B) = DH(s_B, p_A)$ (and DH can be computed quickly) but Eve cannot recover this common secret (in reasonable time) knowing only p_A, p_B. Of course this is a bit too abstract, so to give concrete instances we can consider a category $\mathfrak{C}$ with pushouts. Alice's secret is an arrow: $a : O \to A$, Bob's secret is an arrow $b : O \to B$, and the common secret is the pushout C:

$$\begin{array}{ccc} O & \xrightarrow{a} & A \\ \downarrow{\scriptstyle b} & & \downarrow \\ B & \longrightarrow & C. \end{array}$$

The question is whether Alice and Bob can publish enough informations so that the other can compute the pushout without Eve being able to.

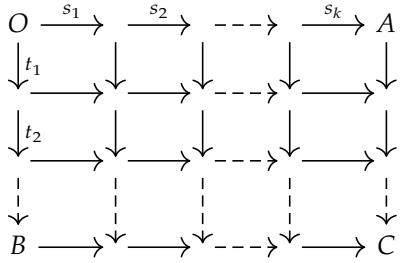
Example 1.1.1.

- If G is a commutative group acting on a set X[1], we can construct the action groupoid where maps are of the form $x \to g \cdot x$. Alice's secret is g_1 and she publishes only the codomain $g_1 \cdot x$. Likewise Bob's secret is g_2, and the shared secret is $g_2 g_1 \cdot x$, which Bob can compute from his secret g_2 and Alice's public $g_1 \cdot x$. (The pushout of $x \to g_1 x$ and $x \to g_2 x$ is indeed $x \to g_1 g_2 x$ if $\mathrm{Stab}(x)$ is trivial.)

  Typically in this sort of situation, the action $g \cdot x$ is computed for a "large" g, via a decomposition $g = \prod s_i$ where s_i are "small" elements whose action can be computed quickly.

  From the categorical point of view, this amount to constructing the pushout via the diagram:

[1] Since a group action is an algebra for the action monad $X \to G \times X$, where the monad $G \times G \times X \to G \times X$ is given by the group law, we could try more generally to work with a monoidal monad on a (symmetric) monoidal category, and use the tensor product on its Eilenberg–Moore algebras. This tensor product exists under wide conditions [Sea13]. But I do not know of any cryptographic application yet!



If G acts freely on X, the resulting groupoid is a graph, and the key exchange simply amount to following paths in the graph labelled by the "small" arrows s_i.

Since Eve can construct $g_3 g_1 \cdot x$ from her own g_3, she can easily construct instances of the hidden congruence problem where the hidden congruence subgroup is given by $g_1 \operatorname{Stab}(x)$.

If the action is free, this becomes an instance of the hidden shift problem. If $X = G/H$, this becomes an instance of the hidden subgroup problem (for $\operatorname{Stab}(1_G)$). We refer to [VHI06] for more details on these problems.

We recall that the hidden subgroup problem can be solved in polynomial time with a quantum computer for a commutative finite group, and the hidden shift problem also if the group G is cyclic (finite). This is an instance of Shor's period finding algorithm [Sho94]. However the hidden shift problem for a commutative (non cyclic) abelian group is only subexponential, by reducing to the hidden subgroup problem on the Dihedral group [Kup05].

- Diffie-Helmman's original secret sharing involves the discrete logarithm problem on a cyclic group G (typically of prime order p). This can be solved via the hidden subgroup problem on $(\mathbb{Z}/\#G\mathbb{Z})^2$, so [Sho94] gives a polynomial time quantum algorithm which breaks the discrete logarithm problem.

  We may also reformulate Diffie-Hellman's as an instance of a group action where $(\mathbb{Z}/p\mathbb{Z})^*$ acts via $n \cdot g = g^n$ [De 17, § 14.1].

- Another example is given by the elliptic curves with complex multiplication by O_K, a maximal order in an imaginary quadratic field K. By the theorems of complex multiplication, this set is a torsor (ie a principal homogeneous space) under $\mathrm{Cl}(O_K)$. This example was introduced by Couveignes in [Cou06] and developed by Rostovtsev and Stolbunov in [RS06; Sto10].

  There has been a lot of renewed interest in this protocol following practical improvements in [DKS18; CLM+18].

- In the category of abelian varieties and isogenies, the pushout of $f : A \to A/K_A$ and $g : A \to A/K_B$ is given by $h : A \to A/(K_A + K_B)$. Indeed, if $u : A \to C$ factors through A/K_A and A/K_B, then $\operatorname{Ker} u$ contains both K_A and K_B. If we restrict to principally polarised abelian varieties and ℓ-isogenies, we may localize the endomorphisms $[\ell]$ to get a groupoid.

  Modern cryptographic proposals use isogeny graphs (ie the skeleton of the isogeny category above) of supersingular elliptic curves. It is well known that this graph is an expander graph, so uniform mixing occurs rapidly after a small amount of random walking.

  This illustrates a case where Bob needs more information than just the codomain of $A \to A/K_A$ to complete the push-out square. Namely, Alice also publishes the image by the isogeny of some points. It seems (for now) that this extra information is not enough for Eve to break the key exchange.

  After improvements to the cryptosystem (called SIDH: supersingular isogenies Diffie-Hellman) [JD11; DJP14; CLN16], SIKE (supersingular isogenies key exchange) is now a Round 3 alternate candidate for the NIST PQC.

  Isogeny based cryptography is now a very active topic of research, in particular to adopt protocols from the DLP to this new setting. For instance a lot of progress has been made for an efficient signature scheme [DKL+20]. Other applications of isogeny based schemes are hash functions [CLG09], and more recently Verifiable Delay Functions [BBB+18; DMP+19].

1.2 PURPOSE OF THIS DOCUMENT AND A BRIEF DESCRIPTION OF MY RESEARCH

1.2.1 *Overview*

This document is the occasion to give a summary of my research on the algorithmic aspects of abelian varieties and their moduli space [LR10; FLR11; LR12; LR13; CR15; LR15a; LR15b; LR16; MR21; DJR+17; KPR20; KNR+20b].This is also the occasion to describe results not published or not yet in public preprint form [BCR11; SR11; BLR11; IMR+14; MR19; LR20b; MR20a; LR20a], along with some brand new results.

The intended audience is cryptographers and algorithmicians. All algorithms developed here are related to cryptographic aspects of abelian varieties (both classical and post quantum): efficient arithmetic, pairings, isogenies and modular correspondances, point counting, liftings, and the Complex Multiplication method. While the cryptographic applications are essential, I will focus in this document only in the algorithmic aspects. But this explain why abelian varieties over finite fields will play a major role. A remark on the "efficient" part of the title: ideally we would like algorithms that are quasi-linear in the size of their inputs and outputs (I will also call them quasi-optimal). If this is not achievable, we want at least a polynomial time algorithm with the smallest exponent possible. Remarkably, with the tools we can use on abelian varieties, quite a lot of the algorithms presented here will be quasi-optimal.

Likewise, I only give very brief introductions and theoretical background to the different topics in each Chapters. One exception is Chapter 2 were I develop a bit Mumford's theory of algebraic theta functions. There are several reasons for this: first the algorithmic aspect of the arithmetic of theta functions was developed as we needed it in several articles [LR12; LR10; CR15; LR15a; LR16], so this was a nice occasion to summarize them. Secondly, for reasons explained in Section 2.1, I wanted to extract the exact properties which we used in the theta model in order to adapt them to other models. Finally this allows to give a general recipe to construct theta functions (Recipe 2.5.3), in order to get Thomae like formula (see Section 2.9).

My goal in this document is to give a general overview and philosophy of the algorithms, skipping over details if possible (somewhat sacrificing a bit of rigour, but the technical details can be found in the articles). Still, since there are several new results along with a description of some results which are not yet in preprint form, I had to be quite a bit longer than I originally intended. I apologize for the length.

Every paper I wrote is algorithmic, and has been implemented (either by me or my coauthors, in particular my wonderful PhD students). Not all the implementations are mature enough to be published, some are just proof of concepts, but this is a general philosophy: always have at least a fully implemented example by paper. This is the difference between the "brand new results" mentioned above (I did not have time to implement them yet[2]), and the results not published or not yet in public preprint form (the articles may not quite yet be done[3], but we have working implementations). The main public software I maintain is [BCR10] which contains tools for computing isogenies and pairings with theta functions, and conversion back and from hyperelliptic curves (with admittedly a less than perfect API). Some not yet public branches contain code for computing cyclic isogenies. There is also the more recent library [KNR+20a] (well it was published recently, but the development started in 2011). This adds support for the description of the isogeny class of a power of an elliptic curve, tools for genus 3 quartic curves to theta conversion, and the evaluation of algebraic modular forms (derived from theta constants) to evaluate Serre's obstruction when $g = 3$ and Schottky's obstruction when $g = 4$.

A specificity of my research is that I try to develop algorithms that work for any abelian varieties, not only Jacobians. This is less motivated by cryptography: because of faster index calculus attack [Gau09; EGT09; DT08] when $g > 2$, classical DLP based cryptosystems are more efficient for elliptic curves and Jacobians of hyperelliptic curves of genus 2 than in higher dimension; while SIDH based cryptography are more efficient when restricting to the supersingular elliptic curves rather than to supersingular/superspecial abelian varieties (see [CS20] which reduces the problem to lower dimensional varieties).

In practice, this means that I use models given by algebraic theta functions, which form a universal model over any (algebraically closed) field (with an exception in the case of characteristic 2, but see [GL09]). Technically, by [Mum66; Mum67a; Mum67b], we have equations for the universal abelian scheme $\mathcal{X}_{g,n} \to \mathcal{A}_{g,n}$ with a totally symmetric (normalised) ample line bundle $\mathcal{L}$ and a symmetric theta structure of level $n \geq 4$, and furthermore $\mathcal{A}_{g,n}$ is a quasi-projective scheme smooth over $\mathbb{Z}[1/n]$ with coordinates (ie modular invariants) on $\mathcal{A}_{g,n}$ given by the theta constants.

Even when working with abelian varieties of small dimension ($g \leq 3$), where every principally polarised (with indecomposable polarisation) $(A, \mathcal{L})$ is a Jacobian, having a universal model is convenient because this means we

[2] A notable exception is the improvement of Satoh's algorithm described in Section 6.6 which is already implemented for elliptic curves in the development branch of Pari [21].

[3] I can provide a preliminary version if requested.

don't need to treat specially the case where A splits into smaller dimensional Jacobians. This allows us to not treat specially isogenies $A \to B$ where A is a Jacobian but B is a product of Jacobians. In genus 3, when working with Jacobians, the situation is even more annoying because different invariants are used when dealing with Jacobians of hyperelliptic curves (Shioda invariants), compared to plane quartics (Dixmier-Ohno). This complicates the definition of class polynomials (if the CM locus has both hyperelliptic and quartic curves), and also their reduction (eg if a quartic reduces to an hyperelliptic curve [LLG+20]).

Still there are drawback to theta functions (most notably we need to take a field extension so that the level structure becomes rational, this is especially annoying for algorithmic applications over number fields). So a very recent shift in my work was to extract the exact underlying algorithmic hypotheses of a model under which we can build the arithmetic (ie pairings, isogenies, …) of abelian varieties given by this model. Two keys examples of such models are given by the theta model of course, and Jacobians of curves. I refer to Chapter 2 for more details.

By contrast, I am less interested in, for example, gaining one multiplication in the arithmetic of a specific model of an elliptic curve (even though it is obviously important). As part of the SIMPATIC ANR project, I have worked on improving pairing computations on a phone's SIM card. And although I have worked on improving the arithmetic of abelian varieties (see for instance [LR16]), in that paper we do give a description of the arithmetic of any Kummer variety (in any dimension), before applying it to the theta model.

Likewise, rather than developing a custom isogeny algorithm for every model of elliptic curves, I find more interesting to develop an isogeny algorithm from "first principles". Namely let $f : A \to B$ be the isogeny we want to compute, s_i our current coordinates on A, t_i the coordinates we want to use on B. Representing the isogeny then becomes a question of interpreting the $t_i \circ f$ as rational fractions in the s_i. We can decompose this in three steps:

- If the t_i are sections of a line bundle $\mathcal{M}$ on B and s_i of $\mathcal{L}$ on A, we first try to construct $f^*\mathcal{M}$ from $\mathcal{L}$;
- Then we try to find the sections of $f^*\mathcal{M}$ that descend to B;
- Then we identify the t_i among these sections.

This approach is developed in Chapter 4. For instance, on every model of elliptic curve on which we have explicit formula to compute the Miller functions (eg to compute pairings), we can use this framework to derive an isogeny formula. (Still the case of elliptic curves is easier, because hidden in the approach outlined above is the action of the theta group, but for elliptic curves this action can remain implicit, see Example 4.2.1 as to why.)

1.2.2 Algorithms for abelian varieties

Now let us go back to the subject of this hdr. For *efficient* algorithms of abelian varieties A, we would like:

- The arithmetic of course;
- Pairings;
- Isogenies computations (from the kernel);
- Being able to change models;
- Sampling points or finding points in the variety;
- Computing associated data: if A is defined over a finite field its number of points and endomorphism ring; if A is defined over $\mathbb{C}$ a period matrix…

Not all of these problems are solved in this document: endomorphisms rings and (fast) period matrix computations are for now just research projects (see Sections 5.7 and 6.7), and I don't treat finding points at all. This last topic is a hard problem, especially over a number field, and numerous tools have been developed by number theoricians (eg Heegner points, Chabauty-Coleman and its extensions). Even for elliptic curves over a finite field this is an interesting problem: while finding points is easy, sampling points uniformly is not too difficult, hashing deterministically (and uniformly) into a curve is much harder, see eg [CK12]. Likewise, sampling (somewhat uniformly) points in a Jacobian over a finite field is not too hard (simply sum $g+1$ points on the curve if it is of genus g and the base field is not too small), but finding points in the theta model is harder (except in very small dimension).

We treat arithmetic in Chapter 2 along with an outline on how to transform any model where we have an explicit version of the theorem of the square and of the action of the theta group into a theta model. This applies in particular to Jacobians. The converse (from theta to Jacobians) is not treated in this document; it is well known for Jacobians of

hyperelliptic curve, but the general case is probably much harder, given its relationship with the Schottky problem. Pairings are handled in Chapter 3 and isogenies in Chapter 4. We give an overview of point counting via the SEA method in Chapter 5 and via p-adic lifting in Chapter 6 (missing point counting algorithms in this document are Kedlaya's algorithm and deformation based methods).

There is a gap between algorithms for elliptic curves and higher dimensional abelian varieties. In the case of elliptic curves, we have efficient algorithms for most of the topics above: isogenies [Vél71; Koh96; Elk92; BMS+08], point counting [Sch85; Sch95; Elk92; Mor95; Elk97; Satoo; Kedo1] endomorphism rings [Koh96; FM02; BS09; Bis11], modular polynomials [Eng09a; BLS12], class polynomials [Sut11; ES10]…Often, adapting one algorithm from dimension 1 to dimension 2 is enough that the adaptation to higher dimensions is relatively straightforward (especially if the adaptation was done via a theta model). Still, there are of course specificities to the dimension 2 case:

- All indecomposable abelian surfaces are Jacobians of hyperelliptic curves of genus 2;
- In the case of real multiplication, for abelian surfaces the real orders are quadratic, hence Gorenstein (ie their trace dual is inversible). This simplify quite a bit the study of their modules;
- We have fast (quasi-linear) evaluation of theta functions (hence modular invariants) and period matrices when $g \leq 2$;
- Sometimes the objects are simply too big to be computed in higher dimensions. For instance the Siegel ℓ-modular polynomials with $g = 1$ are of size $\widetilde{O}(\ell^3)$, and of size $\widetilde{O}(\ell^{15})$ when $g = 2$. When $g = 3$, their size is $\widetilde{O}(\ell^{48})$. As an example, the modular polynomials with $g = 2$ and $\ell = 7$ using the smaller theta invariants already take 29GB. It is safe to say that nobody will ever compute it[4] for $g = 3$ and $\ell = 7$.

Most importantly, there is no hope to extend some algorithms from elliptic curves (with the same complexity) without assuming (explicit) real multiplication. Indeed, in higher dimensions, we can only have cyclic isogenies between principally polarised abelian varieties if the Néron-Severi group $NS(A)$ is different from $\mathbb{Z}$. For elliptic curves, then endomorphism ring always contains $\mathbb{Z}$, which can be thought of as the real multiplication order. So for abelian surfaces over a finite field $\mathbb{F}_q$, while we can obtain an $\widetilde{O}(q^4)$ (heuristic) SEA like algorithm (see Section 5.5) like in the elliptic curves case, the algorithm is not uniform; the constants in the $\widetilde{O}$ depend on the real multiplication order. From this point of view, one should consider the moduli of abelian surfaces over $\mathbb{F}_q$ with the strata given by the real quadratic orders.

1.2.3 *Algorithms for moduli spaces*

Likewise, some algorithms we would like for moduli spaces of abelian varieties are:

- Explicit equations for moduli or even just birational models. Eg construct models of integral Shimura varieties of PEL type.
- Description of the tangent spaces.
- Fast evaluation of modular functions (ie of coordinates on the moduli).
- Explicit modular/Hecke correspondances.
- Explicit maps between the moduli: forgetting structure/level, or conversely lifting level, ie computing preimages.
- Sampling points.
- Lifting and reduction[5].

[4] What can be done however is to compute evaluated modular polynomials, see Section 5.3.8. Also Hilbert modular polynomials are much smaller: their size is $\widetilde{O}(\ell^4)$ when $g = 2$, and $\widetilde{O}(\ell^5)$ when $g = 3$. As an example, the modular polynomial for $\mathbb{Q}(\sqrt{2})$ with $\ell = 41$ is only of size 7.2MB.

[5] Of course this also belon to "algorithms on abelian varieties", the distinction between the two aspects is somewhat arbitrary.

Some of these topics reach the limit of our current theoretical knowledge on automorphic forms. In fact even when $g = 1$, not quite all modular curves of genus ≤ 1 have been computed yet (see [SZ17; BS19] for recent progress). As for moduli of abelian surfaces, I don't know how to compute efficiently (apart from evaluation and linear algebra or linear algebra on the Fourier coefficients) interesting moduli lying in the Siegel moduli like Humbert surfaces, Shimura curves and generalised Humbert varieties. Broadly: once we have a Shimura variety of PEL type (eg the Siegel or Hilbert varieties), it is easier to change the level structure and polarisation type (eg to build the modular polynomials) than the endomorphism type. Indeed in one case this corresponds to changing the compact open subgroup of $G(\mathbb{A}_f)$ while in the other case this corresponds to changing the reductive group G itself.

In this document, we will mainly focus on modular correspondances and modular polynomials (on Siegel and Hilbert moduli spaces), and their applications to isogeny computations and point counting. We will also describe tangent spaces (ie the explicit Kodaira-Spencer isomorphism) when $g = 2$. We will then use modular polynomials to compute canonical lifts and class polynomials. The CM locus, being a Shimura variety of dimension 0, is easier to compute efficiently than the higher dimensional Shimura varieties. And thanks to the Taniyama-Shimura formula the reduction of CM points is well known.

We will also use the forgetting map from Hilbert to Siegel, and the Torelli map. We have fast evaluation of theta functions and period matrix only up to dimension $g \leq 2$ [Dup06; Lab16], so we will give alternative methods to compute modular polynomials for higher dimensional abelian varieties (ie p-adic lifting and CRT).

Like for abelian varieties, sampling points in moduli can be a hard problem. Of course when the moduli is unirational this is easier (we recall that $\mathcal{A}_g$ is unirational when $g \leq 5$, and is in fact rational when $g \leq 3$). On the other hand, sampling uniformly and efficiently supersingular elliptic curves (*without* starting with a well known one and taking random isogenies) is a hard problem that would be very useful for supersingular based cryptosystems.

1.2.4 The yin and yang of arithmetic, pairings, and isogenies

As we have seen, isogenies will play a key role for algorithms not only for abelian varieties but also for moduli spaces. There is a strong relationship between the three aspects of arithmetic, pairings, and isogenies. The multiplication by $[n]$ map is a special isogeny. In [Mum66], Mumford even recovers the arithmetic on theta functions using the isogeny $(x, y) \mapsto (x + y, x - y)$. We can also use isogenies to speed up duplication and triplication formulae [Gau07; DIK06]. Pairings can be seen as a refinement of the arithmetic: for instance on an elliptic curve, if we know not only to compute $P + Q$ but also the function $\mu_{P,Q}$ whose divisor is $(P) + (Q) - (P + Q) - (0_E)$ then we know how to compute the Weil and Tate pairing. The same holds for an abelian variety provided we have an explicit version of the theorem of the square, see Chapter 3. Also, to compute isogenies we need their kernel to be isotropic, see Chapter 4. But conversely, given an isogeny f, we have the Weil-Cartier pairing on the kernel of f and the kernel of its dual $\hat{f}$.

In fact, all three aspects are somewhat unified in the notion of the theta group $G(\mathcal{L})$ of a line bundle $\mathcal{L}$: $G(\mathcal{L})$ encodes descent, hence isogenies, the Weil pairing $e_{\mathcal{L}}$ is recovered as the commutator pairing on $G(\mathcal{L})$, and the action of $G(\mathcal{L})$ on $\Gamma(\mathcal{L})$ is irreducible, hence gives information on the arithmetic, see Chapter 2.

We have seen that isogenies allows to build cryptosystems [Cou06; Tes06; RS06; Sto10; CLG09; JD11; DJP14; CLN16; DKS18; CLM+18; DMP+19] (and many more), but they have also been used in the classical DLP to extend attacks [GHS02; Smi08] or prove random self-reducibility [JW15], to reduce the impact of side channel attacks [Sma03]. Further applications include (via elliptic periods) constructing irreducible polynomials or a normal basis over a finite field [CL13; CL09], computing isomorphisms or embeddings between finite fields [Nar18; BDD+19], finding smoothness basis invariant by automorphisms [CL08b], and probably many more!

1.2.5 An ode to algebraic geometry

I thought this hdr was a nice occasion to write a bit about some of the theory of abelian varieties (ie fun topics I learned a bit about because they have applications to algorithms on abelian varieties), but this was way too ambitious a project. However since this will allow me to skip over developing too much theory on this (already too long) document, a draft (currently of around 120 pages) is available as [Rob21]. See [Rob21, Chapter 1][6] for more motivations.

For those who wonder why on earth the theory of algebraic stacks and abelian schemes can be useful for algorithms on an abelian variety over a finite field, let me give an example. First an abelian scheme A over S is simply a family of abelian varieties of the same dimension for all geometric points of S, at least if S is reduced and

[6] The hyperlinks should work if both documents are in the same folder

connected (and the neutral points should vary continuously along the fibers, ie come from a section $\epsilon : S \to A$.) Secondly using algebraic stacks allows to construct *fine* moduli spaces rather than *coarse* moduli spaces (since they keep track of the automorphisms). In particular the moduli stack $\mathcal{A}_g$ of principally polarised abelian varieties is smooth over $\mathbb{Z}$, and we have a universal abelian stack $X_g \to \mathcal{A}_g$ over it. (The stack $\mathcal{A}_g$ is a Deligne-Mumford stack, so has an étale cover by a scheme, so all étale-local properties of abelian schemes make sense over $\mathcal{A}_g$.)

Smoothness of $\mathcal{A}_g$ allows to lift abelian varieties to characteristic zero (we can also lift étale isogenies, and of course for ordinary abelian varieties we can take the canonical lift which lift all endomorphisms, see Chapter 6). But smoothness is not the only trick we can use: to prove results on the universal scheme, we can also use rigidity or simple flatness arguments, see [Rob21, Section 2.3.6].

For instance, to give an algebraic meaning of a complex modular form $\mathfrak{g}$ on an ordinary abelian variety $A/\mathbb{F}_q$, one can take its canonical lift to $\mathbb{Q}_q$, and embed it into $\mathbb{C}$. Since the lift $\widetilde{A}$ is CM, by CM theory its period matrix τ is defined over a number field and so is $\mathfrak{g}(\tau)$ (over a larger number field) if $\mathfrak{g}$ is a suitable integral modular form (which can be checked from its Fourier coefficients). Then we need to carefully check if $\mathfrak{g}(\tau)$ reduces well. This was the original proof in [KNR+20b] to show that our computation of Serre's algebraic obstruction made sense over $\mathbb{F}_q$.

An alternative and simpler method, which also works for non ordinary abelian varieties, is to use the definition of a weight scalar k modular form as a section of the k-th tensor power of the Hodge line bundle $\mathfrak{h}^{\otimes k}$, defined over a suitable compactification[7] of $\mathcal{A}_g$, ie as a functorial application $\mathfrak{g} : (A, w_A) \mapsto \mathfrak{g}(A, w_A)$ satisfying $\mathfrak{g}(A, \gamma \cdot w_A) = \det^{-k} \gamma \cdot \mathfrak{g}(A, w_A)$. Once we have defined algebraically a candidate $\mathfrak{g}_0$, we can check directly over $\mathbb{C}$ that it coincides with $\mathfrak{g}$. Indeed, since $\mathcal{A}_g$ is smooth over $\mathbb{Z}$, $\mathfrak{h}^{\otimes k}$ is flat over $\mathbb{Z}$, so we may check equality of sections over their pullback to $\mathbb{C}$. Alternatively, we may invoke the q-expansion principle.

1.3 A CHRONOLOGY

This Section is mainly for myself, I recommend the reader to skip directly to Section 1.5.

Research

It is an interesting exercice (for me!) to give a (very partial and probably somewhat biased) chronology of my research. Indeed publication date does not always corresponds to the date an idea was first worked out, and it is interesting to look back in retrospect at how we explored things.

I started my PhD in 2007 under the direction of Guillaume Hanrot, and defended it in 2010. The goal was to compute isogenies between hyperelliptic curves of genus 2. At the time, I naively tried to adapt Vélu's formula by computing traces under the kernel of the projective coordinates given by Cassel and Flynn in [CF+96] (these are given by sections of 4Θ where Θ is the theta divisor). But experiments showed that we did not get correct coordinates on the isogenous Jacobian $B = \mathrm{Jac}(C')$. Now I know that while the traces of course descend to B, the problem is that they are sections of (a divisor algebraically equivalent to) $4\ell\Theta$ rather than 4Θ. See Example 4.2.1 for more details[8].

The solution was proposed by David Lubicz, to whom I owe a lot, whose wonderful insight was that Mumford's algebraic theory of theta functions was perfectly adapted to construct isogenies (if only because of Mumford's isogeny formula). He had constructed a modular correspondance $\mathcal{A}_{g,\ell n} \to \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$ (where $\mathcal{A}_{g,n}$ is the moduli of principally polarised abelian varieties with a symmetric theta structure of level n), where each projection correspond to an isogeny, as a more convenient way to compute canonical lifts for point counting than the multiplication formula used in [CL08a]. We first studied fibers of this theta modular correspondance in [FLR11]. A more detailed analysis on how to compute these fibers in practice (without a Grobner basis), ie how to raise the level, led to our first isogeny algorithm in [LR12]. At the time we first raised the level via an isogeny, before applying Mumford's isogeny formula to descend back in level, ie we could only compute ℓ^2-isogenies. But Koizumi's formula quickly gave us a formula to descend the level on the same abelian variety, which led to the algorithm for ℓ-isogenies in [CR15], and the start of AVIsogenies [BCR10] developed with my co-PhDs Gaetan Bisson and Romain Cosset.

During the implementation for dimension $g = 2$, I realised that our formula when using differential additions to compute the multiples $nP \widetilde{+} mQ$ of two points generating the kernel did not depend on the order of P and Q. I had the intuition that it was because of the commutativity of the elements of the theta group above the kernel (since the

[7] Analytically this corresponds to the condition of holomorphy at the cusps, and this is automatic by Koecher's principle when $g > 1$. When $g = 1$, holomorphy at the boundary can tested at the Tate curve.

[8] By the way I should probably integrate the code I wrote at the time to convert from Mumford coordinates to these rational projective coordinates in [BCR10].

kernel is assumed isotropic), and realised that when applied to P and Q non isotropic, this gave us a new formula to compute the Weil pairing via its avatar as the commutator pairing of the theta group. We proved this in [LR10], and also gave a formula for the Tate pairing. We later made the link with the standard construction of these pairings via Miller's algorithm in [LR15a], so that we could give an algorithm to compute the different variants (ate, optimal ate, …) introduced in cryptography. The PhD [Tra14] also explains the link between our algorithms and elliptic nets.

After my PhD I did an internship at Microsoft, and then a postdoc at Inria Bordeaux and then another one at Microsoft, between 2010–2012. At Microsoft, I worked with Kristin Lauter to compute class polynomials for $g = 2$ via the CRT method using isogenies [LR13]. As a postdoc there, I also implemented an homomorphic scheme based on RLWE using Microsoft's internal library, using negacyclic convolutions to speed up multiplications. Unfortunately I don't quite remember the timings we obtained to compute an homomorphic multiplication. When Gaetan Bisson was also an intern there, we developed another method to compute a maximal genus 2 curve (ie a curve whose endomorphism ring of its Jacobian is the maximal order), using "horizontal isogenies" rather than "vertical isogenies" to determine how to go up. This approach was never published because to go up in practice we still needed to compute the kernel, hence the ℓ-torsion, so this did not gain much. Now that we have modular polynomials we should revisit this. I also worked with Shamir to use comparative side channels attacks against Edward curves. This was presented at the Crypto 2011 rump session. I did not publish the paper because I felt the attack was not strong enough, but retrospectively this was a completely stupid idea: one should not pass the occasion to collaborate with a world class cryptographer.

According to `git`, we also started the paper [BCR11] explaining some tips and tricks in our computations with AVIsogenies (how to find the kernels, how to adjust the formula in characteristic two) in February 2011. It is still not published (even in preprint form), and this time I don't have any good excuse for that except that we have been too lazy to polish the few remaining details (and Romain Cosset left research for academia). I also started around this time (November 2011) with Christophe Ritzenthaler the project to compute Serre's obstruction algebraically. This obstruction allows to determine whether the Jacobian of a quartic curve descend to the base field *as a Jacobian* (the curve C and the abelian variety $A = \mathrm{Jac}(C)$ both descend over a finite field, but the Jacobian of the descent of C over the base field may only be a quadratic twist of the descent of A). The idea was to find genus 3 curves with many points by computing Jacobians isogenous to the threefold power of an elliptic curve with many points. A blocking point was how to ensure we were computing the full isogeny class to get all the curves, this was finally resolved recently and published in [KNR+20b].

I was recruited as an Inria CR in 2011 and took the position in Mars 2012. This is an important shift because I started to take M2 students and then PhD students. I have had three PhD students: Enea Milio, Abdoulaye Maiga and Jean Kieffer. I have been very lucky to find excellent PhD students that were a joy to work with (both academically *and* socially). I cosupervised Milio with Andreas Enge, and he defended in 2015. Maiga's supervision is unofficial, because we did not receive the Marie-Curie fellowship that would have allowed a joint supervision between Dakkar and Bordeaux. I cosupervise Kieffer with Aurel Page (prior to this HDR I obtained an ADT which allows me to be the main supervisor). Maiga and Kieffer will defend this year. Looking back at the masters or other internship projects I proposed, they reflect pretty well the research subjects I was interested in at the time.

My first master subject in 2012 was done by Ilaria Lovato, on "Computing Modular Polynomials with Theta Functions". She did not pursue a PhD (she went to be a a scientific journalist in Italia), but the subject was picked up by Enea Milio: "Calcul de polynômes modulaires en dimension 2". In [Mil15a], he gave a quasi-linear algorithm to compute Siegel modular polynomials (using Igusa invariants or theta constants) via evaluation/interpolation. As expected, theta constants gave much smaller modular polynomials (3000 times smaller for $\ell = 3$), so he could push the computations up to $\ell = 3$ with Igusa invariants, and $\ell = 7$ with theta constants. We later extended this algorithm in [MR20b] to handle Hilbert modular polynomials. (The paper was published recently, and the first preprint version was in 2017, but he had already computed Hilbert modular polynomials in July 2014. Since the paper was too long we had to cut out an appendix on the denominator of these polynomials [MR19], which we should probably publish separately.) Milio computed Hilbert modular polynomials for $\mathbb{Q}(\sqrt{2})$, $\mathbb{Q}(\sqrt{3})$ and $\mathbb{Q}(\sqrt{5})$, they are much smaller than Siegel polynomials and he went up to norm 97 for $\mathbb{Q}(\sqrt{2})$ and 59 for $\mathbb{Q}(\sqrt{5})$. He then did a postdoc in Nancy, and then at EPFL.

At roughly the same time Chloe Martindale, supervised by Marco Streng and cosupervised by Andreas Enge was also working on Hilbert modular polynomials using a different strategy. When she went to visit us in Bordeaux in 2014–2015, I worked with her, Sorina Ionica who was a postdoc at the time, and Marco Streng about isogeny graphs of abelian surfaces (from the CM point of view) [IMR+14]. This work was not published because meanwhile [BJW17] had very similar results to us using Tate modules.

In 2013, Giulio Di Piazza worked on "Arithmetic on Jacobians of algebraic curves". The goal was to look in more details at the arithmetic of curves of small genus, and then study the general purpose algorithms (based on

computing Riemann-Roch spaces). Piazza looked at Hess' algorithm [Hes02] but did not have time to look at Khuri Makdisi's algorithms [Khu04; Khu07]. He now works at the European Food Safety Authority.

In 2014, Illaria Chillotti worked on "Pairings over elliptic curves using isogenies". The goal was to study the relationship between pairings and isogenies, and see if we could get fast formula to compute the Weil-Cartier pairing. With the advent of isogeny based cryptography I think this is even more important, but unfortunately I still do not know fast formulae. Chillotti then did a Phd at Versailles on "Towards efficient and secure Fully Homomorphic Encryption and cloud computing".

In 2016, Liu Zhengying (this time as an internship for the third year of Polytechnique) worked on "Height of class polynomials". The goal was to study in more detail the height of class polynomials of imaginary quartic fields (especially when using other class invariants). I was also interested in the height of modular polynomials, but Zhengying did not have the time to pursue this. This was later picked up by Kieffer. Zhengying then did a PhD on "Automation du design des réseaux de neurones profonds".

In 2016, I also worked with Andreas Enge's master student Gregor Seiler on a CRT approach for the computation of ray class fields of quadratic imaginary fields [ERS16].

In 2016 I started to unofficially supervise Abdoulaye Maiga on "Computing canonical lift of abelian surfaces". There were some difficulties along the way for funding (we did not receive the Marie-Curie Eiffel fellowship), but Maiga persevered admirably. In [MR20a] we explain how to compute canonical lift for abelian surfaces using either Siegel modular polynomials or Hilbert modular polynomials (and then lifting the Verschiebung for point counting) in the case of odd characteristic. Then in [MR21] we adapt these algorithms to the special case of characteristic 2, the Siegel modular invariants classically used for the modular polynomials or class polynomials having bad reduction in characteristic 2. This last paper has been accepted first, but our computations where first done in odd characteristic.

Kieffer started his PhD on "Computing isogenies between abelian surfaces and applications" in 2018. He had an impressive array of results since he started: deriving isogenies between abelian surfaces using modular polynomials [KPR20], evaluating the height and degree of modular polynomials on any Shimura variety of PEL type [Kie20a], with more precise bounds for Hilbert and Shimura modular polynomials of abelian surfaces (along with an interesting bound on the height of a rational function when many of its evaluation points are small), fast evaluation of modular polynomials for abelian surfaces [Kie20b], and a positive answer to a conjecture by Dupont on the "topological" sign choices of the Borchardt mean when $g = 2$ [Kie20c]. Kieffer is currently implementing a SEA like algorithm for abelian surfaces, which will give an (heuristic) complexity of $\widetilde{O}_K(\log q^4)$ for counting the points of an abelian surface $A/\mathbb{F}_q$ with real multiplication by K (the constants depending on K). Following his work, a lot of topics in the algorithmic of abelian surfaces have reached maturity with their elliptic curve counterparts.

In parallel, I continued working on isogenies. After a talk on computing cyclic isogenies using real multiplication [Rob13] in 2013, Dimitar Jetchev contacted me to work this out practically and implement the algorithm with his PhD students Alina Dudeanu and Marius Vuille in [DJR+17].

Of course, I also continued my collaboration with David Lubicz. In 2014 we wrote [LR15b] which explain how to compute isogenies given only equations of the kernel (due to the way the algorithm of [LR12; CR15] works when having a basis of the kernel, the adaptation was more complicated than a simple resultant computation in order to apply it to when only the kernel equations are given). In 2014 we also wrote [LR16], where we develop the arithmetic of Kummer varieties given by a symmetric theta structure of level 2. In [LR20b] (started in 2019 with a working example when $g = 2$) we revisit how to get the full conjugacy class of the Frobenius when computing canonical lifts, rather than just the product of its inversible eigenvalues. We are currently writing [LR20a] (also started in 2019 with some working examples) about how to go up in level with theta functions, and also a faster way (than Koizumi's formula) to descend level, which gives us quasi-optimal isogeny computations in all cases (and not just when ℓ is a sum of two squares as in [CR15]).

We also have some partial results on the algebraic choice of sign in the agm and in Thomae's formulas, getting equations for the image of the theta modular correspondances, and algorithms to get equations of Kummer varieties in the theta model. We also have a project with Xavier Caruso on computing p-adic Hecke correspondances.

The Industrial ANR Project Simpatic, "SIM et théorie des couplages pour la sécurité de l'information et des communications" was a motivation to write a book on pairings, I contributed to some chapters in 2017 [Rob17].

Responsabilities

When growing older a double sentence is that we often have more academic responsibilities. I was a joint leader (with Tony Ezome) of the Lirimia international team Macisa and then Fast from 2014 to 2019. I was a member of the Jury Agregation de Mathématiques from 2014 to 2020, and in charge since 2016 with Alain Couvreur of the

Option C "Algèbre et Calcul Formel". Recently I have been coopted by the Lfant Inria team to present the new team project (Lfant being 12 years old has to stop).

We also need to find funding (participating to ANR projects and ERC is nice, as long as you are not the coordinator!). I have candidated three times as a coordinator to an AAP ANR project (involving Bordeaux, Marseille, Nancy, Rennes, Versailles), it was preselectionned twice but never selectionned. So I candidated to a JCJC ANR project instead, this was accepted (this is ANR Project Ciao) in 2019.

Other results and failures

Working in the Inria Lfant team at Institut de Mathématiques de Bordeaux and its strong software culture, is always a nice opportunity to solve fun questions arising from implementations (especially in Pari/GP), not research level but interesting, namely:

- Identify needed functions for elliptic curve in Pari/GP. The list I gave to Bill Allombert and Karim Belabas in 2011 has been implemented over the years, the new challenge are genus 2 curves.
- When implementing Miller's algorithm for the Tate pairing, the intermediate computations give intermediate zeroes and poles, so the computation can fail. This is not a problem in the cryptographic setting because there is a wealth of points we can use for translation, but this can become a problem when computing the Tate pairing on curves without many points. A solution is to simply work out the Laurent series expansion (along uniformisers) of the functions involved in the pairing computations, see Lemma 3.5.3 for the formulae.
- If we have an elliptic curve E over $\mathbb{F}_q$ whose j-invariant is defined over a smaller field (say $\mathbb{F}_p$), then to do point counting on E we really want to work over $\mathbb{F}_p$ rather than over $\mathbb{F}_q$. This is not hard: let $E'/\mathbb{F}_p$ be the elliptic curve such that $j(E) = j(E')$, we can count the number of points of E' over $\mathbb{F}_p$, this easily gives us its number of points over $\mathbb{F}_q$. Now E is a twist of E' over $\mathbb{F}_q$, so we just need to identify this twist as an element of $H^1(\mathbb{F}_q, \mathrm{Aut}(E'))$ and use it to get the correct twist of the Frobenius.

  As an easy example: if $E : y^2 = x^3 + Ax$ over $\mathbb{F}_q$, the j-invariant is 1728. So E' is given b $y^2 = x^3 + x$, and we know its Frobenius π' over $\mathbb{F}_q$. If $A = B^4$, then our element of $H^1(\mathbb{F}_q, \mu_4 = \mathrm{Aut}(E'))$ is given by $\sigma \mapsto \sigma(B)/B$, and $\pi = \pi'\zeta$ where $\sigma(\pi_q) = \zeta \in \overline{\mathbb{F}}_q$. As always, the most annoying case to treat explicitely are supersingular curve of characteristic two (the automorphism group can be of size 24 and isomorphisms are slightly less convenients to describe).

Not every research project is a success. I have tried, but failed, to break SIDH[9]. One of my idea was to use the formal group law to somehow identify the correct direction to follow. Another idea was as follow: suppose that Alice or Bob leaks the action on tangent space of its isogeny. For instance in the Montgomery model $By^2 = Cx^3 + Ax^2 + Cx$, the Kummer line is represented by the projective point $(A : C)$ [CLN16]. The isogeny computations on (A, C) are normalised, so if Alice sends (A, C) rather than A/C, she leaks the action on tangent spaces. Then we can encode the isogeny by a differential equation as in Section 4.7. This encodes the coefficients a_i of the isogeny developed as a power series in the uniformisers. The degree of the isogeny is then the smallest dimension in which a certain linear system on the a_i (the one encoding the rational reconstruction) becomes non inversible. The question is then whether a quantum computer can tell if a linear algebra problem of dimension N whose coefficients are encoded by a quantum circuit of polynomial length in $\log N$ can determine whether a solution exists in time subexponential in N, possibly by adapting [HHL09] (a difficulty is that our system is not unitary). I have not pursued this further because the hypothesis that there is a leak has no reason to happen in practice (it would increase the size of the messages, in fact one of the first compression scheme on SIDH was to send $j(E)$ rather than its coefficients), and there is an easy countermeasure anyway.

1.4 SOME USELESS TRIVIA

As mentioned already, this hdr is way longer than originally envisaged, but I have a tendency to be too verbose in my articles already. Some quick statistics, among my publications and preprints, using a similar class template: [LR10] is 21p., [FLR11] 37p., [LR12] 42p., [LR13] 29p., [CR15] 24p., [LR15a] 33p., [LR15a] 19p., [LR16] 20p.,

[9] I did not believe in its security at first: publishing the image of points leak informations, and the length of the isogeny path followed is too short for uniform mixing

[MR20b] 53p., [MR21] 28p., [DJR+17] 37p., [KPR20] 64p., [KNR+20b] 38p., [LR20b] 29p., [MR20a] 27p. for a total of 501p., ie 33p. by paper.

So this hdr only achieves a one third summary, which is not great not terrible. My excuse is that this does not count the non published results, and I had to give more details to explain the new results mentioned above (see Section 1.5 for more details): generalised Thomae formula from an explicit version of the theorem of the square, change of level and faster descent for theta functions, applications to faster isogeny formula, efficient computation of evaluated modular polynomial via an analytic or a CRT or a p-adic approach, applications to point counting via a SEA like approach, improvements to the canonical lift approach to point counting, improvements to the p-adic approach to compute class polynomials.

The real purpose of this Section is that I wanted to mention the following bragging rights: at the time of writing, I am the only person in the world[10] with accepted commits both in `git`, the stacks project [Stacks] and `iproute2`[11].!

1.5 OUTLINE

The outline of this document is as follow. The Chapters are written to be (mostly) independent from each other, at the cost of some redundancy.

We warn that, while there are several new results, they are far less polished (and sometime only heuristic or prospective) than the published results. But I was too excited to not include them in this document.

In Chapter 2, we first study the arithmetic of abelian varieties. This regroup several results that were used as we needed them in [LR12; CR15; LR10; LR15a; LR16]. For reasons further explained in Section 2.1, we first begin by a brief explanation of Mumford's theory of algebraic theta functions. This allows us to develop a general recipe (Recipe 2.5.3) on how to recover the theta coordinates of $\mathcal{L}$ if we have a basis of sections and the explicit action of the theta group $G(\mathcal{L})$ on them. As a new result, we give in Algorithmic Hypothesis 2.9.2 general hypotheses which allow from a model induced by $\mathcal{L}$, to compute a basis of sections of $\mathcal{L}^n$ and the action of $G(\mathcal{L}^n)$ on them (provided we have the points of $K(\mathcal{L}^n)$). These hypotheses are satisfied by the model given by theta constants, or the model given by Jacobians (using the work of [CE14]). As a corollary we obtain a very general algorithmic Thomae formula, ie an algorithm to compute theta functions of level n on a Jacobian or on an abelian variety given by theta functions of lower level, provided we have the points of n-torsion. We also give faster formula for descending level (this will appear in [LR20a]), and as a corollary of changing level combined with Mumford's isogeny formula we get (fast) isogeny algorithms (this is a joint work with David Lubicz).

In Chapter 3, we summarize the results of [LR10; LR15a] on pairings for abelian variety. These articles dealt with the theta models, but it is straightforward to extend the algorithm on a model which satisfy Algorithmic Hypothesis 2.9.2. We also give formulas for elliptic curves from [Rob17]. The theoretical aspect of these pairings is in [Rob21, Chapter 4]. It is customary in cryptography to consider Jacobians, because one can work only with divisors on curves and use Weil's reciprocity to prove the standard formulas for the pairing, but I explain in [Rob21, Section 4.1.2] why we can use Lang's reciprocity to get similar formulas for abelian varieties.

In Chapter 4, we give a general framework for isogeny computations from "first principles", ie only assuming Algorithmic Hypothesis 2.9.2. This provides a common generalisation of [LR12; CR15] and [CE14]. We also explain how to adapt these algorithms for cyclic isogenies. Thus this Chapter extends the results of [LR12; CR15; LR15b; LR20a] and gives more efficient algorithms for cyclic isogenies than [DJR+17]. We also explain how to recover isogenies from differential equations as in [KPR20].

Next we deal with moduli spaces. Our most important topic is modular correspondances and modular polynomials in Chapter 5. We explain (an improved version of) the theta modular correspondance studied in [FLR11]. We then explain the work of Milio on the computation of Siegel and Hilbert modular polynomials for abelian surfaces [Mil15a], [MR20b; MR19], and the work of Kieffer on bounds on their size [Kie20a] and how to evaluate them [Kie20b; Kie20c]. These computations were done via an analytic method, and a drawback is that quasi-linear computation via evaluation/interpolation requires being able to compute modular invariants and period matrices in quasi-linear time in the precision, which we only have for $g \leq 2$. As a new result we describe here alternative strategies using a CRT method or a p-adic method. We don't quite get a quasi-linear algorithm, but we formulate some strategies to obtain one. We then give applications of modular polynomials: determining isogenies as in [KPR20], by reinterpreting Elkies method as a computation of a deformation map induced by the modular correspondance. We then explain how to use this to speed up point counting. Kieffer is currently working on an implementation of the SEA like algorithm for point counting for abelian surfaces, but I spoil his results in Section 5.5: in the Siegel

[10] Now this is a lot less impressive if you look at the commits and see that they are completely trivials! In fact, I can reverse brag that one of my patch series sent to `git` was such a disaster that it was talked about in Git Rev News…

[11] I had to add `iproute2` because the intersection of the contributors to `git` and the stacks project is of cardinal two…

case he will have a complexity of $\widetilde{O}(\log^8 p)$ for point counting of an abelian surface $A/\mathbb{F}_p$ (as in the complexity of a Schoof approach [GS12], so one should look more closely at the logarithmic factors), but only $\widetilde{O}(\log^7 p)$ if the curve is given by small parameters. In the Hilbert case he will have a complexity of $\widetilde{O}(\log^4 p)$ compared to $\widetilde{O}(\log^5 p)$ from a Schoof approach [GKS11] (here the $O()$ notations hide constants depending on the RM field). In Section 5.5.5, I outline the work that remains for a strategy that could potentially give an $\widetilde{O}(\log^4 p)$ for point counting on hyperelliptic curves of genus g with RM. Another application is given to exploring isogeny graphs; this summarizes the results of [BCR11; IMR+14; KNR+20b]. We also outline what a potential Atkin like algorithm algorithm for abelian surfaces in the Siegel case could look like in Section 5.6.3.

In Chapter 6 we review the point counting algorithms based on canonical lifts. There are two slightly different algorithms on whether we use the theta modular correspondance [FLR11] or modular polynomials [MR20a; MR21]. A "new result"[12] is on how to use the change of level of Chapter 2 to speed up the initialisation phase of [FLR11]. We also describe the results of [LR20b] about how to get the full action on the tangent space rather than just its determinant. An exciting new result is how to do the same using only modular polynomials using the results from Section 5.4: this has the big advantage to not require to lift equations for the kernel of the Verschiebung, and for elliptic curves already yields a large speedup, see Table 6.1. In both cases, we recover a quasi-quadratic $\widetilde{O}(d^2)$ point counting algorithm over $\mathbb{F}_{p^d}$ (with constants depending on p), where the dependency on p can be controlled explicitly (eg by the cost of evaluating the modular polynomial Φ_p; this cost is an explicit polynomial in p).

In Chapter 7 we give a summary of the results of [LR13; BLR11; ERS16] about the CRT method to compute class polynomials. A new result is an improvement of the p-adic method to compute these polynomials, using isogenies to lift all CM points, rather than just lifting one of them and then doing an LLL step. The cost is then dominated by the initialisation step: finding one CM point over the residue field of a totally split prime.

Along the way, we answer some small questions or conjectures in the literature, and improve some results. Let me give pointers here for references. We answer a conjecture on the degrees of Cantor polynomials [AGS19b, § 6; Abe20, § 2] in Remark 4.7.3. We answer [Mil15a, Conjecture 41] on the absence of "parasite" factors in the denominator of modular polynomials when using theta functions in Section 5.3.2. We also answer [MR20b, Conjecture 5.2] (the appendix was cut-out from the publication because the paper was too long but is still available on HAL) about the irreducible components of the denominators of Hilbert modular polynomials of abelian surfaces at the end of Section 5.3.6. We answer in Remark 5.3.5 a question by Labrande in [Lab16, p. 168] about the precision needed to compute an isogenies between two elliptic curves defined over a number field via complex analytic method.

Since modular polynomials for abelian surfaces are too big to be used directly for application to point counting, I suggested to Kieffer to look at evaluating them on a modular invariant over $\mathbb{F}_q$ by lifting the invariant to a number field, evaluate there, and then reducing the evaluation. This is done in [Kie20b], see also Section 5.3.8. But it appears that this strategy is already interesting for ellliptic curves, see Remark 5.3.9. As applications, we explain in Section 5.4.3 how this gives a more efficient way to lift isogenies between elliptic curves than in [LS08], and at the end of Section 5.4.1, we give another approach to the determination of the isogeny between two given isogenous elliptic curves over $\mathbb{F}_q$ than in [DHP+16], which yields a better bound.

1.6 PERSPECTIVES

It is interesting, 10 years after my PhD defense, to look at the perspectives of [Rob10]. Things that have been realised are computing optimal pairings in the theta model, improved arithmetic on abelian varieties using the Kummer theta model, computing modular polynomials, computing class polynomials for abelian surfaces by the CRT method, and improving the initialisation step of point counting via canonical lifting over the theta modular correspondance. Things still missing are efficient triplication formula in a theta model of level n divisible by 3, transferring the DLP from hyperelliptic curves to quartic curves using isogenies in the theta model (in [KNR+20b] we implemented the formula to go from the theta constants to the curves, and also to descend the curve over its base field, but we have not yet implemented the formula which gives the divisor corresponding to a point given in theta coordinates; [BCR10] has only implemented the formulae for hyperelliptic curves). But see [Tia20] which is based on [CE14; Mil20] instead.

What is more interesting are the results I had not anticipated: the importance of real multiplication for the algorithms (smaller modular polynomials, cyclic isogenies), and most importantly the rise of isogeny based cryptography.

[12] This is an easy consequence of [LR12], but we forgot to state this application in that article.

We give detailed perspectives at the conclusion of each Chapter. Not unsurprisingly, I would like to extend and generalize our current algorithms, so a research program includes: computing integral model of Shimura varieties, efficient evaluation of modular forms, models of curves and abelian varieties (especially in higher dimension), algorithmic aspects of isogeny based cryptosystems (optimize isogenies, endomorphism rings, VDF, …).

In this Section, I will instead focus on more prospective perspectives:

- I would like to do more computations over number field, eg testing the BSD conjecture for abelian surfaces (as given by Tate, Beilinson, Bloch and Kato), or maybe even the paramodular conjecture.
- I would also like to learn more about formal proof software like Coq and Lean, eg to prove that the SIKE key exchange do give the same shared secret.
- While key exchange based on graphs are different from the DLP, a lot of protocols have been adapted to this new setting. A missing tool is pairings: it would be nice to have an analogue (and even define what an analogue should look like) for isogenies. Somewhat related: I would like to explore in more details Huang's proposal for trilinear maps [Hua18; Hua19].
- While we are on fancy things: the algorithmic theory of modular form is well developed (see eg modular symbols). What about the algorithmic aspects of topological modular forms (eg from the context of HoTT)? Could they ever have cryptographic applications?
- One of the biggest societal impact of cryptography in recent years has been the rise of Bitcoin and other cryptocurrencies. These have led to new cryptographic challenge (eg constructing VDF for proof of stake, or the challenge of constructing distributed or threshold EC-DSA since this is the signature used for Bitcoin rather than Schnorr signatures). Could isogeny based cryptosystems have other novel applications? The existence of VDF constructed from isogenies and pairings already show that not all applications are for post-quantum cryptography.

Part I

ALGORITHMS FOR ABELIAN VARIETIES

2 ARITHMETIC OF ABELIAN VARIETIES

CONTENTS

2.1 INTRODUCTION

In this Chapter, we study the arithmetic of abelian varieties, given a particular model. Complex abelian varieties are particularly convenient for two reasons: first they are completely determined by a lattice Λ. Secondly, if we represent Λ as $\mathbb{Z}^g \oplus \Omega\mathbb{Z}^g$ where Ω is in the Siegel space (this is essentially the same as choosing a principal symplectic form E on Λ), then we can use the analytic theta functions $\theta\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right](z, \Omega)$, which give both analytic coordinates on A, but also coordinates on the moduli space of abelian varieties via the theta constants $\theta\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right](0, \Omega)$.

A remarkable fact of the theory of abelian varieties is that most of the results on complex abelian varieties hold for a general abelian variety A/k, possibly making adjustments when dealing with level p in characteristic p. For instance $A[\ell](\bar{k})$ is of cardinal ℓ^{2g} in characteristic $p \neq \ell$, but $A[p](\bar{k})$ has at most p^g points. However $A[p]$ is always a finite flat commutative group scheme over k of rank p^{2g}.

One way to "recover" the lattice Λ algebraically is to instead use the Tate modules $T_\ell A$ and $\mathbf{D}_p A$ (here $\mathbf{D}_p A$ is the Dieudonné module associated to $A(p)$ rather than the "physical" Tate module $T_p(A)$). Like Λ, which is dual

of the singular homology $H^1(A,\mathbb{Z})$ in the complex case, the Tate module $T_\ell A$ is the dual of the étale cohomology $H^1_{\text{ét}}(A,\mathbb{Z}_\ell)$ and $T_p(A)$ (which is already defined contragradiently) is given by the crystalline cohomology $H^1_{crys}(A_k/W(k),\mathbb{Z}_p)$. We refer to [Rob21, Section 2.2.5] for more details.

In this Chapter we will explain how Mumford's theory [Mum66; Mum67a; Mum67b] gives an algebraic theory of theta functions. Since they are keys to a lot of the algorithmic result developed in this document, I give a quick summary of Mumford's result in Sections 2.2 to 2.7. Ideally I would have just referred to [Rob10], but in my thesis I only develop the theory of theta functions given by a symmetric theta structure on a totally symmetric line bundle. This conflates three different algorithmic tools:

1. the theta group $G(\mathcal{L})$ and its action on the section $\Gamma(A,\mathcal{L})$, which encode descent (and will be extremely useful for isogenies). We stress that the theta group exists for any model.

2. the notion of a (symmetric) theta structure, which gives an explicit isomorphism of $G(\mathcal{L})$ with an Heisenberg group $H(\delta)$. This is what allows to pick up theta functions as the unique (up to a constant) basis of $\Gamma(A,\mathcal{L})$ induced by an explicit description of the unique irreducible representation of $H(\delta)$. A drawback is that the isomorphism may only be defined over a field extension.

3. A symmetric theta structure when $\mathcal{L}$ is furthermore assumed to be totally symmetric (ie the square of a symmetric line bundle), hence of level n even. This notion is extremely useful because Mumford (and Kempf) give the equations of (a compactification of) the universal abelian scheme $\mathcal{X}_{g,n} \to \mathcal{A}_{g,n}$, where $\mathcal{A}_{g,n}$ is the moduli of abelian varieties with a symmetric level n theta structure. In particular the theta constants $\theta_i^A(0)$ are coordinates on $\mathcal{A}_{g,n}$, and Mumford gives both the equations that the theta constants need to satisfy, but also all the equations of A, given only the theta constants.

Having explicit equations is of course key for algorithmic applications. Moreover, since the theta structure allows to control the action of the theta group explicitly, we have very nice isogeny formula, see Theorem 2.5.6.

There are some drawbacks in working only with symmetric theta structure of even level. First, if we have a principal polarisation $\mathcal{L}$ on A and we want to compute an ℓ-isogeny with ℓ odd, we only need to know the action of the theta group $G(\mathcal{L}^\ell)$ to compute the isogeny, so we would like to work in level ℓ odd. Likewise, level $n=3$ is sufficient to get a very ample line bundle by Lefschetz theorem, but if we want n even we need to work with $n=4$. Finally for reasons that will become apparent in Section 2.6, the notion of symmetry is easier to handle (in particular it is canonical, hence invariant by the Galois action) when the level is odd. Apart from dealing with odd level theta structure, we may want to be able to compute the arithmetic of an abelian variety (addition, pairings, isogenies) on models not coming from theta structures (if only for rationality consideration). For instance we may want to work with Jacobians by using the curve directly and not imposing a theta structure. The arithmetic on Jacobians reduces to explicit Riemann-Roch on the curves, we refer to [Hes02; Khu04; Khu07; ACL20; ACL21] for efficient algorithms.

So I needed to extract what exactly was used in our algorithms for isogenies and pairings in [LR12; CR15; LR10; LR15a] using the theta models and see how to extend them to other models. Of course the drawback is that for other models it is harder to get a full set of equations; we cannot just compute a theta null point as in the theta model!

This allows to distinguish between the algorithmic applications of the theta group from that of the theta structures. For instance:

1. If we have an explicit way to compute the action of $G(\mathcal{L})$ on $\Gamma(A,\mathcal{L})$ and a section $s\in\Gamma(A,\mathcal{L})$, we can deduce a basis of $\Gamma(A,\mathcal{L})$. (This is simply because the action is irreducible).

2. If furthermore we fix a symplectic basis of $K(\mathcal{L})$ with respect to the commutator pairing $e_{\mathcal{L}}$, the explicit action of $G(\mathcal{L})$ allows us to construct explicitly the theta basis of $\Gamma(A,\mathcal{L})$ induced by a (symmetric) theta structure compatible with the basis, see Recipe 2.5.3

3. The explicit action of $G(\mathcal{L})$ also allows us to compute the isogeny by any kernel $K\subset K(\mathcal{L})$ isotropic for $e_{\mathcal{L}}$ (almost by definition of the theta group).

4. In Section 2.9, I explain how given the explicit action of $G(\mathcal{L})$ and an explicit version of the theorem of the square, we can compute both sections of $G(\mathcal{L}^\ell)$ and the explicit action of $G(\mathcal{L}^\ell)$ on these sections.

5. There are two models where I know how to compute Item 4. The theta model (of even level n) of course, where the explicit action of $G(\mathcal{L})$ is part of the definition and the theorem of the square is given by Riemann relations (in the form of differential additions). And the Jacobian model, where $\mathcal{L}$ is the polarisation induced

by the theta divisor Θ. Here the action of $G(\mathcal{L})$ is trivial since $\mathcal{L}$ is principal, and the theorem of the square is (implicitly) given by the wonderful article [CE14].

6. In particular, for both these models we can compute a theta structure for $G(\mathcal{L}^\ell)$ using Item 2 or compute the isogeny for K isotropic in $K(\mathcal{L}^\ell)$ using Item 3, eventually combining both to compute a theta model of the isogenous abelian variety.

   So we have a very general framework for isogeny computations, unifying the approach of [CE14] using Jacobian models and the approach of [LR12; CR15] using the theta model, which we will develop further in Chapter 4. We also have a general framework for generalised Thomae formula, in particular for Jacobians. In fact this gives not only the theta constant, but the theta coordinates of any point on the Jacobian. This last application is a new result.

The outline is as follow: we briefly describe complex abelian varieties in Section 2.2, polarisations and the theta group in Section 2.3, and algebraic theta functions in Section 2.4. Using the theta group for descent is in Section 2.5, as a special case we give Mumford's isogeny theorem for the theta model. Symmetric theta structures are described in Section 2.6, and Mumford's explicit equations (as a generalisation of Riemann's relations) in Section 2.7.

This (finally) allow us to do some algorithmic work: we explain how to compute additions (and generalisations) in Section 2.8. In Section 2.9 we explain how to apply the tools of Section 2.8 in other models. We describe how to apply Item 6 in practice in the theta model in Section 2.10, namely we give formula to change the level (both up and down) and to compute isogenies. In Section 2.11 we give a rationality criteria for a symmetric theta structure over a finite field, this refines [Rob10, § 3.7]. In Section 2.12 we describe what kind of arithmetic we can compute on Kummer varieties. Finally we give some perspectives in Section 2.13.

2.2 ABELIAN VARIETIES OVER $\mathbb{C}$

We very briefly summarizes the core structure of complex abelian varieties. Longer summaries can be found in [Rob21, Section 2.1] and [Rob10, § 2], and good references are [Mum70a, Chapter 1], [BL04].

A complex abelian variety is a torus $A = \mathbb{C}^g/\Lambda$, where Λ is a $\mathbb{Z}$-lattice, that is algebraisable (which means that there is an embedding of A into projective space). By Appell-Humbert's theorem, this last condition is equivalent to the existence of a positive Hermitian form H on $\mathbb{C}^g$ such that if we let $E = \Im H$ be the symplectic form associated to H, $E(\Lambda, \Lambda) \subset \mathbb{Z}$.

In summary, a complex abelian variety $A/\mathbb{C}$ is given by three datum (we refer to [Rob21, Section 2.1] for more details):

- *Linear data*: A complex vector space V of dimension g;
- *Arithmetic data*: A $\mathbb{Z}$-lattice Λ of rank $2g$ in V;
- *Quadratic data*: A positive Hermitian form H such that $E(\Lambda, \Lambda) \subset \mathbb{Z}$.

The hermitian form is not intrinsic to the abelian variety, but it allows:

- To define projective coordinates on A (ie functions on $V = \mathbb{C}^g$ which satisfy an automorphic equation with respect to Λ, see [Rob10, § 2.3][1]);
- A morphism Φ_H (a polarisation) from A to its dual abelian variety $\widehat{A}$: $x \mapsto H(x, \cdot)$ [Rob10, § 2.4].

So H is intrinsic to the notion of polarized abelian variety. The kernel of the polarisation Φ_H is given by $\Lambda^\star/\Lambda$ where $\Lambda^* = \{x \in \mathbb{C}^g \mid E(x, \Lambda) \subset \mathbb{Z}\}$ is the $\mathbb{Z}$-orthogonal of Λ.

We can describe principally polarised abelian varieties very concretely as given by a period matrices $\Omega \in \mathfrak{H}_g$, the Siegel space of symmetric matrices Ω such that $\Im\Omega > 0$. The lattice is then $\Lambda = \mathbb{Z}^g \oplus \Omega\mathbb{Z}^g$, and the principal polarisation is $H = (\Im\Omega)^{-1}$ (see [Rob10, §2.5] for the description of the moduli space of polarisations of type $(\delta_1, \dots, \delta_g)$).

Using the period matrices, we can then construct theta functions, which give the projective coordinates corresponding to H. For $\Omega \in \mathfrak{H}_g$ and $a, b \in \mathbb{Q}^g$, analytic theta functions are defined as

$$\theta\left[\begin{smallmatrix} a \\ b \end{smallmatrix}\right](z, \Omega) = \sum_{n \in \mathbb{Z}^g} e^{\pi i {}^t(n+a)\Omega(n+a) + 2\pi i {}^t(n+a)(z+b)}. \tag{2.1}$$

[1] Technically the polarisation H only induces an ample line bundle when we fix a semi-character for it. In other words H only determines the algebraic equivalence class of the line bundle. But for an ample line bundle, all algebraically equivalent line bundles are translate of it since the polarisation is an isogeny, so we gloss over this detail.

A basis of sections for the polarisation nH is then given by $(\theta \left[\begin{smallmatrix} 0 \\ b \end{smallmatrix}\right](\cdot, \Omega/n))_{b\in Z(\overline{n})}$ where $Z(\overline{n}) = (\mathbb{Z}/n\mathbb{Z})^g$. By Lefschetz theorem, they give a projective embedding whenever $n \geq 3$. It is often convenient to consider other basis given for $n = n_1 n_2$ by $(\theta \left[\begin{smallmatrix} a/n_1 \\ b/n_2 \end{smallmatrix}\right](n_1\cdot, n_1\Omega/n_2))_{a\in Z(n_1), b\in Z(n_2)}$ (see [Rob10, Exemple 4.4.9], for instance if $n = 4$ it is customary to take $n_1 = n_2 = 2$).

The lattice Λ and the theta functions play a key role in making the theory of complex abelian varieties over $\mathbb{C}$ explicit and algorithmic [BL04; Mum83; Mum84]. Luckily they can be (mostly) extended over an arbitrary field k (by relying on much deeper results from algebraic geometry). When A/k be an abelian variety over a field k, The lattice Λ is replaced by the Tate modules $T_\ell(A)$ and $\mathbf{D}_p(A)$ (when k is of characteristic p). If $\mathcal{L}$ is an ample line bundle on A, the polarisation H corresponding to it is instead represented by the induced isogeny $\Phi_{\mathcal{L}} A \to \widehat{A}$ and the induced Weil pairing on the Tate modules $T_\ell(A)$. Finally Mumford constructed an algebraic theory of theta functions in [Mum66; Mum67a; Mum67b]. We now explain this.

2.3 COORDINATES AND POLARISATIONS

We will denote by k a field, which will be assumed perfect for simplicity[2], and $\overline{k}$ its algebraic closure. We denote by p the characteristic of k. By abuse of notation, we say that a number n is prime to p whenever $p = 0$ or when $p > 0$ and $n \wedge p = 1$.

Let A/k be an abelian variety. To represent points $A(\overline{k})$ of A we need coordinates. Since A is proper, there is no nonconstant morphism from A to an affine variety, hence in particular there is no global affine coordinates (ie no affine embedding $A \to \mathbb{A}^N_k$). So we either need to work with local affine coordinates (for instance Mumford coordinates on the Jacobian of an hyperelliptic curves), or with projective coordinates.

Projective coordinates can be characterized as follow: assume that we have a map (defined everywhere), $\phi : A \to \mathbb{P}^N_k$. Then the coordinates $X_0, \dots, X_N$ on $\mathbb{P}^N_k$ induces coordinates $X_i \circ \phi$ on A. More precisely: X_i are global sections of the line bundle $O_{\mathbb{P}^N_k}(1)$, and $X_i \circ \phi$ are global sections of its pullback $\mathcal{L} = \phi^* O_{\mathbb{P}^N_k}(1)$.

Conversely, if $\mathcal{L}$ is a line bundle on A, we may use its global sections (assuming it has some) to construct a birational map from A to $\mathbb{P}^N = \mathbb{P}(\Gamma(\mathcal{L}))$.

Proposition 2.3.1 (Lefschetz). *Let $\mathcal{L}$ be an ample line bundle on A. Then*

- *$\mathcal{L}^2$ is always base point free. This means that $A \to \mathbb{P}(\Gamma(\mathcal{L}^2))$ is defined everywhere on A. If $\mathcal{L}$ is indecomposable, this map is an embedding if $\mathcal{L}$ is not principal, and an embedding of the Kummer variety $A/\pm 1$ if $\mathcal{L}$ is principal.*
- *$\mathcal{L}^3$ is always very ample. This means that $A \to \mathbb{P}(\Gamma(\mathcal{L}^3))$ is an embedding.*

Proof. This is [Mum70a, §17] and [BL04, §4.4, §4.5, §4.8]. See also [Rob21, Theorems 2.1.2 and 2.2.11 and Remark 2.1.3]. □

For a complex abelian variety, (the algebraic equivalence class of) a line bundle is represented by its associated Hermitian form H, which induces a polarisation $A \to \widehat{A}$. Algebraically we can construct the polarisation associated to a line bundle $\mathcal{L}$ as follow. The dual abelian variety is defined as $\widehat{A} = \mathrm{Pic}^0(A)$, and the polarisation is given by $\Phi_{\mathcal{L}}(P) = t_P^*\mathcal{L} \otimes \mathcal{L}^{-1}$. If $\mathcal{L}$ is ample, $\Phi_{\mathcal{L}}$ is an isogeny, which means that every line bundle algebraically equivalent to $\mathcal{L}$ (ie of the form $\mathcal{L}' = \mathcal{L} \otimes \mathcal{M}$ where $\mathcal{M} \in \mathrm{Pic}^0(A)$) is a translate $t_P^*\mathcal{L}$ of $\mathcal{L}$.

If $\mathcal{P}$ is the Poincare bundle on $A \times \widehat{A}$, and $y \in \widehat{A}$, we denote by $\mathcal{P}_y$ its restriction to the fiber $\mathcal{P}|_{A\times\{y\}}$; this is the line bundle algebraically equivalent to 0 on A represented by y. If $\Phi_{\mathcal{L}}(x) = y$, then $t_x^*\mathcal{L} \simeq \mathcal{L} \otimes \mathcal{P}_y$.

Furthermore, if $\Phi_{\mathcal{L}}$ is a separable isogeny, the kernel $K(\mathcal{L})(\overline{k})$ of the polarisation is of the form $(\mathbb{Z}^g/\delta\mathbb{Z}^g)^2$ where $\delta = (\delta_1, \dots, \delta_g)$, with $\delta_1 \mid \dots, \delta_g$ defines the type (or level) of the polarisation.

For simplicity we will mainly deal with principally polarised abelian varieties[3], ie with an ample line bundle $\mathcal{L}_0$ such that $\Phi_{\mathcal{L}_0}$ is an isomorphism, and consider line bundles of the form $\mathcal{L} = \mathcal{L}_0^n$, with n prime to p. Then $\mathcal{L}$ is of level n, ie $K(\mathcal{L})(\overline{k}) \simeq (\mathbb{Z}^g/n\mathbb{Z}^g)^2$. By [Mum66], a line bundle $\mathcal{L}$ is of the form ${\mathcal{L}_0}^m$ if and only if $A[m] \subset K(\mathcal{L})$. So conversely if we have a line bundle $\mathcal{L}$ of level n, it is the n-th power of a principal polarisation $\mathcal{L}_0$.

There is a canonical pairing $e_{\mathcal{L}}$ on $K(\mathcal{L})$, see Chapter 3. Gluing together the pairings $e_{\mathcal{L}^n}$ along the $K(\mathcal{L}^n)$ yield a pairing on the Tate modules $T_\ell(A)$. This is the algebraic interpretation of the hermitian form H.

[2] In practice k will be a finite field, a number field, or the complex numbers.

[3] Of course Mumford's theory of algebraic theta function holds for a general separable polarisation, see [Rob10, § 3] for a summary.

By definition, if $P \in K(\mathcal{L})$, $t_P^*\mathcal{L} \simeq \mathcal{L}$. Of great importance in the theory of descent (which will be used to construct isogenies in Chapter 4) and the theory of algebraic theta function is Mumford's theta group which encode these isomorphisms:

Definition 2.3.2. The group $G(\mathcal{L})(\bar{k})$ is the set $\{(P, \psi_P) \mid P \in K(\mathcal{L}) \text{ and } \psi_P : \mathcal{L} \to t_P^*\mathcal{L} \text{ is an isomorphism}\}$, with the natural composition law $(P, \psi_P) \cdot (Q, \psi_Q) = (P + Q, t_P^*\psi_Q \circ \psi_P)$. (The definition is functorial, hence does define a finite flat group $G(\mathcal{L})$ over k).

The theta group acts on the global sections $s \in \Gamma(A, \mathcal{L})$ of $\mathcal{L}$ via $(P, \psi_P) \cdot s = t_{-P}^*\psi_P(s)$.

Some terminology: if $\mathcal{L}$ is separable, a subgroup $K \subset K(\mathcal{L})$ is isotropic if $e_{\mathcal{L}}(x, y) = 1$ for all $x, y \in K(\bar{k})$. It is maximal isotropic if it is not included in a larger isotropic subgroup. A symplectic decomposition is a decomposition $K(\mathcal{L}) = K_1(\mathcal{L}) \oplus K_2(\mathcal{L})$ where $K_i(\mathcal{L})$ are isotropic (they are then necessarily maximal isotropic). We need to be careful that a maximal isotropic group K' needs not come from a symplectic decomposition, for instance if $\mathcal{L} = \mathcal{L}_0^n$ is a power of a principal polarisation of level n and $n = m^2$ is a square, $A[m]$ is maximal isotropic in $K(\mathcal{L}) = A[n]$. Since the distinction is important, we say that K is maximal totally isotropic if there is a symplectic decomposition $K(\mathcal{L}) = K \oplus K'$ (we call K' a symplectic supplement of K), and K is totally isotropic if it is included in a maximal totally isotropic group.

2.4 ALGEBRAIC THETA FUNCTIONS

We explain Mumford's theory of algebraic theta functions, developed in [Mum66; Mum67a; Mum67b; Mum69; Mum91]. The articles by Kempf [Kem88; Kem89a; Kem89b; Kem90; Kem92] also provide useful refinements. We fix an n prime to p, and an ample line bundle $\mathcal{L}$ of level n.

Let $Z(\overline{n}) = \mathbb{Z}^g/n\mathbb{Z}^g$, and $\hat{Z}(\overline{n}) = \mu_n^g$ be its Cartier dual ($\hat{Z}(\overline{n})(\bar{k})$ is simply the group of characters on $Z(\overline{n})$), and $K(\overline{n}) = Z(\overline{n}) \times \hat{Z}(\overline{n})$. The canonical duality $\langle x_1, x_2\rangle = x_2(x_1)$ on $Z(\overline{n}) \times \hat{Z}(\overline{n})$, induces a canonical symplectic pairing e_n on $K(\overline{n})$ via $e_n((x_1, x_2), (y_1, y_2)) = \langle x_1, y_2\rangle\langle y_1, x_2\rangle^{-1}$. We may conveniently recover $\langle x_1, x_2\rangle$ as $e_n((x_1, 0), (0, x_2))$.

Since $e_{\mathcal{L}}$ is non degenerate, there is a symplectic isomorphism $(K(\mathcal{L}), e_{\mathcal{L}}) \simeq (K(\overline{n}), e_n)$. We will often denote $K(\overline{n})_1 = Z(\overline{n})$ and $K(\overline{n})_2 = \hat{Z}(\overline{n})$ so that $K(\overline{n}) = K(\overline{n})_1 \oplus K(\overline{n})_2$ is the canonical symplectic decomposition of $K(\overline{n})$. The symplectic isomorphism above then induces a symplectic decomposition $K(\mathcal{L}) = K(\mathcal{L})_1 \oplus K(\mathcal{L})_2$.

We have a canonical exact sequence

$$1 \to \bar{k}^* \to G(\mathcal{L})(\bar{k}) \to K(\mathcal{L})(\bar{k}) \to 0 \tag{2.2}$$

where $\bar{k}^* = Z(G(\mathcal{L})(\bar{k}))$ is the centralizer of $G(\mathcal{L})$. Since $G(\mathcal{L})$ is a central extension of $K(\mathcal{L})$, it is represented by a 2-cocycle $\psi : K(\mathcal{L})^2 \to \bar{k}$.

But on $K(\overline{n})$ there is a canonical 2-cocycle given by $\psi((x_1, x_2), (y_1, y_2)) = \langle x_1, y_2\rangle$, which corresponds to the Heisenberg group $\mathcal{H}(\overline{n}) = \mathbb{G}_m \times Z(\overline{n}) \times \hat{Z}(\overline{n})$, with group law on $\mathcal{H}(\overline{n})(\bar{k})$ given by $(\alpha, x) \cdot (\beta, y) = (\alpha\beta\psi(x, y), x + y) = (\alpha\beta\langle x_1, y_2\rangle, x_1 + y_1, x_2 + y_2)$. Since $e_n(x, y) = \frac{\psi(x,y)}{\psi(y,x)}$, we may recover this pairing as the commutator pairing $e_n(x, y) = \tilde{x}\tilde{y}\tilde{x}^{-1}\tilde{y}^{-1}$ for any lifts $\tilde{x}, \tilde{y} \in \mathcal{H}(\overline{n})(\bar{k})$ of $x, y \in K(\overline{n})(\bar{k})$.

Theorem 2.4.1 ([Mum66]). *Any isomorphism $\overline{\Theta}_{\mathcal{L}} : (K(\overline{n}), e_n) \to (K(\mathcal{L}), e_{\mathcal{L}})$ (over $\bar{k}$) extends to an isomorphism $\Theta_{\mathcal{L}} : \mathcal{H}(\overline{n}) \to G(\mathcal{L})$. The isomorphism $\Theta_{\mathcal{L}}$ is said to be a theta structure (of level n) for $G(\mathcal{L})$ (or $(A, \mathcal{L})$).*

Furthermore the action of $G(\mathcal{L})$ on $V = \Gamma(A, \mathcal{L})$ is irreducible, hence is isomorphic (unique up to the action of $\mathbb{G}_m$) to the unique irreducible action of $\mathcal{H}(\overline{n})$ on W (a vector space of dimension n^g) such that $\mathbb{G}_m$ acts naturally on W.

This Theorem has two consequences. The first is that the pairing $e_{\mathcal{L}}$ may be recovered as the commutator pairing on $G(\mathcal{L})$. The second is that once we have chosen a theta structure $\Theta_{\mathcal{L}}$, there is a canonical way to fix a basis of sections $(\theta_i)_{i \in Z(\overline{n})}$ of $\mathcal{L}$ (over $\bar{k}$). For instance, if we choose for W, $W = \mathrm{Hom}(Z(\overline{n}), \mathbb{G}_m)$ with the representation of $\mathcal{H}(\overline{n})$ on $\bar{k}$-points given by $(\alpha, x_1, x_2) \cdot f(y) = \alpha\langle y, -x_2\rangle f(y - x_1)$, then θ_i is the unique basis (up to the action of $\mathbb{G}_m$) such that $\Theta_{\mathcal{L}}(\alpha, x_1, x_2)\theta_i = \alpha\langle i + x_1, -x_2\rangle\theta_{i+x_1}$. These are Mumford's algebraic theta functions.

In particular, the action by translation by the points of $K(\mathcal{L})$, ie of n-torsion, is completely specified when working with these theta coordinates. It is convenient to rewrite the action above using $e_{\mathcal{L}}$ rather than as $\langle x_1 + i, -x_2\rangle = e_n((x_1 + i, 0), (0, -x_2))$. So via the isomorphism $\overline{\Theta}_{\mathcal{L}}$, we may reindex our theta function by $(\theta_i)_{i \in K_1(\mathcal{L})}$ and then we have if $u \in K(\mathcal{L})(\bar{k})$, $(\theta_i(x - u))_{i \in K_1(\mathcal{L})} = (e_{\mathcal{L}}(i + u_1, -u_2)\theta_{i+u_1}(x))_{i \in K_1(\mathcal{L})} \in \mathbb{P}(V)$.

One very important point is that the theta structure induces more: the projective action of $K(\mathcal{L})$ on $\mathbb{P}(V)$ lift to an affine action of $G(\mathcal{L})$ on V, and the theta structure fix a canonical basis (up to a constant) for this affine action, not only for the projective action.

We note that the theta structure $\Theta_{\mathcal{L}}$ induces via $\overline{\Theta}_{\mathcal{L}}$ a symplectic basis on $K(\mathcal{L})$. But if we index the theta functions by $K_1(\mathcal{L})$ the only really important data for the characterisation of the theta functions is [Rob10, § 3.3]:

- The symplectic decomposition $K(\mathcal{L}) = K_1(\mathcal{L}) \oplus K_2(\mathcal{L})$;
- The choice of lifts $\widetilde{K}_i(\mathcal{L})$ of $K_i(\mathcal{L})$ in $G(\mathcal{L})$ for $i = 1, 2$.

We call this data an *level structure*. A further choice of symplectic basis (ie a choice of theta structure) then just corresponds to the level structure and an isomorphism $Z(\overline{n}) \simeq K_1(\mathcal{L})$, ie a numbering of theta functions.

Remark 2.4.2. It is interesting to describe the automorphisms of the theta group. First we note that the translation by $c \in A(\overline{k})$ yields an isomorphism between $G(\mathcal{L})$ and $G(t_c^*\mathcal{L})$. If $\psi : \mathcal{L} \to \mathcal{L}'$ is an isomorphism of line bundles, it induces an isomorphism between $G(\mathcal{L})$ and $G(\mathcal{L}')$. When $c \in K(\mathcal{L})(\overline{k})$, we can combine the two isomorphisms to get an automorphism conj_c of $G(\mathcal{L})$. This is simply the action by conjugation of any lift $\tilde{c} \in G(\mathcal{L})(\overline{k})$ of c [Rob10, p. 45]. Concretely, $\mathrm{conj}_c \cdot (x, \psi) = (x, e_{\mathcal{L}}(c, x)\psi)$ (hence does not depends on $\tilde{c}$). Every automorphism of $G(\mathcal{L})$ inducing the identity on $K(\mathcal{L})$ is of this type [Rob10, Proposition 3.5.1]. More generally we have the following exact sequence for the Heisenberg group of level δ:

$$0 \longrightarrow K(\delta) \longrightarrow \mathrm{Aut}(H(\delta)) \longrightarrow \mathrm{Sp}(K(\delta)) \longrightarrow 0.$$

And the action of conj_c on the theta functions is given by [Rob10, § 3.5]

$$\theta'_i = \langle -i, c_2 \rangle \theta_{c_1+i}. \tag{2.3}$$

Next every symplectic automorphism of $K(\mathcal{L})$ lift to an automorphism of the theta group (since we can always find level subgroups when $k = \overline{k}$), a precise description of these lifts is in [Rob10, Remarque 3.5.2].

One important automorphism is the automorphism S which transposes the level subgroups $\widetilde{K}_1(\mathcal{L})$ and $\widetilde{K}_2(\mathcal{L})$, and which acts on the theta functions by:

$$\theta'_i = \sum_{j \in Z(\overline{n})} \langle -j, \sigma(i) \rangle \theta_j. \tag{2.4}$$

The Segre embedding will be algorithmically very useful Chapter 4. This comes in part from the fact that it is induced by a product theta structure:

Lemma 2.4.3. *Let $(A, \mathcal{L})$ and $(B, \mathcal{M})$ be two polarised abelian varieties. Let $\mathcal{L} \star \mathcal{M}$ denote the line bundle $p_1^*\mathcal{L} \otimes p_2^*\mathcal{M}$ on $A \times B$, where p_i are the projections. Then $G(\mathcal{L} \star \mathcal{M}) \simeq G(\mathcal{L}) \times G(\mathcal{M})/\mathbb{G}_m$ (where $\mathbb{G}_m$ acts via $\alpha \mapsto (\alpha, \alpha^{-1})$), and if we fix a theta structure $\Theta_{\mathcal{L}}$ and $\Theta_{\mathcal{M}}$ respectively, the induced product theta structure $\Theta_{\mathcal{L}\star\mathcal{M}}$ satisfy*

$$(\theta_i^{\mathcal{L}\star\mathcal{M}})_{i \in K_1(\mathcal{L}) \times K_1(\mathcal{M})} = \theta^{\mathcal{L}}_{p_1(i)} \theta^{\mathcal{M}}_{p_2(i)}.$$

Proof. This is [Mum66, Lemma 3 p. 323]. □

2.5 DESCENT THEORY AND MUMFORD'S ISOGENY FORMULA

2.5.1 *Descent theory*

Descent theory will mainly be useful to construct isogeny algorithms, but we also need it here because the arithmetic relations on the theta functions are derived by Mumford in [Mum66] from the isogeny theorem.

Let $f : A \to B$ be an étale isogeny, and let $K = \mathrm{Ker} f$ be its kernel (which is separable since f is étale). Descent theory tells us when the line bundle $\mathcal{L}$ on A is the pullback of a line bundle $\mathcal{M}$ on B, and also how their sections relate. This is a special case of Grothendieck's fpqc descent of quasi-coherent sheaves.

If we fix an isomorphism $\psi : f^*\mathcal{M} \simeq \mathcal{L}$, then clearly if $P \in K(\overline{k})$, not only $t_P^*\mathcal{L} \simeq \mathcal{L}$, so $P \in K(\mathcal{L})$, but ψ induces a canonical lift $g_P \in G(\mathcal{L})$ of P (which does not depend on the choice of ψ). Moreover, if $s \in \Gamma(B, \mathcal{M})$ is a section, then f^*s is clearly a section of $\mathcal{L}$ invariant by $\widetilde{K}$. Descent theory tells us that both converse are true.

Theorem 2.5.1. *Let $f : A \to B$ be an isogeny, and $\mathcal{L}$ a line bundle on A. Then the line bundle $\mathcal{L}$ descends to a line bundle $\mathcal{M}$ on B if and only if $K = \operatorname{Ker} f$ is a subgroup of $K(\mathcal{L})$ which lift (as a group) into $G(\mathcal{L})$.*

Let $\rho : G(\mathcal{L}) \to K(\mathcal{L})$ denote the projection. Then the following conditions are equivalent:

- *K admits a lift $\widetilde{K}$;*
- *K is isotropic for $e_{\mathcal{L}}$;*
- *$\rho^{-1}(K)$ is a commutative group.*

If these conditions are satisfied, then the following data are equivalent:

- *the choice of a lift $\widetilde{K}$ of K (over $\bar{k}$);*
- *a choice of descent $\mathcal{M}$ of $\mathcal{L}$ on B;*
- *a character χ' on $\rho^{-1}(K)$, which is the identity on $\mathbb{G}_m$ (via $\widetilde{K} = \ker \chi'$).*

Then the choice of lifts form a torsor under the action of the characters $\chi \in K^\vee$ on χ' (hence on the equivalent data $\widetilde{K}$ and $\mathcal{M}$). Since $e_{\mathcal{L}}$ is non degenerate, χ is of the form $\chi_P := e_{\mathcal{L}}(P, \cdot)$, for a $P \in K(\mathcal{L})$ well defined modulo $K^\perp$, ie $K^\vee \simeq K(\mathcal{L})/K^\perp$. The corresponding descent of $\mathcal{L}$ is then $\mathcal{M}_{\chi_P} := t_{f(P)}\mathcal{M}$.

Assume now that f if of degree prime to p, and fix a lift $\widetilde{K}$ corresponding to $\mathcal{M}$. Then the action of $\widetilde{K}$ on $\Gamma(A, \mathcal{L})$ is semi-simple, so we have a decomposition of the global sections in terms of the eigenvalues:

$$\Gamma(A, \mathcal{L}) = \oplus_{\chi \in K^\vee} \Gamma(A, \mathcal{L})^\chi.$$

In particular, for $\chi = \mathrm{Id}$, we get $\Gamma(A, \mathcal{L})^{\widetilde{K}} = \Gamma(B, \mathcal{M})$, via $s \in \Gamma(B, \mathcal{M}) \mapsto f^*s = s \circ f$. More generally, if $\chi = \chi_P = e_{\mathcal{L}}(P, \cdot)$, we have $\Gamma(A, \mathcal{L})^{\chi_P} = \Gamma(B, \mathcal{M}_{\chi_P})$.

Fixing lifts $g_P \in G(\mathcal{L})(\bar{k})$ of a set of representative of $K(\mathcal{L})(\bar{k})/K^\perp$, then each g_P induces an isomorphism $\Gamma(A, \mathcal{L})^{\widetilde{K}} \to \Gamma(A, \mathcal{L})^{\chi_P}$. Hence a section $s \in \Gamma(A, \mathcal{L})$ decomposes uniquely as $s = \sum_{P \in K(\mathcal{L})/K^\perp} g_P f^*(s_P)$, for sections $s_P \in \Gamma(B, \mathcal{M})$.

Proof. This is proved in [Mum66] and summarized in [Rob10, § 3.3, § 3.4]. The last part is [Kem89a, §5].

Here, to prove that if K is isotropic there is always a lift $\widetilde{K}$, it is important that the polarisation is separable. Indeed, if K is isotropic for $e_{\mathcal{L}}$, then lifting K amount to finding a section of the projection $\pi^{-1}(K) \subset G(\mathcal{L}) \to K$ where $\pi : G(\mathcal{L}) \to K(\mathcal{L})$ is surjective with kernel $\mathbb{G}_m$ and $\pi^{-1}(K)$ is abelian by isotropy of K. Since the polarisation is separable, the degree of K is prime to p, hence a lift exists by general theory. A word of caution: even if K is rational its lift may not be. We will come back to this in Section 2.11. □

Remark 2.5.2. More generally, for an isogeny $f : A \to B$, descent theory gives an equivalence of category between descent line bundles of B and descent data of line bundles on A. In particular, if $\mathcal{L}$ descends to $\mathcal{M}$, an isomorphism $\psi : \mathcal{L} \to t_x^*\mathcal{L}$ descends to an isomorphism $\mathcal{M} \to t_{f(x)}^*\mathcal{M}$ if and only if it commutes with the level subgroup $\widetilde{K}$. We deduce that $G(\mathcal{M}) \simeq \mathcal{Z}(\widetilde{K})/\widetilde{K}$.

Recipe 2.5.3. If a theta structure is fixed on $G(\mathcal{L})$, with the corresponding lifts $\widetilde{K}(\mathcal{L})_1, \widetilde{K}(\mathcal{L})_2$ of the symplectic decomposition, we recover the theta basis as follow: θ_0 is a section $\theta_0 \in \Gamma(A, \mathcal{L})^{\widetilde{K}(\mathcal{L})_2}$ (which is of dimension 1), and θ_i for $i \in K(\mathcal{L})_1$ is $g_i \cdot \theta_0$ where $g_i \in \widetilde{K}(\mathcal{L})_1$ is the unique lift above i.

This gives the following recipe to construct the theta basis on a general model, provided we can compute sections of $\Gamma(\mathcal{L})$ and know how to evaluate the action of $\mathbf{G}(\mathcal{L})$. First take a section $s \in \Gamma(\mathcal{L})$, and take its trace under $\widetilde{K}(\mathcal{L})_2$. If this trace is non zero, this is our θ_0. Then the action of $\widetilde{K}(\mathcal{L})_1$ gives the θ_i.

As a corollary of Theorem 2.10.12, if $s \in \Gamma(\mathcal{L})$ is any non zero section, then the $g_P \cdot s$ for $P \in K(\mathcal{L})$ (any choice of g_P above P) generates $\Gamma(\mathcal{L})$. And the $g_P \cdot s$, for $P \in K_1(\mathcal{L})$ form a basis for a generic s; this is also the case when s is invariant by $\widetilde{K_2(\mathcal{L})}$.

More generally Theorem 2.5.1 gives the following recipe to construct a basis of sections, given a level subgroup $\widetilde{K}$. Take a basis of sections of $\Gamma(A, \mathcal{L})^{\widetilde{K}}$, either taking under $\widetilde{K}$ traces of sections in $\Gamma(A, \mathcal{L})$, or by computing a basis of $\Gamma(B, \mathcal{M})$, where $\mathcal{M}$ is the descent of $\mathcal{L}$ to $B = A/K$ via $\widetilde{K}$ and then taking their pullback via $A \to B$. Then a basis is given by taking the action of lifts of a set of representatives of $K(\mathcal{L})/K^\perp$.

2.5.2 The isogeny formula for θ-functions

From the description of theta functions in Recipe 2.5.3, it is easy to derive an isogeny formula for an isogeny $f : (A, \mathcal{L}) \to (B, \mathcal{M})$ provided we have two compatible theta structures for $\mathcal{M}$ and $\mathcal{L}$. Since compatibility of theta structure is a key concept, it is worthwhile to decompose it in several steps, this will allow us to simplify checking compatibility (see Section 2.6).

Definition 2.5.4 (Compatibility definitions)**.** If $f : A \to B$ is an isogeny such that $\mathcal{L} = f^*\mathcal{M}$, and we have a theta structure $\Theta_{\mathcal{L}}$ on A, there are several natural compatibility conditions [Rob10, § 3.6].

- The kernel $K = \operatorname{Ker} f$ is compatible with $\Theta_{\mathcal{L}}$ (or simply with the symplectic decomposition) if $K = K_1 \oplus K_2$ with $K_i \subset K(\mathcal{L})_i$, $i = 1, 2$.
- In this case $K^\perp$ also decomposes along the $K(\mathcal{L})_i$. Since $K(\mathcal{M}) \simeq K^\perp/K$, the symplectic decomposition $K(\mathcal{L}) = K(\mathcal{L})_1 \oplus K(\mathcal{L})_2$ induces a unique compatible symplectic decomposition $K(\mathcal{M}) = K(\mathcal{M})_1 \oplus K(\mathcal{M})_2$ (equivalently we may define it as $K(\mathcal{M})_i = f(K(\mathcal{L})_i) \cap K(\mathcal{M})$). We say that the symplectic decomposition of $K(\mathcal{M})$ is compatible with the symplectic decomposition of $K(\mathcal{L})$.
- The decomposition of K above along with the lifts $\widetilde{K}(\mathcal{L})_i$ induce a canonical lift $\widetilde{K} = \widetilde{K}_1 \oplus \widetilde{K}_2$ of K (since K is isotropic this is indeed a group). Then $\mathcal{M}$ is compatible with $\Theta_{\mathcal{L}}$ if it is the descent of $\mathcal{L}$ with respect to this $\widetilde{K}$. In this case we say that $\mathcal{M}$ (or $\widetilde{K}$) is compatible with $\Theta_{\mathcal{L}}$.

  We may assume this is the case up to changing $\mathcal{M}$ in its algebraic equivalence class. Or if we want to keep $\widetilde{K}$, then we can always change the theta structure and extend $\widetilde{K}_i$ to lifts of $\widetilde{K}(\mathcal{L})_i$ for $i = 1, 2$ by Lemma 2.5.5.
- Likewise, the decomposition of $K^\perp$ above induces a canonical lift of $K_1^\perp$ and of $K_2^\perp$. (Alternatively, we have $Z(K)$, the centralizer of any lift of K in $G(\mathcal{L})$ is equal to $\pi^{-1}(K^\perp)$, and we take the lift $\widetilde{K}_i^\perp$ as $Z(K) \cap \widetilde{K}(\mathcal{L})_i$.) We have $G(\mathcal{M}) \simeq Z(\widetilde{K})/\widetilde{K}$, hence there is a unique level structure on $G(\mathcal{M})$ compatible with the one on $G(\mathcal{L})$. In this case we say that the level structure of $\mathcal{M}$ is compatible with the level structure of $\mathcal{L}$.
- Once this compatible level structure is fixed, the theta structures on $G(\mathcal{M})$ is then simply given by a numerotation of $K_1^\perp/K_1$.

It is easy to check that if $\Theta_{\mathcal{L}}$ is compatible with $\Theta_{\mathcal{M}}$, then $\operatorname{conj}_c \Theta_{\mathcal{L}}$ is compatible with $\operatorname{conj}_{f(c)} \Theta_{\mathcal{M}}$.

Lemma 2.5.5. *With the notations above, if $\Theta_{\mathcal{M}}$ is a theta structure on $\mathcal{M}$, there is always a compatible theta structure on $\mathcal{L}$.*

Proof. It is obvious that there is a symplectic decomposition of $K(\mathcal{L})$ compatible with K and the symplectic decomposition of $K(\mathcal{M})$. The key point is to show that we can always find a $\Theta_{\mathcal{L}}$ compatible with $\widetilde{K}$ and the level structure of $\mathcal{M}$.

This result from the following fact: if $K_a \subset K_b$ are isotropic subgroups of $K(\mathcal{L})$, then a lift $\widetilde{K}_a$ extends to a lift $\widetilde{K}_b$, which follow easily from the proof of Theorem 2.5.1 □

This was a bit long to define, but once we have compatible theta structures we are rewarded by the fact that the theta functions are compatible.

Theorem 2.5.6 (Mumford's isogeny formula)**.** *Let $f : A \to B$ be an isogeny, $\mathcal{M}$ a line bundle on B, $\mathcal{L} = f^*\mathcal{M}$, and fix compatible theta structures on $G(\mathcal{M})$ and $G(\mathcal{L})$. Then, with the notations above, writing $(\theta_i^{\mathcal{M}})_{i \in K_1 \mathcal{M}}$ and $(\theta_j^{\mathcal{L}})_{j \in K_1(\mathcal{L})}$ the corresponding theta basis, we have up to a constant $\lambda \in \overline{k}^*$:*

$$f^*\theta_i^{\mathcal{M}} = \sum_{j \in K_1(\mathcal{L}) | f(j) = i} \theta_j^{\mathcal{L}} = \sum_{j \in K_1} \theta_{i_0 + j}^{\mathcal{L}},$$

where i_0 is any preimage of i by f.

Proof. This is an elementary computation, proved in [Mum66]. See [Rob10, Théorème 3.6.4] for an overview of the proof.

We may recover this Theorem by applying our recipe from Recipe 2.5.3. The function $\theta_0^{\mathcal{M}}$ is a section invariant by $\widetilde{K}_2(\mathcal{M})$. Its pullback $f^*\theta_0^{\mathcal{M}}$ is also invariant by $\widetilde{K}$. So it suffice to find a non zero trace under $\widetilde{K} + Z(\widetilde{K})_2$. But if apply the trace to $\theta_0^{\mathcal{L}}$ we get $\sum_{j \in K_1} \theta_j^{\mathcal{L}}$ since it is invariant by $\widetilde{K}(\mathcal{L})_2$. To get $f^*\theta_i^{\mathcal{M}}$ for i in $K_1(\mathcal{M})$, by compatibility

of the theta structure we need to apply g_{i_0}, the canonical lift above i_0, to $\sum_{j\in K_1}\theta^{\mathcal{L}}_j$, for any $i_0 \in K_1(\mathcal{L})$ such that $f(i_0) = i$. We get $f^*\theta^{\mathcal{M}}_i = \sum_{j\in K_1}\theta^{\mathcal{L}}_{i_0+j}$.

Of course if $K \subset K_2(\mathcal{L})$, it is immediately obvious that $\theta^{\mathcal{L}}_0$ descends to B and is invariant by $K_2(\mathcal{M})$, and that if $i_0 \in K_1(\mathcal{L})$ is the unique point above $i \in K_1(\mathcal{M})$, $\theta^{\mathcal{L}}_{i_0}$ descends to $\theta^{\mathcal{M}}_i$ (alternatively, that the action of the lift of i on $\theta^{\mathcal{M}}_0$ is the same as the action of the lift of i_0 on $\theta^{\mathcal{L}}_0$, by definition of compatible theta structures). By applying the transposition matrix S we also get the formula for $K \subset K_1(\mathcal{L})$, and the general case $K = K_1 \times K_2$ follows from these two cases. □

A key point is that λ does not depend on the points of A and B where the theta functions are evaluated. In other words Theorem 2.5.6 can be interpreted (up to this constant, that we will usually normalize to 1) as an *affine* isogeny formula, where we see the θ_i not only as projective coordinates on $\mathbb{P}(V)$ (where $V = \Gamma(A, \mathcal{L})$), but as affine coordinates on the affine cone $\mathbb{A}(V) \setminus 0$ above it. We can reformulate this as follow: if $f : (A, \mathcal{L}) \to (B, \mathcal{M})$ is an isogeny, $\mathcal{M}$ is induced by the choice of $\widetilde{K}$, and we have an isomorphism between $\mathcal{L}$ and $f^*\mathcal{M}$. The group $\mathbb{G}_m$ act on these isomorphisms, and we may rigidify things by choosing a rigidificiation of $\mathcal{L}$ at 0_A and $\mathcal{M}$ at 0_B and taking $\tilde{f}$ to be compatible with these rigidifications, via $\tilde{f}(\theta_i(\tilde{0}_A)) = \theta_i(\tilde{0}_B)$.

Example 2.5.7. Let $(A, \mathcal{L})$ be an abelian variety, $\mathcal{L}$ a polarisation of level n, and assume we have a symmetric theta structure on $\mathcal{L}^\ell$. Let $A[\ell] = A_1[\ell] \oplus A_2[\ell]$ be the symplectic decomposition induced by this theta structure, and let $B = A/A_2[\ell]$, $C = A/A_1[\ell]$, $\pi_1 : A \to B$, $\pi_2 : A \to C$. Then if we let $\mathcal{L}$ descend to $\mathcal{M}$ and $\mathcal{N}$ on B and C respectively as induced by the theta structure on $\mathcal{L}^\ell$, and fix a compatible theta structure on $\mathcal{M}$ and $\mathcal{N}$, we get that (fixing constants equal to one) $\theta^{\mathcal{M}}_i = \theta^{\mathcal{L}}_i$ for $i \in Z(\overline{n}) \subset Z(\overline{\ell n})$ and $\theta^{\mathcal{N}}_i = \sum_{j\in Z(\overline{\ell n})|\sigma(j)=i\in Z(\overline{n})}\theta^{\mathcal{L}}_j$ for $i \in Z(\overline{n})$ and $\sigma : Z(\overline{\ell n}) \to Z(\overline{n})$ the natural quotient map.

We say that the first isogeny is of the first type, and the second isogeny of the second type. We have seen in the proof of Theorem 2.5.6 that the formula for isogenies of the second type follow from the formula for the first type combined with the action of S, and that we can combine both types to treat the general case of this Theorem.

Recipe 2.5.8. We may combine our recipe from Recipe 2.5.3 and Theorem 2.5.6 as follow. Suppose that we have a model $(A, \mathcal{L})$ (not necessary a theta model), where we know how to compute sections of $\mathcal{L}$, how to compute the action of $G(\mathcal{L})$ on these sections, and know $\widetilde{K}$. We want to compute a theta model of $(B, \mathcal{M})$, where $\mathcal{M}$ is the descent of $\mathcal{L}$ by $\widetilde{K}$, and $f : A \to B = A/K$ is the isogeny.

We let $\widetilde{K'}_2$ be an extension of $\widetilde{K}$ above a maximal isotropic subgroup $K'_2 \supset K$ in $f^{-1}K(\mathcal{M}) = K^\perp$ for the commutator pairing $e_{\mathcal{L}}$. Fix a symplectic decomposition $f^{-1}K(\mathcal{M}) = K'_1 \oplus K'_2$ and let $\widetilde{K'}_1$ be any lift in $G(\mathcal{L})$. Then $\widetilde{K'}_1, \widetilde{K'}_2$ induce a theta structure on $(B, \mathcal{M})$, and for this theta structure $f^*\theta^{\mathcal{M}}_0$ is any non zero trace of a section $s \in \Gamma(A, \mathcal{L})$ under $\widetilde{K'}_2$. The isogeny f induces a bijection between K'_1 and $K_1(\mathcal{M})$, and for $i \in K_1(\mathcal{M})$, if $i_0 \in K'_1$ is its unique preimage, $\theta^{\mathcal{M}}_i$ is given by the action of g_{i_0} the lift of i_0 in $\widetilde{K'}_1$ on $f^*\theta^{\mathcal{M}}_0$ (seen in $\Gamma(A, \mathcal{L})$). Since $i_0 \in K^\perp$, g_{i_0} commutes with $\widetilde{K}$, so $g_{i_0}f^*\theta^{\mathcal{M}}_0$ is invariant under $\widetilde{K}$, so descends to $(B, \mathcal{M})$.

Anticipating Section 2.6, if we want a symmetric theta structure on $(B, \mathcal{M})$, and K is of odd order, then there is a unique symmetric lift of $\widetilde{K}$. Furthermore if $\mathcal{L} = \mathcal{L}_0^\ell$, $\mathcal{L}_0$ of level n even, ℓ prime to n and we have a symmetric theta structure on $\mathcal{L}_0$, there is an unique extension to a symmetric theta structure on $\mathcal{L}$, hence a unique symmetric theta structure on $(B, \mathcal{M})$ compatible with the symmetric theta structure on $\mathcal{L}_0$. The symmetric choices of $\widetilde{K'}_i$ (compatible with $\widetilde{K}_i(\mathcal{L}_0)$) are thus canonicals.

2.6 SYMMETRY AND SYMMETRIC THETA STRUCTURES

The descent theorem will be a key theorem for isogeny algorithms, but a drawback is that there are many possible lifts for a kernel K, hence many possible ways to descend a line bundle $\mathcal{L}$ to B.

Indeed we saw in Theorem 2.5.1 that the possible lifts form a torsor under $K(\mathcal{L})/K^\perp$. So if we take for f an ℓ-isogeny, with $\mathcal{L} = \mathcal{L}_0^\ell$ ($\mathcal{L}_0$ of level n prime to ℓ) and K maximal isotropic in $A[\ell]$, there are ℓ^g possibilities. Algorithmically, this means that the possible lifts will be described by equations of large degree, from which it will be hard to find a point (let alone a rational point). So we want to rigidify the number of choices. Mumford's key idea in [Mum66] is that a good way to rigidify choices is to impose $\mathcal{L}$ and $\mathcal{M}$ to be symmetric.

Unfortunately, this is not always possible: a symmetric line bundle may not always descend to a symmetric line bundle. So we need a bit of technicality in this section to describe the obstruction, and to carefully define symmetric theta structures. We will be rewarded by precise conditions for rationality of a symmetric theta structure

in Section 2.6 and as mentioned this will be very useful for isogenies: in the situation of an ℓ-isogeny as above when ℓ is odd there is only one possible symmetric lift!

So it will be worthwhile to analyze the situation in more detail, since symmetry will greatly simplify compatibility conditions, see Section 2.6.3.

2.6.1 *Descending symmetric line bundles*

Let $\mathcal{L}$ be a line bundle. There is always a symmetric line bundle in its equivalence class. Indee, since $[-1]^*\mathcal{L}$ is algebraically equivalent to $\mathcal{L}$ (because $\mathrm{Pic}^0(A)$ are exactly the antisymmetric line bundles by [Rob10, p. 63]), $[-1]^*\mathcal{L} = \mathcal{L} \otimes \mathcal{U}$ where $\mathcal{U} \in \widehat{A} = \mathrm{Pic}^0(A)$, so $\mathcal{L} \otimes \mathcal{U}$ where $\mathcal{U}^2 = \mathcal{U}$ is a symmetric line bundle algebraically equivalent to $\mathcal{L}$. The other symmetric ones are given by $\mathcal{L} \otimes \mathcal{T}$ where $\mathcal{T} \in \widehat{A}[2]$. If $\mathcal{L}$ is ample, these are the $t_c^*\mathcal{L}$ where $c \in [2]^{-1}K(\mathcal{L})/K(\mathcal{L})$.

The symmetry $[-1]^*\mathcal{L} \simeq \mathcal{L}$ of $\mathcal{L}$ then induces an external action γ_{-1} on $G(\mathcal{L})$. An element $g \in G(\mathcal{L})(\bar{k})$ is said to be symmetric if $\gamma_{-1} \cdot g = g^{-1}$, and a level subgroup $\widetilde{K}$ of K is said to be symmetric if all its element are. Symmetry rigidifies the choices, because if g is symmetric, then αg is symmetric for $\alpha \in \bar{k}^*$ if and only if $\alpha = \pm 1$. So there are only two symmetric lifts above each g (if we are not in characteristic two). Indeed if we take any g, $\gamma_{-1}g = \alpha g^{-1}$ for an $\alpha \in \bar{k}^*$, so βg is symmetric whenever $\beta^2 = 1/\alpha$.

Remark 2.6.1. If g is symmetric, the g^n are symmetric for all $n \in \mathbb{Z}$. More generally if g_1 and g_2 are symmetric, then $g_1 g_2$ is symmetric if and only if they commute. So the set of symmetric elements above an isotropic subgroup K is a group.

If g is symmetric above a point of order m, there are two cases: if m is even then g is of order m. If m is odd, then either g or $-g$ (the other symmetric element above the same point) is of order m, the other is of order $2m$ [Rob10, Remarque 4.2.11]. In particular there is a unique symmetric lift of $\langle x \rangle$ when x is a point of odd order in $K(\mathcal{L})$.

There is an obstruction to lifting symmetrically an isotropic subgroup K of $K(\mathcal{L})$. We assume from now on that $p \neq 2$. If $\phi : \mathcal{L} \xrightarrow{\sim} [-1]^*\mathcal{L}$ is an isomorphism, normalised by being the identity on 0_A, and $x \in A[2](\bar{k})$ is a point of 2-torsion, then ϕ induces an isomorphism $\phi(x) : \mathcal{L}(x) \xrightarrow{\sim} [-1]^*\mathcal{L}(-x) = \mathcal{L}(x)$, and we define $e_*^{\mathcal{L}}(x) \in \bar{k}^*$ as

$$\phi(x) : \mathcal{L}(x) \xrightarrow{e_*^{\mathcal{L}}(x)} \mathcal{L}(x).$$

Proposition 2.6.2. *The application $e_*^{\mathcal{L}} : A[2] \to k^*$ has value in $\{\pm 1\}$, and satisfy the following properties:*

(i) $e_*^{\mathcal{L}\otimes\mathcal{M}} = e_*^{\mathcal{L}} \times e_*^{\mathcal{M}}$ *if $\mathcal{L}$ and $\mathcal{M}$ are symmetric on A.*

(ii) If $f : A \to B$ is a morphism, and $\mathcal{M}$ a symmetric line bundle on B, then for all $x \in A[2]$ $e_*^{f^*\mathcal{M}}(x) = e_*^{\mathcal{M}}(f(x))$.

(iii) If $y \in \widehat{A}[2](\bar{k})$ correspond to a symmetric line bundle $\mathcal{L}$ on A (algebraically equivalent to zero), then $e_*^{\mathcal{L}}(x) = e_2(x, y)$ where e_2 is the Weil pairing on $A[2] \times \widehat{A}[2]$.

(iv) The form $e_*^{\mathcal{L}}$ is the quadratic from associated at the pairing $e_{\mathcal{L}^2}$ on $A[2] \times A[2]$:

$$e_*^{\mathcal{L}}(x + y) = e_*^{\mathcal{L}}(x) e_*^{\mathcal{L}}(y) e_{\mathcal{L}^2}(x, y).$$

Furthermore, if $\mathcal{L}$ is represented by a symmetric divisor D, then $e_*^{\mathcal{L}}(x) = (-1)^{m(x)-m(0_A)}$ where $m(x)$ is the multiplicity of D at x.

Proof. This is all proved by Mumford in [Mum66, §2], see [Rob10, Proposition 4.2.2]. □

By **??** 2.6.2.(i)**??** 2.6.2.(iii), we have that $e_*^{t_c^*\mathcal{L}}(x) = e_*^{\mathcal{L}}(x) e_{\mathcal{L}^2}(x, c)$, if $c \in [2]^{-1}K(\mathcal{L})$ (so that $t_c^*\mathcal{L}$ is symmetric). In particular $e_*^{\mathcal{L}}$ determines the class of the symmetric $\mathcal{L}$ in its algebraic equivalence class.

This quadratic form measures the obstruction to finding a symmetric lift of K.

Proposition 2.6.3. *Let $\mathcal{L}$ be a symmetric line bundle on A and K an isotropic subgroup of $K(\mathcal{L})$. The following are equivalent:*

i) There exists a symmetric level subgroup $\widetilde{K}$ above K.

ii) For all $x \in K[2]$, $e_*^{\mathcal{L}}(x) = 1$.

iii) If $B = X/K$ and $f : A \to B$ is the corresponding isogeny, there exists a symmetric line bundle $\mathcal{M}$ on B such that $f^*\mathcal{M} \simeq \mathcal{L}$.

Proof. Once again this is proved by Mumford in [Mum66, §2], see [Rob10, Proposition 4.2.12]. The link between $e_*^{\mathcal{L}}$ and the obstruction to lifting comes from the fact that if $x \in K(\mathcal{L})[2]$ and $\tilde{x}$ is any lift to $G(\mathcal{L})$, then $\gamma_{-1}(\tilde{x}) = e_*^{\mathcal{L}}(x)\tilde{x}$. □

If $e_*^{\mathcal{L}}(x) = 1$ for all $x \in A[2]$ we say that $\mathcal{L}$ is totally symmetric. Mumford shows that $\mathcal{L}$ is totally symmetric if and only if it is the square of a symmetric line bundle (equivalently if it descends to $A/\pm 1$, [Mum66, § 2, Proposition 1]). In this case the level is divisible by two, and $\mathcal{L}$ is the unique totally symmetric line bundle in its algebraic equivalence class (by Proposition 2.6.2 or because two symmetric line bundles differ by an element of $\widehat{A}[2]$ which is killed when taking squares). By Proposition 2.6.2, the pullback of a totally symmetric line bundle is totally symmetric.

2.6.2 *Symmetric theta structures*

There is also a canonical action of γ_{-1} on the Heisenberg group via $\gamma_{-1} \cdot (\alpha, x) = (\alpha, -x)$.

Definition 2.6.4. A symmetric theta structure is an isomorphism $\mathbb{Z}/2\mathbb{Z} \rtimes \mathcal{H}(\overline{n}) \xrightarrow{\sim} \mathbb{Z}/2\mathbb{Z} \rtimes G(\mathcal{L})$, where $1 \in \mathbb{Z}/2\mathbb{Z}$ acts by γ_{-1}.

Equivalently, this is the same thing as a theta structure $\Theta : \mathcal{H}(\overline{n}) \to G(\mathcal{L})$ which commutes with the action of γ_{-1}, or to the fact that the level subgroups $\widetilde{K}(\mathcal{L})$ are symmetric, or that Θ is $[-1]$-compatible with itself [Rob10, Proposition 4.2.9].

The theta functions induced by a symmetric line bundle have a symmetry:

$$\gamma_{-1}\theta_i = \epsilon\theta_{-i}, \tag{2.5}$$

where $\epsilon = \pm 1$ [Mum66].

By Proposition 2.6.3, if $\mathcal{L}$ is totally symmetric there is no obstruction to lifting isotropic subgroups symmetrically. In particular a symmetric theta structure always exist.

Conversely, for a general $\mathcal{L}$, a symmetric theta structure exists by Proposition 2.6.3 if and only if $e_*^{\mathcal{L}}$ is trivial on $K(\mathcal{L})_1[2]$ and $K(\mathcal{L})_2[2]$. But since $e_{\mathcal{L}^2}$ is always trivial on $A[2] \cap K(\mathcal{L})$, this is equivalent to $e_*^{\mathcal{L}}$ being trivial on $K(\mathcal{L})[2]$. Hence if n is even, a symmetric theta structure exists if and only if $e_*^{\mathcal{L}}(x) = 1$ for all $x \in A[2]$, ie if $\mathcal{L}$ is totally symmetric. By contrast if n is odd, there is no obstruction, so a symmetric theta structure always exist.

Lets say that $\mathcal{L}$ is symmetrisable if $\mathcal{L}$ is symmetric and $e_*^{\mathcal{L}}$ is trivial on $K(\mathcal{L})[2]$. It is instructive to look at what happens when we use the conjugation by $c \in K(\mathcal{L})(\bar{k})$. First if $\mathcal{L}$ is symmetrisable, the other symmetric line bundles are given by the action of $y \in \widehat{A}[2](\bar{k})$. If y corresponds to $\mathcal{P}_y$, then $\mathcal{L} \otimes \mathcal{P}_y \simeq t_c^*\mathcal{L}$ where $\Phi_{\mathcal{L}}(c) = y$ is symmetrisable if and only if y is orthogonal to $K(\mathcal{L})[2](\bar{k})$ for the Weil pairing e_2, so the set of symmetrisable line bundles is in bijection with $A[2](\bar{k})/K(\mathcal{L})[2](\bar{k})$. Conversely, once a symmetric theta structure is fixed, all other ones (inducing the identity on $K(\mathcal{L})$) are induced by the conjugation action of $K(\mathcal{L})[2](\bar{k})$ [Rob10, p. 67].

We formulated the above paragraph in such a way that it is valid for a general type of polarisation. If we go back to our usual setting of $\mathcal{L} = \mathcal{L}_{A,1}^n$, we see that there are two cases: if n is even, there is one symmetrisable line bundle in its equivalence class, the totally symmetric one. But there are 2^{2g} symmetric theta structures possible (once a symplectic decomposition of $K(\mathcal{L})$ is fixed). But if n is odd, all symmetric line bundles are symmetrisable, but each has only one symmetric theta structure on it.

Remark 2.6.5. By Remark 2.6.1, if we have a symmetric theta structure on $G(\mathcal{L})$ and ℓ is odd, then given a symplectic decomposition of $K(\mathcal{L}^\ell)$ compatible with the decomposition of $K(\mathcal{L})$, there is a unique extension to a symmetric theta structure on $G(\mathcal{L}^\ell)$. If ℓ is prime to the level n of $\mathcal{L}$, it suffices to give a symplectic decomposition of $A[\ell]$.

Remark 2.6.6 (Symmetric automorphisms)**.** The symmetric automorphisms of the Heisenberg group (hence the theta group) fit into the exact sequence

$$0 \longrightarrow K(\delta)[2] \longrightarrow \mathrm{Aut}^{sym}(H(\delta)) \longrightarrow \mathrm{Sp}(K(\delta)) \longrightarrow 0.$$

If $\delta = (n, \dots, n)$ and $2 \mid n$, analytically, the symmetric automorphisms correspond to the action of $\mathrm{Sp}_{2g}(\mathbb{Z})/\Gamma(n, 2n)$ where $\Gamma(n, 2n) \subset \mathrm{Sp}_{2g}(\mathbb{Z})$ is Igusa's level subgroup of matrices $M = \left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ such that $M \equiv \mathrm{Id} \pmod{n}$ and $\mathrm{diag}(b) = \mathrm{diag}(c) = 0 \pmod{2n}$.

Using Remark 2.4.2 we may recover the projective action of $\mathrm{Sp}_{2g}(\mathbb{Z})$ on theta functions. Let us detail this since this will be useful to study the fibers of the modular correspondance in Section 5.2.2. Since we know how to act by $\overline{c}$ for $c \in K(n)[2]$, it remains to explain how to compute the action of any symmetric lift ψ of a symplectic automorphism $\overline{\psi}$ of $K(n)$. These symplectic automorphisms are generated by:

- The matrix S, which transpose K_1 and K_2. We may lift the action by using the same level subgroups $\widetilde{K}_i(\mathcal{L})$ (just permuted), so we get that the action described in Remark 2.4.2 is symmetric.
- Matrices of the form $\left(\begin{smallmatrix} a & 0 \\ 0 & {}^t a^{-1} \end{smallmatrix}\right)$, in other word a permutation of $K_1(n)$. We may also keep the same level subgroup, and we have the same theta functions, just numeroted under the new permutation.
- Matrices of the form $\left(\begin{smallmatrix} 1 & 0 \\ c & 1 \end{smallmatrix}\right)$. In otherwords, $\overline{\psi}(i,j) = (i, \psi_0(j) + j)$ where ψ_0 has for matrix c in the symplectic basis. In this case we let ζ be a primitive $2n$-root of unity, so that we have an isomorphism $\alpha : Z(\overline{\ell n}) \to \hat{Z}(\overline{\ell n})$, in such a way that if $x, y \in Z(\overline{\ell n})$, $\langle x, \alpha(y)\rangle = \zeta^{2\sum_{i=1}^g x_i y_i}$. We may define a symmetric lift via $\psi(\alpha, i, j) = (\alpha^{\frac{\langle i, \psi_0(i)\rangle}{2}}, i, \psi_0(i) + j)$, where $\frac{\langle x, f(y)\rangle}{2} := \zeta^{\sum_{i=1}^g x_i y_i}$. The action is given by [Rob10, p. 141] as $\psi.\theta_i = \frac{\langle i, -\psi_0(i)\rangle}{2}\theta_i$.

Of course this is only one part of the content of the full theta modular equations [Mum83; BL04], whose affine constant determines the full behaviour of the theta constants as modular functions. We refer to [Can20] for an exciting approach to the functional equation of theta functions from an algebraic point of view.

2.6.3 Symmetry and isogenies

By compatibility of $e_*^{\mathcal{L}}$ with isogenies, if $f : A \to B$ is an étale isogeny of degree prime to 2, then if $\mathcal{L}$ is symmetrisable and descends to a symmetric line bundle $\mathcal{M}$ on $B = A/K$ (ie we have a symmetric lift $\widetilde{K}$) then $\mathcal{M}$ is symmetrisable. Furthermore $Z(K) \subset G(\mathcal{L}) \twoheadrightarrow G(\mathcal{M})$ sends symmetric elements into symmetric elements. So if $\Theta_{\mathcal{L}}$ is a symmetric theta structure and $\Theta_{\mathcal{M}}$ is compatible to it, it is symmetric (without conditions on the degree of f).

Lemma 2.6.7. *In the case of symmetric theta structures, compatibility is easy to check:*

- *If K is compatible with the symplectic decomposition of $K(\mathcal{L})$, ie $K = K_1 \oplus K_2$, and $K \subset K(\mathcal{L})[2]^\perp$ (or equivalently $K(\mathcal{L})[2] \subset K^\perp$; we also remark that we always have $2K(\mathcal{L}) \subset K(\mathcal{L})[2]^\perp$) then any symmetric theta structure on $G(\mathcal{L})$ induces the same symmetric $\widetilde{K}$, hence the same $\mathcal{M}$.*
- *If furthermore $K^\perp \subset K(\mathcal{L})[2]^\perp$ (or equivalently $K(\mathcal{L})[2] \subset K$), then every symmetric theta structure on $G(\mathcal{L})$ induce the same symmetric theta structure on $G(\mathcal{M})$.*

Proof. Indeed, changing the symmetric theta structure correspond to acting by conjugation by $c \in K(\mathcal{L})[2]$, so leave $\widetilde{K}$ invariant if $K \subset K(\mathcal{L})[2]^\perp$. The same reasoning holds for $K^\perp$. □

Remark 2.6.8. One needs to be careful that even if K is compatible with the symplectic decomposition of $K(\mathcal{L})$, there may not be a choice of symmetric theta structure compatible with a given symmetric $\widetilde{K}$ (ie extending $\widetilde{K}$ to $\widetilde{K}(\mathcal{L})_i$ may involve non symmetric elements), as the above Lemma shows. In other words one may need to change $\mathcal{M}$ by an algebraically equivalent line bundle. However, if $\mathcal{M}$ is totally symmetric, the corresponding $\widetilde{K}$ can always be extended to a symmetric theta structure. Indeed take any symmetric theta structure on $\mathcal{L}$, this gives an $\mathcal{M}'$ which is symmetrisable. But $\mathcal{M}$ is the only symmetrisable line bundle in its equivalence class, so is equal to $\mathcal{M}'$. Since a totally symmetric line bundle $\mathcal{M}$ is of the form $\mathcal{M}_0^2$ with $\mathcal{M}_0$ symmetric, we have that $B[2] \subset K(\mathcal{M})$, hence $A[2] \subset K^\perp$, so this is coherent with the fact that every symmetric theta structure induce the same $\widetilde{K}$ by Lemma 2.6.7, which is thus the only one corresponding to the totally symmetric $\mathcal{M}$. In other words: if $B[2] \subset K(\mathcal{M})$, the only possible symmetric descent of $\mathcal{L}$ induced by a symmetric theta structure is the only totally symmetric line bundle in the equivalence class of $\mathcal{M}$.

When $\mathcal{L}$ is symmetric, Mumford introduces in [Mum66, § 2] maps $\eta_2 : G(\mathcal{L}^2) \to G(\mathcal{L})$ and $\epsilon_2 : G(\mathcal{L}) \to G(\mathcal{L}^2)$, and corresponding morphisms on the Heisenberg groups H_n and H_{2n} (see **??** 2.8.7.(vi)). A compatible theta structures for $(\mathcal{L}, \mathcal{L}^2)$ [Mum66, p. 317] is a pair of theta structures that is compatible with η_2 and ϵ_2; the theta structure on $\mathcal{L}^2$ is then automatically symmetric since $\epsilon_2 \circ \eta_2 = \delta_2$ where $\delta_2(z) = z^3\gamma_{-1}(z)$. The same holds for the theta structure on $\mathcal{L}$. Furthermore if $x \in A[2]$, and $z \in G(\mathcal{L}^2)$ is of order 2 above x, $\eta_2(z) = e_*^{\mathcal{L}}(x)$ [Mum66, Proposition 6]. Mumford then shows, if $\mathcal{L}$ is totally symmetric:

- Every symmetric theta structure on $\mathcal{L}$ extends to a compatible symmetric theta structure on $(\mathcal{L}, \mathcal{L}^2)$.
- There is a bijection between symmetric theta structures on $\mathcal{L}^2$ and compatible (symmetric) theta structures on $(\mathcal{L}, \mathcal{L}^2)$.
- A symmetric theta structure on $\mathcal{L}$ is completely determined by a symplectic basis of $K(\mathcal{L})$ and a compatible symplectic basis of $K(\mathcal{L}^2)$. And conversely any symplectic basis of $K(\mathcal{L}^2)$ corresponds to one symmetric theta structure on $K(\mathcal{L})$.

  But beware that two different symplectic basis of $K(\mathcal{L}^2)$ above the one of $K(\mathcal{L})$ may lead to the same symmetric theta structure on $K(\mathcal{L})$, we will see below in Lemma 2.11.1 and Corollary 2.11.2 that the symmetric theta structure on $K(\mathcal{L})$ only depends on the symplectic decomposition of $K(\mathcal{L}^2)$ (and even less than that).

Mumford then shows how to derive the duplication formula from a pair of compatible theta structures for $(\mathcal{L}, \mathcal{L}^2)$ (see Corollary 2.7.2).

With this we can complement Lemma 2.6.7 as follow:

Lemma 2.6.9. *Let $f : (A, \mathcal{L}) \to (B, \mathcal{M})$ be an isogeny with $\mathcal{M}$ totally symmetric, $K = \operatorname{Ker} f$, $\mathcal{L} = f^*\mathcal{M}$. Fix a symplectic decomposition of $K(\mathcal{L})$ compatible with K. Then $\mathcal{L}$ is totally symmetric, and all symmetric theta structure on $(A, \mathcal{L})$ are compatible with $\mathcal{M}$. Let $K(\mathcal{L}^2) = K_1(\mathcal{L}^2) \oplus K_2(\mathcal{L}^2)$ be a symplectic decomposition inducing a given symmetric theta structure on $\mathcal{L}$. This induces a symplectic decomposition of the orthogonal of K in $K(\mathcal{L}^2)$, hence via the isogeny a symplectic decomposition of $K(\mathcal{M}^2)$. This induces the symmetric theta structure on $\mathcal{M}$ given by the descent of the one on $\mathcal{L}$.*

This reduces checking compatibility to checking compatibility of symplectic decompositions.

Since we want a generalised addition formula and we do not want to always restrict to totally symmetric line bundles $\mathcal{L}$, we now follow Kempf [Kem89a]. We want a compatibility for all multiplications by $[n]$. Compatibility with $[-1]$ means we want a symmetric theta structure. For compatibility with $[2]$ we need a $[2]$-compatible theta structure on $\mathcal{L}^4$ and $\mathcal{L}$. Fix a symplectic decomposition of $K(\mathcal{L}^4)$. Then by Lemma 2.6.7 all symmetric theta structures (compatible with this decomposition) on $\mathcal{L}^4$ induce the same $[2]$-descent $\mathcal{L}'$ and the same symmetric theta structure on $\mathcal{L}'$. So up to changing $\mathcal{L}$ in its equivalence class, there exists a (unique) symmetric theta structure compatible with the decomposition of $K(\mathcal{L}^4)$. By Remark 2.6.8, if $A[2] \subset K(\mathcal{L})$, then $\mathcal{L}'$ is the only totally symmetric line bundle in the equivalence class of $\mathcal{L}$.

Once we have replaced $\mathcal{L}$ by $\mathcal{L}'$ so that we have $[2]$-compatibility, we automatically are $[n]$-compatible for any n:

Lemma 2.6.10. *Fix a compatible symplectic decompositions for all $K(\mathcal{L}^n)$ (n prime to p). We will call this an ∞-decomposition, they can be constructed by choosing symplectic decompositions on all $T_\ell A$, ℓ prime to p.*

Let $\mathcal{L}'$ be the unique symmetric line bundle $[2]$-compatible with the symplectic decomposition of $K(\mathcal{L}^4)$ (so $\mathcal{L}' = \mathcal{L}$ if $\mathcal{L}$ is totally symmetric). Then the symplectic decompositions of all $K(\mathcal{L}^n)$ induce a unique symmetric theta structure on all $\mathcal{L}'^n$. These symmetric theta structures are compatible with all multiplications by $[n]$.

Proof. This is [Kem89a]. Essentially once we have normalized the duplication $[2] : (A, \mathcal{L}^4) \to (A, \mathcal{L})$ (changing $\mathcal{L}$ if needed), this normalizes the rest. Indeed, a symplectic decomposition of $A[\mathcal{L}^4]$ induces a symmetric theta structure on $\mathcal{L}^2$, and one on $\mathcal{L}$ (again here we may have to change $\mathcal{L}$ in its equivalence class if it is not totally symmetric). Since $\mathcal{L}^2$ is totally symmetric, we have compatibility for all other $[n]$-isogenies, n-even by Lemma 2.6.9. And for n odd there is no problem, since the obstructions come from the 2-torsion. □

In the case where $\mathcal{L}$ is totally symmetric, any compatible theta structure on $(\mathcal{L}, \mathcal{L}^2)$ extends to an ∞-decomposition (since they are induced by a symplectic decomposition of $K(\mathcal{L}^4)$), so Kempf's notion is a generalisation of Mumford's.

From now on when we fix an ∞-decomposition on $\mathcal{L}$, we implicitly change $\mathcal{L}$ to $\mathcal{L}'$ if necessary. If $f : A \to B$ is an isogeny, whose kernel is compatible with the symplectic decompositions of $K(\mathcal{L}^\ell)$ for some ℓ, the symplectic decomposition of $K(\mathcal{L}^\ell)$ induces a unique symmetric lift $\widetilde{K}$, hence a unique descent $\mathcal{M}$ to B, and descending the symplectic decompositions of $\mathcal{L}^{n\ell}$ via f induce a symplectic decomposition of all $\mathcal{M}^n$, hence symmetric theta structures on all $\mathcal{M}^n$, which are compatible with the symmetric theta structures on $\mathcal{L}^{n\ell}$.

The ∞-decomposition on A also induces one on all A^r, and using Lemma 2.4.3 the product theta structures are all compatible. Imposing the compatibility conditions also determines the sign of ϵ in Equation (2.5): $\gamma_{-1}\theta_i = \theta_{-i}$ [Kem89a, Theorem 11].

2.7 ADDITION FORMULA AND EQUATIONS FOR ABELIAN VARIETIES

Now that the technicalities are over, we can finally get concrete equations.

In this section, we let $(A, \mathcal{L})$ be a polarised abelian variety, and we fix once and for all an ∞-decomposition on $\mathcal{L}$. We recall that to do that we may need to replace $\mathcal{L}$ by an equivalent, and symmetric, line bundle (which will be the totally symmetric one if the level is divisible by two).

Theorem 2.7.1 (The Koizumi-Kemp addition formula). *Let $f : A^r \to A^r$ be an isogeny given by an integral $r \times r$ matrix F such that*

$$ {}^tF \begin{pmatrix} m_1 & & 0 \\ & \ddots & \\ 0 & & m_r \end{pmatrix} F = \begin{pmatrix} \ell_1 & & 0 \\ & \ddots & \\ 0 & & \ell_r \end{pmatrix} \tag{2.6} $$

where the m_i and ℓ_i are integers prime to p.

If π_i is the projection of A^r to its i-th component, let $\mathcal{L}' = \mathcal{L}^{\ell_1} \star \cdots \star \mathcal{L}^{\ell^r} := \pi_1^*\mathcal{L}^{\ell_1} \otimes \cdots \otimes \pi_r^*\mathcal{L}^{\ell_r}$ and $\mathcal{L}'' = \mathcal{L}^{m_1} \star \cdots \otimes \mathcal{L}^{m_r} = \pi_1^*\mathcal{L}^{m_1} \otimes \cdots \otimes \pi_r^*\mathcal{L}^{m_r}$. By Equation (2.6), $f^*\mathcal{L}'' \simeq \mathcal{L}'$.

Then there is a constant $\lambda \in k^*$ such that for all $(i_1, \dots, i_r) \in K_1(\mathcal{L}^{m_1}) \times \cdots \times K_1(\mathcal{L}^{m_r})$,

$$ f^*(\theta_{i_1}^{\mathcal{L}^{m_1}} \star \cdots \star \theta_{i_r}^{\mathcal{L}^{m_r}}) = \lambda \sum_{\substack{(j_1,\dots,j_r) \in K_1(\mathcal{L}^{m_1}) \times \cdots \times K_1(\mathcal{L}^{m_r}) \\ f(j_1,\dots,j_r) = (i_1,\dots,i_r)}} \theta_{j_1}^{\mathcal{L}^{\ell_1}} \star \cdots \star \theta_{j_r}^{\mathcal{L}^{\ell_r}} \tag{2.7} $$

Proof. The ∞-decomposition induces symmetric theta structures which are automatically compatible for any such F by the discussion at the end of Section 2.6.3. Then Equation (2.7) is just an application of the isogeny formula along with the Segre embedding of Lemma 2.4.3. □

As remarked in Theorem 2.5.6, Theorem 2.7.1 also makes sense when interpreting the θ_i as *affine* coordinates. This will be crucial for a lot of our algorithms, and we will reinterpret this meaning in Sections 2.8.3 and 2.9. To distinguish between projective points and affine points, I will typically denote by x a projective point and $\tilde{x}$ an affine point above it.

Corollary 2.7.2 (Duplication formula). *Let $\xi : A^2 \to A^2, (P, Q) \mapsto (P + Q, P - Q)$. Then $\xi^*\mathcal{L} \star \mathcal{L} \simeq \mathcal{L}^2 \star \mathcal{L}^2$, and if p is prime to 2, there exists $\lambda \in k^*$ such that for all $(i_1, i_2) \in K_1(\mathcal{L})^2$,*

$$ \xi^*(\theta_{i_1}^{\mathcal{L}} \star \theta_{i_2}^{\mathcal{L}}) = \lambda \sum_{\substack{(j_1,j_2) \in K_1(\mathcal{L}^2)^2 \\ j_1 + j_2 = i_1 \\ j_1 - j_2 = i_2}} \theta_{j_1}^{\mathcal{L}^2} \star \theta_{j_r}^{\mathcal{L}^2} $$

Assume now that the level n is divisible by 2. We fix an isomorphism $Z(\bar{2}) \simeq K_1(\mathcal{L})[2]$, and define the change of variable (a partial Fourier transform),

$$ U_{\chi,i}^{\mathcal{L}} = \sum_{t \in Z(\bar{2})} \chi(t) \theta_{i+t}^{\mathcal{L}}, $$

for $\chi \in \hat{Z}(\bar{2})$ and $i \in K_1(\mathcal{L})$. Then there exists $\lambda_1, \lambda_2 \in k^*$ such that

$$ \theta_{i+j}^{\mathcal{L}}(\widetilde{x+y}) \theta_{i-j}^{\mathcal{L}}(\widetilde{x-y}) = \lambda_1 \sum_{\substack{u,v \in K_1(\mathcal{L}^2) \\ u+v=i \\ u-v=j}} \theta_u^{\mathcal{L}^2}(\tilde{x}) \theta_v^{\mathcal{L}^2}(\tilde{y}) = \frac{\lambda_1}{2^g} \sum_{\chi \in \hat{Z}(\bar{2})} U_{\chi,i}^{\mathcal{L}^2}(\tilde{x}) U_{\chi,j}^{\mathcal{L}^2}(\tilde{y}) \tag{2.8} $$

$$ U_{\chi,i}^{\mathcal{L}^2}(\tilde{x}) U_{\chi,j}^{\mathcal{L}^2}(\tilde{y}) = \lambda_2 \sum_{t \in \hat{Z}(\bar{2})} \chi(t) \theta_{i+j+t}^{\mathcal{L}}(\widetilde{x+y}) \theta_{i-j+t}^{\mathcal{L}}(\widetilde{x-y}) \tag{2.9} $$

Proof. The first equation comes from Theorem 2.7.1 applied to the matrix $F = \left(\begin{smallmatrix} 1 & 1 \\ 1 & -1 \end{smallmatrix}\right)$, $m_1 = m_2 = 1$ and $\ell_1 = \ell_2 = 2$ and the second is just a change of variable. See [Mum66], [Rob10, Théorème 4.4.3]. □

Corollary 2.7.2 is the reason why working with totally symmetric line bundles (hence level divisible by two) is so convenient. Starting from now we will essentially assume that we have a symmetric theta structure on a totally symmetric line bundle (I will refer to this as a "theta model"). But see also Section 2.9 for an explanation on how some of the tools we develop in the rest of this Chapter could be extended to other models.

Applying Corollary 2.7.2 twice yields

Theorem 2.7.3 (Riemann relations). *Assume that the level is divisible by 2. Let x_1, y_1, u_1 et v_1 be geometric points on A and $z \in A(\bar{k})$ such that $x_1 + y_1 + u_1 + v_1 = 2z$. Let $x_2 = z - x_1$, $y_2 = z - y_1$, $u_2 = z - u_1$ and $v_2 = z - v_1$. There exists affine lifts of these points, such that for all $\chi \in \hat{Z}(\bar{2})$ and $i, j, k, l, m \in K_1(\mathcal{L})$ with $i + j + k + l = 2m$, if $i' = m - i$, $j' = m - j$, $k' = m - k$ and $l' = m - l$, then*

$$(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{i+t}(\tilde{x}_1)\theta_{j+t}(\tilde{y}_1)).(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{k+t}(\tilde{u}_1)\theta_{l+t}(\tilde{v}_1)) = (\sum_{t\in Z(\bar{2})} \chi(t)\theta_{i'+t}(\tilde{x}_2)\theta_{j'+t}(\tilde{y}_2)).(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{k'+t}(\tilde{u}_2)\theta_{l'+t}(\tilde{v}_2)). \quad (2.10)$$

These points are said to satisfy Riemann relations.

In particular, this yields the differential addition formula:

$$(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{i+t}(\widetilde{x+y})\theta_{j+t}(\widetilde{x-y})).(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{k+t}(\tilde{0}_A)\theta_{l+t}(\tilde{0}_A)) = (\sum_{t\in Z(\bar{2})} \chi(t)\theta_{-i'+t}(\tilde{y})\theta_{j'+t}(\tilde{y})).(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{k'+t}(\tilde{x})\theta_{l'+t}(\tilde{x})). \quad (2.11)$$

We denote this relation by[4]*:*

$$\widetilde{x+y} := \texttt{diff\_add}(\tilde{x}, \tilde{y}, \widetilde{x-y}, \tilde{0}_A).$$

This also yield the three way addition formula:

$$(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{i+t}(\widetilde{x+y+z})\theta_{j+t}(\tilde{x})).(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{k+t}(\tilde{y})\theta_{l+t}(\tilde{z})) = (\sum_{t\in Z(\bar{2})} \chi(t)\theta_{i'+t}(\widetilde{0_A})\theta_{j'+t}(\widetilde{y+z_2})).(\sum_{t\in Z(\bar{2})} \chi(t)\theta_{k'+t}(\widetilde{x+z})\theta_{l'+t}(\widetilde{x+y})). \quad (2.12)$$

We denote this relation by:

$$\widetilde{x+y+z} := \texttt{threeway\_add}(\tilde{x}, \tilde{y}, \tilde{z}, \widetilde{y+z}, \widetilde{x+z}, \widetilde{x+y}).$$

Proof. This is [Mum66], [Kem89a, §5], see also the summary in [Rob10, Théorème 4.4.6]: both members are equal to $U^{\mathcal{L}^2}_{\chi,i}(\tilde{x})U^{\mathcal{L}^2}_{\chi,l}(\tilde{y})U^{\mathcal{L}^2}_{\chi,k}(\tilde{u})U^{\mathcal{L}^2}_{\chi,j}(\tilde{v})$ for any x, y, u, v such that $x+y = x_1, x-y = y_1, u+v = u_1, u-v = v_1$. □

The Riemann relations are core equations for the algorithmic of abelian varieties. The reason is that they provide equations both for the abelian variety but also for the moduli space (of abelian varieties with a symmetric level n theta structure); in fact they provide the universal equation for the universal abelian scheme over this moduli space. The following theorem is so important that I give it for a general polarisation type $\delta = (\delta_1, \dots, \delta_g)$, and we let $d = \prod \delta_i$.

Theorem 2.7.4 (Mumford). *Let $V_\delta = \mathrm{Hom}_{\mathbb{Z}[d^{-1}]}(Z(\delta), \mathbb{Z}[d^{-1}])$ be the free $\mathbb{Z}[d^{-1}]$-module with basis the Dirac functions $(Q_i)_{i\in Z(\delta)}$. Let $\overline{\mathcal{A}}_{g,\delta}$ be the projective subvariety of $\mathbb{P}(V_\delta)$ given by Riemann relations:*

$$(\sum_{t\in Z(\bar{2})} \chi(t)Q_{i+t}Q_{j+t}).(\sum_{t\in Z(\bar{2})} \chi(t)Q_{k+t}Q_{l+t}) = (\sum_{t\in Z(\bar{2})} \chi(t)Q_{i'+t}Q_{j'+t}).(\sum_{t\in Z(\bar{2})} \chi(t)Q_{k'+t}Q_{l'+t}), \quad (2.13)$$

(with χ, i, j, k, l like in Theorem 2.7.3) and the symmetry relations, for all $i \in Z(\delta)$:

$$Q_i = Q_{-i}.$$

Assume that δ is divisible by an even integer $n \geq 4$. Then the functor, who associates to a point $(A_R, \mathcal{L}, \Theta_{\mathcal{L}})$ of $\mathcal{A}_{g,\delta}(R)$ its corresponding theta null point in $\mathbb{P}(V_\delta)(R)$, is an open immersion of $\mathcal{A}_{g,\delta}$ in $\overline{\mathcal{A}}_{g,\delta}$. In particular, $\mathcal{A}_{g,\delta}$ is representable by a quasi-projective variety.

[4] The unicity of $\widetilde{x+y}$ is not clear yet, we will come back to it in Section 2.8.1

The universal abelian scheme over $\mathcal{A}_{g,\delta}$ is given by the equations

$$\Big(\sum_{t\in Z(\overline{2})}\chi(t)\theta_{i+t}\theta_{j+t}\Big).\Big(\sum_{t\in Z(\overline{2})}\chi(t)Q_{k+t}Q_{l+t}\Big) = \Big(\sum_{t\in Z(\overline{2})}\chi(t)\vartheta_{i'+t}\vartheta_{j'+t}\Big).\Big(\sum_{t\in Z(\overline{2})}\chi(t)Q_{k'+t}Q_{l'+t}\Big), \quad (2.14)$$

for all $\chi\in\hat{Z}(\overline{2})$, $i,j,k,l,m\in Z(\delta)$ such that $i+j+k+l=2m$, with $i'=m-i$, $j'=m-j$, $k'=m-k$ et $l'-l$.

Proof. This is proved by Mumford in [Mum67a] when $4\mid\delta$, and extended by Kempf in [Kem89a] when $n\mid\delta$ with n even and $n>4$, and finally in [Kem89b][5] when $n=4$. □

Note that by general theory the moduli stack $\mathcal{A}_{g,n}$ of principally polarised abelian varieties with a level n structure (ie an isomorphism $(\mathbb{Z}/n\mathbb{Z})^g\times\mu_n^g\to A[n]$) is smooth over $\mathbb{Z}[1/n]$, because there is no obstruction to lifting (Grothendieck-Mumford, [Oor71]).

Remark 2.7.5. When $\delta=(n,\dots,n)$, there is a conflict of notation between $\mathcal{A}_{g,n}$ between the moduli parametrizing level n structures and the one parametrizing symmetric level n theta structures.

Since a symmetric theta structure is intermediate between a level structure for $A[n]$ and one for $A[2n]$, a better notation for the latter moduli would have been $\mathcal{A}_{g,n,2n}$ but this notation is a bit unwieldy. It should be clear from the context to which moduli we refer to, and if needed we will use the notation $\mathcal{A}_{g,n,2n}$ or $\mathcal{A}_{g,\delta,2\delta}$ to refer to the moduli of symmetric theta structures (of level n or δ).

The coarse moduli space $\mathsf{A}_{g,n}$ of $\mathcal{A}_{g,n}$, which is the initial map into algebraic spaces and whose functor of points coincide for $\bar{k}$-points ie $\mathcal{A}_{g,n}(\bar{k})\simeq\mathsf{A}_{g,n}(\bar{k})$, and which always exist for a separated Deligne-Mumford stack by [KM97], is a quasi-projective scheme by [MFK94]. Of course when $n\geq 3$, there is no non trivial automorphisms so the inertia is trivial and $\mathcal{A}_{g,n}=\mathsf{A}_{g,n}$. We have a natural map $\mathcal{A}_{g,2\delta_g}\to\mathcal{A}_g^{(2\delta)}$ where the right hand side denote the moduli of abelian schemes with a polarisations $\mathcal{L}$ of type 2δ and a symplectic isomorphism $K(2\delta)\to K(\mathcal{L})$, see [Mum70b; Jon93]. Then the moduli space $\mathcal{A}_{g,\delta,2\delta}$ above is a quotient stack of $\mathcal{A}_g^{(2\delta)}$ by the group $\Gamma(\delta,2\delta)/\Gamma(2\delta)$ (and as remarked above, this is already a space when $\delta_1\geq 4$). We refer to [Rob21, Chapter 5] for more details.

The important point of Theorem 2.7.4 is that when we have a symmetric theta structure, we get completely explicit equations. In particular, Riemann relations encode everything about abelian varieties with such a structure and their moduli, hence is of primordial importance for algorithmic applications. We note that furthermore the equations from Theorem 2.7.4 are a very simple form of the addition relation. They say that a theta null point $0_A=(a_i)_{i\in Z(\delta)}$ needs to satisfy $0_A+0_A=0_A$ (or rather $0_A=\texttt{diff\_add}(0_A,0_A,0_A,0_A)$) and the symmetry is $-0_A=0_A$, while a point $x\in A$ needs to satisfy $x+0_A=x$ (or rather $x=\texttt{diff\_add}(x,0_A,x,0_A)$).

2.8 RIEMANN RELATIONS AND THE DIFFERENTIAL ADDITION

2.8.1 *Unicity of the differential addition*

Let A/k be an abelian variety with a symmetric theta structure of level $n\geq 4$ even. We now explain how the Riemann relations of Theorem 2.7.3 allow not only to compute the standard (projective) addition on A, but also an affine version of the addition law on the affine cone $\widetilde{A}$. This is a trivial but crucial application of Riemann relations, and has been at the core of many algorithmic applications: arithmetic of course [LR16], pairings [LR10; LR15a], and isogenies [LR12; CR15; DJR+17].

First, given $\tilde{x},\tilde{y},\widetilde{x-y}$, we need to explain why $\widetilde{x+y}$ is uniquely determined. In Equation (2.11), we see that we can determine $\sum_{t\in Z(\bar{2})}\chi(t)\theta_{i+t}(\widetilde{x+y})\theta_{j+t}(\widetilde{x-y})$ for all $i,j\in Z(\delta)$ and $\chi\in\hat{Z}(\overline{2})$ whenever we can find k,l such that $i+j+l+k=2m$, and $\sum_{t\in Z(\bar{2})}\chi(t)\theta^{\mathcal{L}}_{k+t}(\tilde{0}_A)\theta^{\mathcal{L}}_{l+t}(\tilde{0}_A)\neq 0$. Indeed, if this is the case, then from all $\sum_{t\in Z(\bar{2})}\chi(t)\theta_{i+t}(\widetilde{x+y})\theta_{j+t}(\widetilde{x-y})$ a simple linear change of variable gives us all $\theta_i(\widetilde{x+y})\theta_j(\widetilde{x-y}),i,j\in Z(\overline{n})$. So, since there is always a j such that $\theta_j(x-y)\neq 0$ the relations from Equation (2.11) allows us both to recover the affine addition law and the projective addition law (where in this case $\theta_j(x-y)$ is interpreted as an unknown but non zero projective factor).

[5] Unfortunately this article is almost impossible to find! I thank Sophie Morel for providing me a version.

But

$$\sum_{t\in Z(\overline{2})}\chi(t)\theta^{\mathcal{L}}_{k+t}(\tilde{0}_A)\theta^{\mathcal{L}}_{l+t}(\tilde{0}_A)=U^{\mathcal{L}^2}_u(\tilde{0}_A)U^{\mathcal{L}^2}_v(\tilde{0}_A)$$

where $u,v\in Z(2\overline{n})$ satisfy $u+v=k, u-v=l$. So we need to find $u',v'\in Z(\overline{n})$ such that $U^{\mathcal{L}^2}_{u+u'}(\tilde{0}_A)U^{\mathcal{L}^2}_{v+v'}(\tilde{0}_A)\neq 0$, so that $k'=k+u'+v', l'=l+u'-v'$ answer the question.

Mumford shows in [Mum66] that the non annulation of theta constants is related to the surjectivity of the multiplication map: let $\xi : A\times A\to A\times A$ be given on geometric points by $(x,y)\mapsto(x+y,x-y)$ (see Corollary 2.7.2). If $\Delta : A\to A\times A$ is the diagonal embedding, and $S : A\to A\times A$ is given by $x\mapsto(x,0)$, we then have a commutative diagram of polarized abelian varieties:

$$\begin{array}{ccc}(A,\mathcal{L}^2) & & \\ \downarrow S & \searrow^{\Delta} & \\ (A\times A,\mathcal{L}^2\star\mathcal{L}^2) & \xrightarrow{\xi} & (A\times A,\mathcal{L}\star\mathcal{L}).\end{array}$$

So the multiplication map $\Delta^* : \Gamma(X,\mathcal{L})\otimes\Gamma(X,\mathcal{L})\to\Gamma(X,\mathcal{L}^2)$, given in term of theta coordinates by $\theta^{\mathcal{L}}_i\star\theta^{\mathcal{L}}_j\mapsto(\theta^{\mathcal{L}}_i\otimes\theta^{\mathcal{L}}_j)$, is the compositium of $\xi^* : \Gamma(X,\mathcal{L})\otimes\Gamma(X,\mathcal{L})\to\Gamma(X,\mathcal{L}^2)\otimes\Gamma(X,\mathcal{L}^2)$ from Corollary 2.7.2 and the evaluation map $S^*(\mathcal{L}^2\star\mathcal{L}^2)\to\mathcal{L}^2$ (via a trivialisation morphism $\gamma_0 : \mathcal{L}^2(0)\to k$), which is given by $S^*\left(\Gamma(X,\mathcal{L}^2)\otimes\Gamma(X,\mathcal{L}^2)\right)\to\Gamma(X,\mathcal{L}^2)$, $\theta^{\mathcal{L}^2}_i\star\theta^{\mathcal{L}^2}_j\mapsto\theta^{\mathcal{L}^2}_i\theta^{\mathcal{L}^2}_j(0)$.

So the multiplication formula $\Gamma(X,\mathcal{L})\otimes\Gamma(X,\mathcal{L})\to\Gamma(X,\mathcal{L}^2)$ is given by

$$\theta^{\mathcal{L}}_i\star\theta^{\mathcal{L}}_j\mapsto\sum_{\substack{u,v\in Z(2\overline{n})\\ u+v=i\\ u-v=j}}\theta^{\mathcal{L}^2}_u\theta^{\mathcal{L}^2}_v(0)$$

and via the usual change of variable:

$$\sum_{t\in\hat{Z}(\overline{2})}\chi(t)\theta^{\mathcal{L}}_{i+t}\star\theta^{\mathcal{L}}_{j+t}\mapsto U^{\mathcal{L}^2}_{\chi,u}U^{\mathcal{L}^2}_{\chi,v}(0). \tag{2.15}$$

The assertion $\Gamma(A,\mathcal{L})\otimes\Gamma(A,\mathcal{L})\to\Gamma(A,\mathcal{L}^2)$ is surjective is then equivalent to the assertion that for all $u\in Z(2\overline{n}),\chi\in\hat{Z}(\overline{2})$, there exists $v\in Z(2\overline{n})$ congruent to u modulo $Z(\overline{n})$ such that $U^{\mathcal{L}^2}_{\chi,v}(0)\neq 0$.

Mumford showed the last assertion directly when $4\mid n$ in [Mum66, p. 339] and deduced the surjectivity of the multiplication map. Koizumi extended this result in [Koi76, Theorem 4.6] by showing analytically that $\Gamma(A,\mathcal{L}_0^n)\otimes\Gamma(\mathcal{L}_0^m)\to\Gamma(A,\mathcal{L}_0^{n+m})$ is surjective whenever $n\geq 2$ et $m\geq 3$, and an algebraic proof of this is given by Kempf in [Kem88]. We deduce that $\Gamma(A,\mathcal{L}_0^n)\otimes\Gamma(\mathcal{L}_0^n)\to\Gamma(A,\mathcal{L}_0^{2n})$ is surjective whenever $n\geq 4$ is even, which concludes our assertion. We refer to [Rob10, §4.4] for more details.

Remark 2.8.1.

- When the level n is divisible by 4, Mumford has a finer result: for all $\chi\in\hat{Z}(\overline{2}), i\in Z(2\overline{n})$, there exists a $j\in Z(\overline{4})$ such that $U^{\mathcal{L}^2}_{\chi,i+j}(0_A)\neq 0$ [Mum66, p.340], [Rob10, Proposition 4.6.11].
- Rather than using ξ, we can play the same game with the matrix $F=\left(\begin{smallmatrix}1 & -s\\ 1 & t\end{smallmatrix}\right)$ with satisfy ${}^tF\left(\begin{smallmatrix}m_1 & 0\\ 0 & m_2\end{smallmatrix}\right)F=\left(\begin{smallmatrix}m_1+m_2 & 0\\ 0 & s^2m_1+t^2m_2\end{smallmatrix}\right)$ whenever $s/t=m_2/m_1$. This relates the multiplication map $\Gamma(A,\mathcal{L}^{m_1})\otimes\Gamma(A,\mathcal{L}^{m_2})\to\Gamma(A,\mathcal{L}^{m_1+m_2})$ with the values of the theta constants of $(A,\mathcal{L}^{s^2m_1+t^2m_2})$ [Kem89a, §4]. The map ξ above is $s=t=1$.
- The same argument shows that under our running hypothesis that $n\geq 4$ is even, in the general Riemann relations we can always determine the member $(\overline{\sum_{t\in Z(\overline{2})}\chi(t)\theta_{i+t}(\tilde{x}_1)\theta_{j+t}(\tilde{y}_1)})$. Indeed, using the notations of the proof of Theorem 2.7.3, we need to show that we can find k,l such that $U^{\mathcal{L}^2}_{\chi,k}(\tilde{u})U^{\mathcal{L}^2}_{\chi,j}(\tilde{v})\neq 0$. This follow by the same proof as above, using that the addition formula is compatible with the translation map [Kem89a, Corollary 9] and that the multiplication map on the translated line bundles is still surjective [Kem88, Theorem 2]. In particular this holds for the three way addition.[6]

[6] Strangely I seem to have forgotten to put this statement in [Rob10]. A specific version for the three way addition under the assumption that $4\mid n$ is in [LR15a, Proposition 1].

Zooming back a bit, we see that a result by Kempf on the surjectivity of the multiplication map, proved by cohomological tools (more precisely the fact that an ample line bundle on an abelian variety has no higher cohomology), has for arithmetic consequences the non annulation of certain theta constants, and as a corollary yields an explicit algorithm for the addition law! In practice, all coordinates are generically non zero (here it is important that the level n is not two, because in this case the odd theta null $U^{\mathcal{L}^2}_{\chi,i}(\tilde{0}_A)$ are always zero, we will go back to this in Section 2.12), hence we use Algorithm 2.8.2 instead which is faster for computations.

Algorithm 2.8.2 (Differential addition). **Precomputations** For all $\chi \in \hat{Z}(\overline{2})$:

$$U^{\mathcal{L}^2}_{\chi,0}(\tilde{0}_A)^{-2} = \left(\sum_{t \in \hat{Z}(\overline{2})} \chi(t) \theta^{\mathcal{L}}_t(\tilde{0}_A) \theta^{\mathcal{L}}_t(\tilde{0}_A) \right)^{-1}.$$

Input Affine geometric lifts $\tilde{x}, \tilde{y}, \widetilde{x-y}$ of $\widetilde{A}$.
Output $\widetilde{x+y} := \texttt{diff\_add}(\tilde{x}, \tilde{y}, \widetilde{x-y}, \tilde{0}_A)$.

➔ For all $i \in Z(\overline{n})$

a. Compute for all $\chi \in \hat{Z}(\overline{2})$:

$$U^{\mathcal{L}^2}_{\chi,i}(\tilde{x}) U^{\mathcal{L}^2}_{\chi,0}(\tilde{y}) = \frac{1}{U^{\mathcal{L}^2}_{\chi,0}(\tilde{0}_A)^2} \left(\sum_{t \in \hat{Z}(\overline{2})} \chi(t) \theta^{\mathcal{L}}_{i+t}(\tilde{x})^2 \right) \left(\sum_{t \in \hat{Z}(\overline{2})} \chi(t) \theta^{\mathcal{L}}_t(\tilde{y})^2 \right).$$

b. Output

$$\theta^{\mathcal{L}}_i(\widetilde{x+y}) = \frac{1}{2^g \theta^{\mathcal{L}}_i(\widetilde{x-y})} \sum_{\chi \in \hat{Z}(\overline{2})} U^{\mathcal{L}^2}_{\chi,i}(\tilde{x}) U^{\mathcal{L}^2}_{\chi,0}(\tilde{y}).$$

We refer to [Rob10, Chapter 4] for more general algorithms for additions and multiplication, [LR15a, § 3] for a specialisation to the case $g = 1$, and [LR16] for the case $n = 2$.

2.8.2 Using the differential addition

We now explain why the differential addition is so fundamental. First, this gives a pseudo group law on the affine cone of the abelian variety, that is compatible with (affine lifts) of isogenies. We summarize the discussion of [Rob10, § 4.4, § 4.5]

Lemma 2.8.3. *Assume that $(A, \mathcal{L})$ is given a symmetric theta structure, and fix once and for all an affine lift $\tilde{0}_A$ of the theta null point of A. Let $x_1, \dots, x_n \in A$ and let $\tilde{x}_i$ denote arbitrary affine lifts. Assume that we are furthermore given affine lifts $\widetilde{x_i + x_j}$ of each $x_i + x_j$, $i \neq j$.*

Then for any $m_1, \dots m_n \in \mathbb{Z}$ one can define a canonical affine lift

$$m_1 x_1 + \widetilde{\cdots} + m_n x_n := \texttt{multi\_add}(m_1, \dots, m_n, \tilde{x}_i, \widetilde{x_i + x_j}, \tilde{0}_A)$$

above $m_1 x_1 + \cdots + m_n x_n$ via Riemann relations from Theorem 2.7.3. This is called the multiway multiplication (or multiway addition if $m_i \in \{-1, 0, 1\}$). More precisely differential additions and three way additions combined with $\texttt{opposite}(\tilde{x}_i) := (\tilde{x}_{-i})$ are enough. This point $m_1 x_1 + \widetilde{\cdots} + m_n x_n$ is unique, which means it does not depend on the choice of Riemann relations used to compute it.

Furthermore affine Riemann relations are compatible with symmetric automorphisms of the theta group and affine lifts of isogenies, so that multi_add *is too. In particular this means that if $\tilde{f} : \widetilde{A} \to \widetilde{B}$ lift an isogeny $f : A \to B$ given via the isogeny formula from Theorem 2.5.6, then $\tilde{f}(\texttt{multi\_add}(m_1, \dots, m_n, \tilde{x}_i, \widetilde{x_i + x_j}, \tilde{0}_A)) = \texttt{multi\_add}(m_1, \dots, m_n, \tilde{f}(\tilde{x}_i), \tilde{f}(\widetilde{x_i + x_j}), \tilde{f}(\tilde{0}_A)))$.*

Corollary 2.8.4. *If we fix affine lifts $\tilde{x}$, $\tilde{y}$ and $\widetilde{x+y}$, then as a special case we can define canonical affine lifts $\widetilde{nx} = \texttt{diff\_mult}(n, \tilde{x})$ and $\widetilde{nx+y} = \texttt{diff\_multadd}(n, \widetilde{x+y}, \tilde{x}, \tilde{y})$ above nx and $nx + y$ respectively (called differential multiplication).*

They can be computed via Luca sequences using the relations:

$$(\widetilde{n_1 + n_2})x = \texttt{diff\_add}(\widetilde{n_1 x}, \widetilde{n_2 x}, (\widetilde{n_1 - n_2})x), \tag{2.16}$$

$$(n_1 + \widetilde{n_2})x + y = \texttt{diff\_add}(\widetilde{n_1 x + y}, \widetilde{n_2 x}, (n_1 - \widetilde{n_2})x + y). \tag{2.17}$$

hence in time $O(\log(n))$ arithmetic operations over k (for instance via a standard double and add algorithm).

Proof. Since these operations are defined over the projective addition law on A, and since an isogeny is a group morphism, all these statements hold up to projective factors λ. To check directly on the equations that these projective factors satisfy $\lambda = 1$ with our choice of normalisations is a bit painful. The strategy in [LR12; Rob10] proceed as follow: first we check that Riemann relations are invariant under the action of S. (Although not proved there it is in fact invariant under all symmetric automorphisms, but the other ones are easier to prove). Then we check that the Riemann relations commute with isogenies of the first type, using the nomenclature of Example 2.5.7. Via the invariance under the action of S they also commute with isogenies of the second type, hence all isogenies coming from the isogeny formula. We remark that if $f : A \to B$ is an isogeny, the compatibility requires that $\tilde{f}(\tilde{0}_A) = \tilde{0}_B$, or equivalently that the normalisation factor $\lambda = 1$ in the isogeny formula.

Then we use the relation between the action of the theta group and differential additions [Rob10, Proposition 4.5] to derive associativity of differential multiplication for $x \in K(\mathcal{L})$. We use compatibility with the isogeny $[\ell]$ to go from associativity in $(A, \mathcal{L}^{\ell^2})$ for $x \in K([\ell]^*\mathcal{L}) = K(\mathcal{L}^{\ell^2})$ to associativity in $(A, \mathcal{L})$ of points in $[\ell]K(\mathcal{L}^{\ell^2}) = K(\mathcal{L}^{\ell})$. We give below an analytic proof, that proves the slightly stronger statement as given in the version of Lemma 2.8.3. □

Remark 2.8.5. There are two strategies to compute $m_1x_1 + \widetilde{\cdots} + m_nx_n$. Either we compute $x_1 + \widetilde{x_2} + x_3, x_1 + x_2 \widetilde{+} x_3 + x_4$ and so on using three way additions and then we do differential multiplications, for a cost of $O(\log m_1 + \log m_2 + \ldots \log m_k)$. Or we compute all $\widetilde{\epsilon_i m_i}$ with $\epsilon_i \in \{0, 1\}$ and then compute $m_1x_1 + \widetilde{\cdots} + m_nx_n$ via a multivariate double and add algorithm, for a total cost of $O(2^n + \log \max m_i)$.

The compatibility of differential additions and the action of the theta structure alluded in the proof above is as follow: given $\tilde{x}, \tilde{y}, \widetilde{x-y}$ in $\widetilde{A}$, and $g_1 = \Theta_{\mathcal{L}}(\alpha, i_1, i_2), g_2 = \Theta_{\mathcal{L}}(\beta, j_1, j_2)$ in $G(\mathcal{L})$, then

$$g_1g_2.\texttt{diff\_add}(\tilde{x}, \tilde{y}, \widetilde{x-y}) = \frac{\langle j_1, j_2\rangle}{\beta^2}\texttt{diff\_add}(g_1.\tilde{x}, g_2.\tilde{y}, g_1g_2^{-1}\widetilde{x-y}). \tag{2.18}$$

In particular:

$$(1, i_1 + j_1, i_2 + j_2).\texttt{diff\_add}(\tilde{x}, \tilde{y}, \widetilde{x-y}) = \texttt{diff\_add}((1, i_1, i_2).\tilde{x}, (1, j_1, j_2).\tilde{y}, (1, i_1 - j_1, i_2 - j_2).\widetilde{x-y}). \tag{2.19}$$

So the differential addition encode the action of $G(\mathcal{L})$ as follow: if g is above a point in $K_1(\mathcal{L})$ or $K_2(\mathcal{L})$, $\texttt{diff\_add}(g.\tilde{x}, g.\tilde{0}_A, \tilde{x}) = g^2\tilde{x}$, and if g_1, g_2 are above points in $K_1(\mathcal{L})$ or $K_2(\mathcal{L})$, $\texttt{threeway\_add}(\tilde{x}, g_1.\tilde{0}_A, g_2.\tilde{0}_A, g_1g_2\tilde{0}_A, g_2.\tilde{x}, g_1.\tilde{x}) = g_1g_2\tilde{x}$.

Key Idea 1. *By Equation (2.18) and Lemma 2.8.3, the differential addition or more generally the multiway multiplications allow to get a handle of the action of the theta group of level ℓn (either on the same abelian variety or via an ℓ-isogeny) by working only in level n.*

We will give several illustration of this key idea:

- How to recover the theta null point of level ℓn from the theta null point of level n and the points of ℓ-torsion;
- How to recover the pairing on $K(\mathcal{L}^{\ell})$ by working with theta coordinates of level n;
- How to compute an ℓ-isogeny $f : (A, \mathcal{L}^{\ell}) \to (B, \mathcal{M})$ using theta coordinates for $\mathcal{L}$.

2.8.3 *Analytic interpretation of the differential addition*

In fact, the differential addition can be seen as a way to encode the action of the theta group of all $G(\mathcal{L}^n)$ at once. To explain this we need to go back to complex abelian varieties. In this case the action of the theta groups is encoded by the transcendantal addition law.

So let $A = \mathbb{C}^g/\Lambda$ be a complex abelian variety, H a polarisation (which we assume is of the form $H = nH_0$ where H_0 is principal for simplicity, once again we refer to [Rob10] for the general case). Fixing an ∞-decomposition amount to fixing a symplectic decomposition $\Lambda = \Lambda_1 \oplus \Lambda_2$. To this symplectic decomposition one can associate a semi-character $\chi_0(\lambda_1 + \lambda_2) = e^{\pi i E(\lambda_1, \lambda_2)}$ (we recall that $E = \Im H$), hence a canonical symmetric line bundle $\mathcal{L}'$. This is the same line bundle as introduced algebraically above to be compatible with all the fixed symplectic decompositions of $K(\mathcal{L}^n)$. If $\mathcal{L}'_0$ is the line bundle corresponding to H_0 induced by the symplectic decomposition Λ, the analytic description above shows that $\mathcal{L}' = (\mathcal{L}'_0)^n$. Since $\mathcal{L}'_0$ is symmetric, we find again that $\mathcal{L}'$ is totally symmetric and does not depend on the decomposition whenever n is even. Let us denote $\mathcal{L} = \mathcal{L}'$.

The isotropic part Λ_1 contains a $\mathbb{C}$-basis, so using this we can rewrite the decomposition as $\Lambda = \mathbb{Z}^g \oplus \Omega\mathbb{Z}^g$ (with $H_0 = \mathfrak{I}\Omega^{-1}$). So the theta functions for the symmetric theta structure on $\mathcal{L}$ associated to the symplectic decomposition of Λ are then given by the $(\theta\left[\begin{smallmatrix}0\\b\end{smallmatrix}\right](\cdot,\Omega/n))_{b\in\frac{1}{n}\mathbb{Z}^g/\mathbb{Z}^g}$. Indeed if $A[n] = (\frac{1}{n}\mathbb{Z}^g \oplus \frac{1}{n}\Omega\mathbb{Z}^g)/\Lambda$, $A/A_2[n] \simeq \mathbb{C}^g/(\mathbb{Z}^g \oplus \frac{1}{n}\Omega\mathbb{Z}^g)$ so $\theta\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right](\cdot,\Omega/n)$ is indeed the unique (up to multiple) section of $\mathcal{L}$ that descends to $A/A_2[n]$, and the automorphic properties of the theta functions show that $\theta\left[\begin{smallmatrix}0\\b\end{smallmatrix}\right](\cdot,\Omega/n)$ is indeed the action of the lift of b on $\theta\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$ via the identification $\frac{1}{n}\mathbb{Z}^g/\mathbb{Z}^g \simeq Z(\overline{n})$. The duplication formula of Corollary 2.7.2 then corresponds to the standard duplication formula on the analytic theta, since an immediate computation shows that $U^{\mathcal{L}}_{\chi,i}$ corresponds to $\theta\left[\begin{smallmatrix}\chi/2\\2i/n\end{smallmatrix}\right](2\cdot,4\Omega/n)$. We refer to [Rob10, § 2.6 and Exemple 4.4.9] for more details. Of course, since algebraic theta functions are only defined up to a constant, this identification is up to a constant, ie as projective coordinates. We refer to [Mum67b; Mum91] for the determination of this constant under suitable normalisations.

Now the important part is that on $\mathbb{C}^g$ there is an addition law above the addition law on $A = \mathbb{C}^g/\Lambda$. And the analytic theta make sense of as *affine coordinates* on $\mathbb{C}^g$. We warn that since $\theta\left[\begin{smallmatrix}0\\b\end{smallmatrix}\right](z+\Omega m_1+m_2,\Omega/n) = e^{-i\pi n\,{}^t m_1\Omega m_1-2\pi i n\,{}^t z m_1}\theta\left[\begin{smallmatrix}0\\b\end{smallmatrix}\right](z,\Omega/n)$, not all affine theta lift $\widetilde{P}$ of a point $P \in A$ comes from the affine theta coordinates of a $z \in \mathbb{C}^g$ above P (ie such that $P = z \mod \Lambda$). Still, there is always a countable number of such *good affine lifts*.

Now the standard analytic Riemann relations are valid for points on $\mathbb{C}^g$, ie are valid for affine theta coordinates, not only projective theta coordinates [Mum83]. (The standard form of Riemann relations is different than the one of Theorem 2.7.3, but Mumford shows in [Mum66, p. 334-335] how this is just a linear change of variable, see [Rob10, Remarque 4.4.8].) Hence:

Key Idea 2. *The differential addition is a way to recover this* transcendantal addition law*: given $\tilde{x},\tilde{y},\widetilde{x-y} \in \mathbb{C}^g$ and letting $\tilde{0}_A = 0 \in \mathbb{C}^g$, and given the values of the affine theta coordinates at these points, $\widetilde{x+y} =$ `diff_add`$(\tilde{x},\tilde{y},\widetilde{x-y},\tilde{0}_A)$ gives the affine theta coordinates of $\tilde{x}+\tilde{y}$.*

We remark that $\mathbb{C}^g$ can be analytically identified with $T_0(A)$. Hence the transcendental addition law on $\mathbb{C}^g$ has an analytic interpretation as the algebraic addition law on $T_0(A)$ which is derived from the addition law on A. If $\pi : \mathbb{C}^g \to A$ is the projection, and $\mathcal{L}$ is a line bundle on A, then $\pi^*\mathcal{L}$ is always trivial on $\mathbb{C}^g$ (since it is simply connected). Hence we may choose a global rigidification. This essentially defines the affine values of the theta functions on $\mathbb{C}^g$ (if we interpret the choice of period matrix as a choice of rigidification).

Let us illustrate this with the isogeny formula. The theta functions for $\mathcal{L}^\ell$ (with the same symplectic decomposition of Λ) are given by the $\theta\left[\begin{smallmatrix}0\\b\end{smallmatrix}\right](\cdot,\Omega/\ell n)$ for $b \in \frac{1}{\ell n}\mathbb{Z}^g/\mathbb{Z}^g$. The symplectic decomposition $A[\ell] = (\frac{1}{\ell}\mathbb{Z}^g \oplus \frac{1}{\ell}\Omega\mathbb{Z}^g)/\Lambda$ shows that $B = A/A_2[\ell] = \mathbb{C}^g/(\mathbb{Z}^g \oplus \frac{\Omega}{\ell}\mathbb{Z}^g)$. The descent of $\mathcal{L}^\ell$ to B has for Hermitian form ℓH on $\mathbb{C}^g$. So it is of level n on B, and the corresponding theta functions are $\theta\left[\begin{smallmatrix}0\\b\end{smallmatrix}\right](\cdot,\frac{\Omega}{\ell}/n)$ for $b \in \frac{1}{n}\mathbb{Z}^g/\mathbb{Z}^g$. We recover exactly the isogeny formula from Example 2.5.7 when writing $b \in \frac{1}{n}\mathbb{Z}^g/\mathbb{Z}^g = \ell b_0$, $b_0 \in \frac{1}{\ell n}\mathbb{Z}^g/\mathbb{Z}^g$. But this actually gives more: for affine theta coordinates, the constant in the isogeny is $\lambda = 1$. In other words, with $\lambda = 1$ in the isogeny formula (which we will henceforth call the affine isogeny formula), it becomes compatible with the differential additions. Equivalently we take an affine lift $\tilde{f}$ of the isogeny f such that $\tilde{f}(\tilde{0}_A) = \tilde{0}_B$. This is valid for isogenies of the first type, but using the action of S as in Example 2.5.7 shows that it is valid for all isogenies (see also [Kem91, §5.3]).

Since there is an infinite number of *good affine lifts*, this shows that the differential addition law commutes with the affine isogeny formula for *arbitrary* affine lifts (this is also easy to check directly by homogeneity once we know it is valid for *one* affine lift). The case of $\tilde{0}_A$ is special since it is fixed to being $0 \in \mathbb{C}^g$ (ie $\tilde{0}_A = \theta\left[\begin{smallmatrix}0\\b\end{smallmatrix}\right](0,\Omega/n)$), but there are still an infinite number of possible affine lift when we acts on Ω by $\Gamma(n,2n) \subset \mathrm{Sp}_{2n}(\mathbb{Z})$. This does not change the projective theta constants, but change their affine lifts due to the functional equation of the theta functions.

By standard lifting arguments (see [Rob21, Section 2.3.6]), the differential addition law commutes with the affine isogeny formula for arbitrary fields of characteristic p prime to ℓn.

It is interesting to compare the anayltic proof with the purely algebraic one outlined in the proof of Lemma 2.8.3. Algebraically, we do not have a lattice Λ but we do have the Tate module $T_\ell(A)$, and the compatibility of the theta structures on $G(\mathcal{L}^{\ell^m})$ fixed by an ∞-decomposition glue together to a theta structure on $T_\ell(A)$ (this theme is explored in much more details in [Mum67a] where Mumford develops a theory of algebraic adic theta functions). If we are interested in understanding the theta groups at finite levels only, there is no need to pass to the limit; we may simply use the isogeny of multiplication by $[\ell]$ to relate $G(\mathcal{L}^{\ell^2})$ and $G(\mathcal{L})$. Since $[\ell]\,K(\mathcal{L}^{\ell^2}) = [\ell]^{-1}K(\mathcal{L})$,

the theta structure of $G(\mathcal{L}^{\ell^2})$ reflects into the behaviour of points of ℓ-torsion in coordinates given by the base polarisation $\mathcal{L}$. We thus get the following refinement of Key Idea 1:

Key Idea 3. *The way the action of the theta group $G(\mathcal{L}^\ell)$ is reflected into Riemann relations on points in $[\ell]^{-1}K(\mathcal{L})$ in $(A,\mathcal{L})$ can be described using the theta structure on $G(\mathcal{L}^{\ell^2})$ and the isogeny $[\ell]$.*

2.8.4 Applications of the differential addition

Compressing coordinates

Example 2.8.6. A first application of the compatibility of differential additions (or more generally Riemann relations) with isogenies is the following: let $(A,\mathcal{L})$ be an abelian variety with a theta structure of level ℓn, let $\pi : A \to B = A/K_2(\mathcal{L})[\ell]$ be the isogeny of the first type from Example 2.5.7 and let $\widetilde{\pi}$ be the canonical affine lift of π (given by the affine isogeny formula).

Then for $i \in K_1(\mathcal{L})$, we define $\widetilde{\pi}_i(P) = \widetilde{\pi}(s(i)\cdot P)$ where $s(i) \in \widetilde{K}_2(\mathcal{L})$ is the lift of i given by the theta structure. Let $e_1,\dots,e_g$ be a basis of $K_1(\mathcal{L})$ and for $i<j \in \{1,\dots,g\}$, define $\widetilde{\pi}_i = \widetilde{\pi}_{e_i}$, $\widetilde{\pi}_{ij} = \widetilde{\pi}_{e_i+e_j}$.

Then by [Rob10, § 4.6]:

- The theta null point $\tilde{0}_A$ is completely determined by the $1+g(g+1)/2$ affine points on $\widetilde{B}$: $\widetilde{\pi}(\tilde{0}_A), \widetilde{\pi}_i(\tilde{0}_A), \widetilde{\pi}_{ij}(\tilde{0}_A)$.
- An affine lift $\tilde{x} \in \widetilde{A}$ is completely determined by $\tilde{0}_A$ and the $1+g$ affine points on $\widetilde{B}$: $\widetilde{\pi}(\tilde{x}), \widetilde{\pi}_i(\tilde{x})$.

Indeed, using the multiway additions from Lemma 2.8.3 and the compatibility with isogenies, we can recover all $\widetilde{\pi}_i(\tilde{0}_A)$ (resp. $\widetilde{\pi}_i(\tilde{x})$) for $i \in K_1(\mathcal{L})$, and by Example 2.5.7 we have $\theta_i(\tilde{x}) = \theta_0(\widetilde{\pi}_i(\tilde{x}))$. We will see in Section 2.10 why the number of points to reconstruct $\tilde{0}_A$ is natural.

Of course, if ℓ is prime to n we may take a basis of $K_1(\mathcal{L})[\ell]$ instead. All these points are needed, so we cannot compress further: from the description of Remark 2.6.6 we can always find a symmetric automorphism that change only one of $\widetilde{\pi}_i(\tilde{0}_A)$, or $\widetilde{\pi}_{ij}(\tilde{0}_A)$. Likewise, we can always find a translation action by $g \in \widetilde{K}_2[\ell]$ such that $\widetilde{\pi}(g.\tilde{x})$ only change one of $\widetilde{\pi}_i(\tilde{x})$.

As a specific example, let $g=1$, $\ell=3$, $n=4$. If $\tilde{x} = (\tilde{x}_0,\tilde{x}_1,\tilde{x}_2,\tilde{x}_3,\tilde{x}_4,\tilde{x}_5,\tilde{x}_6,\tilde{x}_7,\tilde{x}_8,\tilde{x}_9,\tilde{x}_{10},\tilde{x}_{11})$, then it is completely determined by

$$\begin{aligned}\widetilde{\pi}_0(\tilde{x}) &= (\tilde{x}_0,\tilde{x}_3,\tilde{x}_6,\tilde{x}_9)\\ \widetilde{\pi}_4(\tilde{x}) &= (\tilde{x}_4,\tilde{x}_7,\tilde{x}_{10},\tilde{x}_1)\\ \widetilde{\pi}_8(\tilde{x}) &= (\tilde{x}_8,\tilde{x}_{11},\tilde{x}_2,\tilde{x}_5)\end{aligned}$$

But $\widetilde{\pi}_8(\tilde{x}) = \texttt{diff\_add}(\widetilde{\pi}_4(\tilde{x}),\widetilde{\pi}_4(\tilde{x}),\widetilde{\pi}_0(\tilde{x}),\tilde{0}_B)$ so we only need two points.

We will also apply Example 2.8.6 to compute the fibers of the isogeny $f : A \to B$.

Compatibility of Riemann's relations

It remains to explain the compatibility of differential addition and Riemann relations between $G(\mathcal{L})$ and $G(\mathcal{L}^\ell)$ (ie when changing level) once we have fixed an ∞-structure. This is best explained using the Segre embedding.

- Let $(A,\mathcal{L})$ be an abelian variety with a theta structure of level n, and fix the product theta structure on $(A^\ell,\mathcal{L}^{\star\ell})$. Then $\theta_{i_1,\dots,i_\ell}(\tilde{x}_1,\dots,\tilde{x}_\ell) = \prod_{k=1}^{\ell}\theta_{i_k}(\tilde{x}_k)$. It is easy to see that Riemann relations are compatible with the (affine) Segre embedding, in other words if $\tilde{x}_1,\tilde{y}_1,\tilde{u}_1,\tilde{v}_1,\tilde{x}_2,\tilde{y}_2,\tilde{u}_2,\tilde{v}_2$ are in Riemann relations on $(A_1,\mathcal{L}_1)$, and $\tilde{x}'_1,\tilde{y}'_1,\tilde{u}'_1,\tilde{v}'_1,\tilde{x}'_2,\tilde{y}'_2,\tilde{u}'_2,\tilde{v}'_2$ are in Riemann relations on $(A_2,\mathcal{L}_2)$, then the pairs $(\tilde{x}_1,\tilde{x}'_1),\dots(\tilde{v}_2,\tilde{v}'_2)$ are in Riemann relations on $(A_1\times A_2,\mathcal{L}_1\star\mathcal{L}_2)$. In particular $\texttt{diff\_add}(\tilde{x},\tilde{y},\widetilde{x-y})$ is the Segree embedding of the ℓ points $\texttt{diff\_add}(\tilde{x}_i,\tilde{y}_i,\widetilde{x-y}_i)$ of $\widetilde{A}$. This is a consequence of the analytic interpretation of the differential addition, but it is also easy to check directly.
- Now if $\Delta : A \to A^\ell$ is the diagonal embedding, $\Delta^*\mathcal{L}^{\star\ell} = \mathcal{L}^\ell$. Likewise, we can check analytically (hence algebraically by the usual lifting arguments) that the (affine) diagonal embedding is compatible with Riemann relations, since the diagonal embedding on A lifts to a transcendantal diagonal embedding.

- Combining the two, we get that the differential addition on $(A, \mathcal{L}^\ell)$ is compatible with ℓ-fold products of theta functions for the differential addition of $(A, \mathcal{L})$. More precisely, by surjectivity of the multiplication, there is a non inversible base change L between the basis $\theta_i^{\mathcal{L}^\ell}$ and the generators $\prod_{k=1}^{\ell} \theta_{i_k}^{\mathcal{L}}$. In other words L describe the map $(A, \mathcal{L}^\ell) \to (A^\ell, \mathcal{L}^{\star\ell})$, and explains how an affine lift for $\mathcal{L}$ induces an affine lift for $\mathcal{L}^\ell$. We also have the diagonal embedding (which does not respect polarisation), $(A, \mathcal{L}) \to (A^\ell, \mathcal{L}^{\star\ell})$. Then the Riemann relations commute with these two (affine) mappings.

  We note that if $\tilde{x}$ is an affine lift of x for $\mathcal{L}$, then changing $\tilde{x}$ by $\zeta\tilde{x}$ where $\zeta^\ell = 1$ induce the same affine point on $(A, \mathcal{L}^\ell)$. We may reinterpret this as follows: the choice of $\tilde{x}$ may be seen as a choice of trivialisation of $\mathcal{L}$ at x (which we then use to evaluate the theta functions at x). This trivialisation induces a trivialisation of $\mathcal{L}^\ell$, hence an affine lift $\tilde{x}'$ of x for $\mathcal{L}^\ell$, which is exactly the same as the lift above induces by the ℓ-fold products of $\theta_i^{\mathcal{L}}(x)$. But the induced trivialisation of $\mathcal{L}^\ell$ does not change if we act on the trivialisation of $\mathcal{L}$ by ζ.

In Section 2.10.2 we will explain how to use this compatibility to compute the action of the theta group of $G(\mathcal{L}^\ell)$ on the product basis $\prod_{k=1}^{\ell} \theta_{i_k}^{\mathcal{L}}$, and how to recover the theta null point of level ℓn from the theta null point of level n and the coordinates of the points of ℓ-torsion.

Remark 2.8.7. At this point it is probably useful to relate all compatibility of differential additions and Riemann relations we have (and add some others).

(i) Over $\mathbb{C}$, Riemann relations encode the transcendental addition law;
(ii) Differential additions are compatible with the action of the theta group. In particular, whenever we define a group morphism of theta group we may ask if this further induces a compatibility of Riemann relations.
(iii) If $f : (A, \mathcal{L}) \to (B, \mathcal{M})$ is an isogeny, $\widetilde{K}$ the descent data associated to it, we have a morphism $Z(\widetilde{K}) \to G(\mathcal{M}) = Z(\widetilde{K})/\widetilde{K}$. This induces the isogeny formula on theta function, and the Riemann relations are compatible with this formula.
(iv) The Segre embedding $(A, \mathcal{L}) \times (B, \mathcal{M}) \to (A \times B, \mathcal{L} \star \mathcal{M})$ induces a morphism $G(\mathcal{L}) \times G(\mathcal{M}) \to G(\mathcal{L} \star \mathcal{M})$. The theta functions on $A \times B$ with the product theta structure are just the product theta functions, and Riemann relations are compatible with the product theta structure.
(v) We may combine the Segre embedding and isogenies to get what we call a generalised Segre embedding (or more precisely morphism, since it may no longer be an embedding): let $\ell = \sum_{i=1}^{r} n_i^2$ and let $F : A \to A^r, x \mapsto (n_i x)$. Then $F^*(\mathcal{L} \star \cdots \star \mathcal{L}) = \mathcal{L}^\ell$. We may generalise this to endomorphisms $n_i = \alpha_i$ provided we have an affine lift $\widetilde{\alpha_i}$ compatible with Riemann's relations.
(vi) Mumford defines in [Mum66] several morphisms between the $G(\mathcal{L}^n)$: letting $\delta_{-1} = \gamma_{-1}$, we have $\delta_n(g) = g^{(n^2+n)/2}\delta_{-1}(g)^{(n^2-n)/2} : G(\mathcal{L}) \to G(\mathcal{L})$. The reason we don't use g^n is that we want that the version of δ_n on the Heisenberg group to be given by $(\alpha, x_1, x_2) \mapsto (\alpha^{n^2}, nx_1, nx_2)$. If g is symmetric, $\delta_n(g) = g^n$.
We also have a natural morphism $\epsilon_n : G(\mathcal{L}) \to G(\mathcal{L}^n)$ which sends (x, ϕ) to $(x, \phi^{\otimes n})$. The version on the Heisenberg groups map $H(m) \to H(nm)$ via $(\alpha, x_1, x_2) \mapsto (\alpha^n, x_1, x_2)$ where we use the natural embeddings of $Z(m)$ and $\widehat{Z}(m)$ into $Z(nm)$, $\widehat{Z}(nm)$.
Finally we also have a morphism $\eta_n : G(\mathcal{L}^n) \to G(\mathcal{L})$. Mumford's description (using the symmetry of $\mathcal{L}$) is a bit technical, but using a ∞-decomposition, we can describe $\eta_n(g)$ as the descent of $\epsilon_n(g) \in G(\mathcal{L}^{n^2})$ via the isogeny $[n]$. On the Heisenberg groups, this maps $H(nm) \to H(m)$ via $(\alpha, x_1, x_2) \mapsto (\alpha^n, nx_1, nx_2)$ and the natural embeddings above.
They satisfy the following relations: η_n and ϵ_n commute with γ_{-1}, $\epsilon_n \circ \eta_n = \delta_n$ and $\eta_n \circ \epsilon_n = \delta_n$.
Then if we have a symmetric theta structure on $(\mathcal{L}, \mathcal{L}^2)$, the duplication formula is compatible with Riemann relations. In fact we have seen in Theorem 2.7.3 that it is the essence of the proof of Riemann's relations.

2.9 AFFINE LIFTS AND DIFFERENTIAL ADDITION LAW IN OTHER MODELS

2.9.1 *Functions constructed from an explicit version of the theorem of the square*

Given the central role of the Riemann relations in our algorithms, for reasons we outlined in Section 2.8.2, it may seem that to extend them to other models require new ideas. But a recent insight is as follow:

Key Idea 4. *The differential addition and the three way addition are just explicit versions of the theorem of the square.*

We recall that if $x, y \in A$, then the theorem of the square states that $t_{x+y}^*\mathcal{L} \otimes \mathcal{L} \simeq t_x^*\mathcal{L} \otimes t_y^*\mathcal{L}$, see [Rob21, Corollary 2.2.6]. Switching to the vision of divisors, if Θ is a divisor corresponding to $\mathcal{L}$, then an explicit version of the theorem of the square means computing a section of $t_{x+y}^*\Theta + \Theta - t_x\Theta - t_y\Theta$.

To simplify notations, we can send a zero cycle $Z = \sum n_i(P_i)$ to the divisor $s_\Theta(Z) := D_{\Theta,Z} = \sum n_i t_{P_i}^* \Theta$ (we call this D_Z if Θ is clear from the context). If Z is of degree zero, $s_\Theta(Z)$ correspond to a line bundle algebraically equivalent to zero.

Lemma 2.9.1. *With the notations above, if Z is of degree zero, $D_{\Theta,Z}$ is linearly equivalent to zero (ie principal) if and only if its realisation $S(Z) = \sum n_i P_i \in K(\mathcal{L})$, the kernel of the polarisation $\Phi_\mathcal{L} : A \to \widehat{A}$.*

In this case we call $f_{\Theta,Z}$ a function representing $D_{\Theta,Z}$.

Proof. Using the theorem of the square repeatedly, we have that $s_\Theta(Z)$ is linearly equivalent to $s_\Theta((\sum n_i P_i) - (0))$. But by definition of $\Phi_\mathcal{L}$, $s_\Theta((\sum n_i P_i) - (0))$ correspond to the line bundle $\Phi_\mathcal{L}(\sum n_i P_i)$, so this divisor is principal if and only if $\sum n_i P_i \in K(\mathcal{L})$. □

These functions should be reminiscent to the ones used for pairings using Miller's algorithm, and we will of course see them again in Chapter 3.

As a first example, the differential addition which allows to compute products of the form $\theta_i(x+y)\theta_j(x-y)$ in terms of products $\theta_i(x)\theta_j(x)$ (and constants depending on y and 0_A), can be seen as making explicit the isomorphism $t_y^*\mathcal{L} \otimes t_{-y}^*\mathcal{L} \simeq \mathcal{L}^2$. More precisely the products are well defined when we choose a trivialisation of these line bundles at x, and the constants relate these two trivialisations.

Likewise, the three way addition, which relate products of the form $\theta_i(x+y+z)\theta_j(x)$ with products of the form $\theta_i(x+y)\theta_j(x+z)$ (and constants depending on y, z and 0_A) can be seen as an explicit version of the isomorphism $t_{y+z}^*\mathcal{L} \otimes \mathcal{L} \simeq t_y^*\mathcal{L} \otimes t_z^*\mathcal{L}$ via appropriate trivialisations at x.

Conversely, we may ask if it is possible to recover a theta structure given a model with an explicit version of the theorem of the square. In fact, there are two related versions we may need. It is more convenient here to use the language of divisors, we let Θ be a divisor representing $\mathcal{L}$.

Algorithmic Hypothesis 2.9.2.
(i) Given $y, z \in A$, compute a function $g_{y,z}(x) := f_{\Theta,(y)+(z)-(y+z)-(0_A)}(x)$ representing the principal divisor $t_{y+z}^*\Theta + \Theta - t_y^*\Theta - t_z^*\Theta$;
(ii) Given $y \in K(\Theta)$, compute a function $g_y(x) := f_{\Theta,(y)-(0_A)}$ representing the principal divisor $t_y^*\Theta - \Theta$.

Of course Algorithmic Hypotheses 2.9.2.(ii) is vacuous if $\mathcal{L}$ is principal.

Example 2.9.3. If E/k is an elliptic curve and $\Theta = (0_E)$, Algorithmic Hypothesis 2.9.2 is satisfied if we know how to compute the (normalised) function $\mu_{P,Q}$ with divisor $(P) + (Q) - (P+Q) - (0_E)$ on our model.

Algorithmic Hypotheses 2.9.2.(i) allows to compute the functions $f_{\Theta,Z}$ for any 0-cycles of degree zero such that its realisation $S(Z) = 0_A$ (by successive reduction, see also Section 3.3 for a double and add algorithm when applicable), and adding Algorithmic Hypotheses 2.9.2.(ii) allows to handle the case $S(Z) \in K(\Theta)$.

Note that the functions g are only defined up to a constant. This won't matter if they are used for evaluation on a degree zero cycle as in Chapter 3. It is customary to normalize them at a point P, typically $P = 0_A$ by requiring that $g(P) = 1$. This is only possible if g is well defined at P, ie P is neither a zero or pole. Otherwise a possibility is to fix a constant in a Laurent series development along some uniformisers at P. This is classical for elliptic curves for pairing evaluation, we refer to Section 3.5 for some examples.

Remark 2.9.4. An alternative strategy to fix the normalisation constant, when $\mathcal{L}$ is base point free, is to look at translations by points of $K(\Theta)$ to normalize g. Assume for instance that Θ is invariant by $K_2(\mathcal{L})$ (eg Θ is the divisor associated to θ_0), and fix once and for all a function g_y for any $y \in K_1(\mathcal{L})$. Then since the g_y generate the sections of $\mathcal{L}$, there is always one y such that $g(P+y)$ is well defined. Here we use $\widetilde{K}_1(\mathcal{L})$ to translate the rigidification of P to a rigidification of $P+y$. We will go back to this in Remark 3.3.5.

2.9.2 Computing a theta structure

Now we may recover a theta structure from Algorithmic Hypotheses 2.9.2.(ii) as follow. The function g_y of Algorithmic Hypotheses 2.9.2.(ii) defines an element g_y of $G(\mathcal{L})$ above y via its action on the sections $s \in \Gamma(A, \mathcal{L})$ by $\mathrm{g}_y \cdot s = x \mapsto g_y(x)s(x-y)$.

Let $s_1, \dots, s_N$ be a basis of sections of $\Gamma(A, \mathcal{L})$. Evaluating at several points, we may recover the action of g_y on the basis s_i via linear algebra. We may then compute the commutator pairing explicitly (or via the tools of Chapter 3), fix a symplectic decomposition of $K(\mathcal{L})$, and adjust the normalisations of the g_y so that $y \mapsto \mathrm{g}_y$ is a group morphism over $K_1(\mathcal{L})$ and $K_2(\mathcal{L})$. Note that since the action of $G(\mathcal{L})$ on $\Gamma(A, \mathcal{L})$ is irreducible, then picking any section $s \in \Gamma(A, \mathcal{L})$, the set $\mathrm{g}_y \cdot s, y \in K(\mathcal{L})$ is a generator. In particular, the functions g_y for $y \in K(\mathcal{L})$ generate the sections.

Remark 2.9.5. We remark that g_y is symmetric precisely when $g_y(-x) = g_y(x+y)^{-1}$. So if y is of odd order m, there is a unique g_y such that g_y is symmetric and of order m. If $\mathcal{L}$ is of order n even, a symmetric theta structure exists if and only if $\mathcal{L}$ is totally symmetric. Then we may choose any of the two symmetric element g_{e_i} above a symplectic basis e_i of $K(\mathcal{L})$ to get a symmetric theta structure. Indeed we know that such a structure exists, and acting on this by conjugation by points of two torsion we get all the 2^{2g} possible choices.

As an example, if we already have a (symmetric) theta structure on $\mathcal{L}$, we let Θ be the divisor associated to θ_0. A basis of sections of Θ is given by the θ_i/θ_0, $i \in K_1(\mathcal{L})$, and $g_y(x) = \mathrm{g}_y \cdot \theta_0/\theta_0$. For instance if $i \in K_1(\mathcal{L})$, $g_y(x) = \theta_i(x)/\theta_0(x)$.

Remark 2.9.6. Another very important example where we have an explicit version of the theorem of the square (in the form where we may evaluate the functions g_y on points) is in the case $A = \mathrm{Jac}(C)$ is a Jacobian. This is the content of [CE14, § 2, § 3, § 4] to which this section owes a lot.

Assume that $J = \mathrm{Jac}(C)$ and that C as a rational point O. We let Θ be the theta divisor (in practice we translate by a theta characteristic to get a symmetric divisor). If f is a function on C with divisor $\mathrm{Div} f = \sum(P_i) - \sum(Q_i)$, it extends to a function F on $\mathrm{Jac}(C)$ via $F(x) = f(D_x)$ where, if $D_x = \sum(T_i)$, $f(D_x) = \prod f(T_i)$, and D_x is the unique divisor of degree g such that $D_x - g(O)$ represents x. This gives a section $f_{\Theta,Z}$ associated to a zero cycle of the form $Z = \sum n_i(P_i - O)$ where all P_i are on the curve [CE14, § 2.2]. But a general degree zero cycle Z is always equivalent to a cycle Z_0 of the form above, and determinants can be used to compute the linear equivalence between $s_\Theta(Z)$ and $s_\Theta(Z_0)$ [CE14, § 2.5].

Recipe 2.9.7. Algorithmic Hypotheses 2.9.2.(i) can be used as follow:

- Generate sections of $\mathcal{L}^\ell$ by constructing $f_{\Theta,Z}$ for $Z = \ell(P) - \ell(0)$ with $P \in A[\ell]$. If ℓ is prime to the level n of $\mathcal{L}$, these sections suffice to generate $\Gamma(A, \mathcal{L}^\ell)$ when we take their product with generators of $\Gamma(A, \mathcal{L})$.
- If we also have Algorithmic Hypotheses 2.9.2.(ii), we can proceed as above to generate sections of $\mathcal{L}^\ell$ by using $Z = \ell(P) - \ell(0)$ with $P \in [\ell]^{-1}K(\mathcal{L}) = A[\ell n]$ by Theorem 2.5.1. If ℓ is prime to n, it suffice to look at $Z = \ell(P) + (Q) - (\ell+1)(0)$, with $P \in A[\ell]$ and $Q \in A[n]$. We recover the method above as a special case.
- If we don't want to iterate through all points of ℓ-torsion, we may instead look at Z of the form $Z = b(aP) + a(-bP) - \ell(0)$ for fixed $a, b \in \mathbb{N}$ such that $a + b = \ell$ and P a random point, as in [CE14, § 3].
- Once we have a system of generators, we may do linear algebra (by evaluating these generators on points) to extract a basis of $\Gamma(A, \mathcal{L}^\ell)$.
- Since we have just seen that we can compute functions g_y with divisor $\ell t_y^*\Theta - \ell\Theta$ for $y \in A[\ell]$, or even $y \in A[n\ell]$ if we also have Algorithmic Hypotheses 2.9.2.(ii), then we may compute the explicit action of the theta group $G(\mathcal{L}^\ell)$ on a basis. Via a choice of theta structure (using the Remark 2.9.5 if we want a symmetric one) we can then apply Recipe 2.5.3 to get the corresponding theta functions for $\mathcal{L}^\ell$.

  Note that once we have computed the g_y for $y \in K_i(\mathcal{L})$, this Recipe only requires one section s whose trace under $\widetilde{K}(\mathcal{L})_2$ is non zero. The section θ_0 is the trace of s under $\widetilde{K}(\mathcal{L})_1$ and the θ_i are given by the action of the g_i on θ_0.

  Note that if we only have Algorithmic Hypotheses 2.9.2.(i), we may still compute lifts $\widetilde{K}$ of isotropic subgroups K of $A[\ell]$ and compute the action of $\widetilde{K}$ on sections, this is sufficient for isogenies as we will see in Chapter 4.
- In particular, by Remark 2.9.6, we have an explicit algorithm to compute theta constants of any level n (prime to p) on a Jacobian, provided we have its points of n-torsion. This generalizes Thomae's formula.

  In [CE14] the authors use their explicit version of the theorem of the square to compute isogenies on Jacobians, but as we just outlined it is straightforward using Recipe 2.5.3 to extend their construction to get a symmetric theta structure of level ℓn. Then we may compute an ℓ-isogeny $f : A \to B$ directly using Theorem 2.5.6 to get a theta structure of level n on B, hence in particular equations for B by Section 2.7 if $n > 2$ is even. This has the advantage of not requiring B to be a Jacobian, as was needed in [CE14] to do computations (hence restricting its usage to the generic case of genus $g \leq 3$).

Note that the above considerations apply in particular to construct a theta structure of level ℓn given the theta functions of level n. Indeed, in this case Algorithmic Hypotheses 2.9.2.(ii) is built in into the definition of the theta functions, while Algorithmic Hypotheses 2.9.2.(i) is given by the differential addition as we saw. In practice, it

will be easier to apply a slightly different strategy to construct sections of $\Gamma(A, \mathcal{L}^\ell)$ by using the surjectivity of the multiplication map instead, ie to look at ℓ-fold products of theta functions of level n. Indeed in this case the action of $G(\mathcal{L}^\ell)$ on such products is easier to describe (see Section 2.10.2), so we do not need to use the general strategy. We note that these products essentially correspond to functions $f_{\Theta,Z}$ for Z of the form $Z = \sum(P_i) - \ell(0)$ where $P_i \in K_1(\mathcal{L})$ (if we choose Θ given by θ_0), so can be seen as a special case of the general strategy.

2.9.3 *Trivialisations of the line bundle*

We have seen that differential additions and three way additions relate suitable trivialisations of the line bundle $\mathcal{L}$ at points $x \in A(\bar{k})$ (or alternatively trivialisation of $t_{-x}^*\mathcal{L}$ at 0_A). We note that Key Idea 4 is selling the differential addition a bit short: the point is that furthermore the normalisations used in making the theorem of the square explicit are uniform. One way to look at this is that the theorem of the square is a special form of the theorem of the cube. For instance, $p_{x+y+z}^*\mathcal{L} \otimes p_{x+y}^*\mathcal{L}^{-1} \otimes p_{x+y}^*\mathcal{L}^{-1} \otimes p_{y+z}^*\mathcal{L}^{-1} \otimes p_{x+z}^*\mathcal{L}^{-1} \otimes p_z^* \otimes p_x^* \otimes p_y^*$ is trivial on $A \times A \times A$, where I denote by p_{x+y} the morphism $A \times A \times A \to A, (x, y, z) \mapsto x + y$. So fixing a trivialisation of this line bundle allows to fix (uniformly) a trivialisation of $t_{x+y+z}\mathcal{L}$ from trivialisations of $t_x\mathcal{L}$, $t_y\mathcal{L}$, $t_z\mathcal{L}$, $t_{y+z}\mathcal{L}$, $t_{x+z}\mathcal{L}$, $t_{x+y}\mathcal{L}$. We won't need this strengthening of Algorithmic Hypothesis 2.9.2 in the following.

Finally, there is a very important modular interpretation of fixing a trivialisation of $\mathcal{L}$ at 0. If we have a symmetric theta structure of level n even on $(A, \mathcal{L})$ with $n \geq 4$, a trivialisation of $\mathcal{L}$ at 0_A, ie of the choice of an affine lift $\tilde{0}_A$ of the theta null point, is determined, up to a sign, by the choice of a differential basis on A.

Indeed, let $\mathcal{X}_{g,n} \to \mathcal{A}_{g,n}$ be the universal abelian variety, $s : \mathcal{A}_{g,n} \to \mathcal{X}_{g,n}$ be the zero section, and $\mathcal{H} = s^*\Omega_{\mathcal{X}_{g,n}}$ be the Hodge vector bundle. A trivialisation of $\mathcal{H}$ over an abelian variety A then corresponds to a choice of basis of the cotangent sheaf of A at 0_A. Alternatively, since on an abelian scheme all global differentials are invariant by translation (since there are no global sections of the structure sheaf), one can also see the Hodge vector bundle as $\pi_*(\Omega_{\mathcal{X}_{g,n}})$, and a trivialisation of $\mathcal{H}$ as a basis of global differentials of A. We denote by $\mathfrak{h} = \Lambda^g s^*\Omega_{\mathcal{X}_{g,n}}$ the Hodge line bundle. Let $\mathcal{L}_{\mathcal{A}_{g,n}}$ be the ample line bundle given by the pullback of the $O(1)$ sheaf via the quasi projective embedding of $\mathcal{A}_{g,n}$ of Theorem 2.7.4 given by the theta null coordinates.

Lemma 2.9.8. *Over $\mathcal{A}_{g,n}$, we have $\mathfrak{h} = \mathcal{L}^2_{\mathcal{A}_{g,n}}$.*

So the trivialisation we chose on $\mathcal{L}$ induces a trivialisation of $\mathfrak{h}_A$. On the other hand a choice of a differential basis w_A also fixes a trivialisation on $\mathcal{H}_A$ so on $\mathfrak{h}_A$ over A (which depends only on $\Lambda^g w_A$), hence of $\mathcal{L}^2$.

Proof. The dual of $\mathcal{L}_{\mathcal{A}_{g,n}}$, which is then the pullback of the $O(-1)$ sheaf is called $\mathcal{K}$ in Mumford [Mum67a, p. 82]. In [Can16, Th. 4.2.1], it is proved that $\mathcal{K}$ is the pullback of the inverse of a square root of the Hodge line bundle defined over $\mathcal{A}_g$: $K = \mathrm{Des}^* w_\Theta^{-1/2}$, see [Can16, Rem. 4.2.2]. There, w_Θ is the Hodge line bundle twisted by the theta multiplier bundle M_Θ, but the pullback of this multiplier bundle is trivial on $\mathcal{A}_{g,n}$, hence $\mathrm{Des}^* w_\Theta = \mathfrak{h}$ on $\mathcal{A}_{g,n}$. Indeed, the symmetric theta structure of level n trivialises the theta multiplier bundle (Candelori, private communication). See also [Can20]. In particular this shows that $\mathcal{L}^2_{\mathcal{A}_{g,n}}$ is the Hodge bundle on $\mathcal{A}_{g,n}$, as expected from the fact that analytically products of theta constants $\theta_i(0)\theta_j(0)$ are modular forms of weight 1 for $\Gamma(n, 2n)$, as shown by the functional equation of theta constants. We could also have deduced the Lemma 2.9.8 from this functional equation, as in [Mor90].

Note that more generally, if $\pi : X_g \to \mathcal{A}_g$ is the universal principally polarised abelian stack, and $\mathcal{L}_{X_g}$ the principal polarisation, then $\pi_\star \mathcal{L}_{X_g}$ is isomorphic to $\mathfrak{h}_{\mathcal{A}_g}$ up to a μ_4-torsor [Can20, Theorem 5.0.1]. See also [Mor85, Appendice 2; FC90, Theorem 5.1], and improved bounds for the determinant bundle (when $\mathcal{L}$ is not supposed principal) in [Kou00; Pol00; MR08].

□

2.10 CHANGING LEVEL AND APPLICATION TO ISOGENIES

If $(A, \mathcal{L})$ has a symmetric theta structure (of level n even), an important question is how to obtain a symmetric theta structure on $\mathcal{L}^\ell$. For simplicity, we assume here that ℓ is prime to n (we will explain how to handle the general cases in Remarks 2.10.3, 2.10.7 and 2.10.14). The converse, going from a theta structure for $\mathcal{L}^\ell$ to a theta structure for $\mathcal{L}$ will be treated in Section 2.10.3. We will revisit these algorithms in Chapter 4 from a more general point of view, where we focus on the explicit action of the theta group (see Remark 4.2.2).

2.10.1 Raising level via an isogeny

We first digress by looking back at Example 2.8.6. Assuming we have a theta structure of level ℓn on $(A, \mathcal{L}^\ell)$, its theta null point $\tilde{0}_A$ is completely determined by the affine points $\widetilde{\pi}(\widetilde{P})$ where $\widetilde{\pi}$ is the affine lift of the isogeny $\pi : A \to B = A/K_2(\mathcal{L}^\ell)[\ell]$ and $\widetilde{P}$ are the canonical lifts of the points $P \in K_1(\mathcal{L}^\ell)[\ell]$.

Suppose that we forget about the affine lifts, and that we only have the projective coordinates $\pi(P)$ in B, for $P \in K_1(\mathcal{L}^\ell)[\ell]$. Fix arbitrary lifts of $\pi(P)$ and introduce projective factors λ_P such that $\widetilde{\pi}(\widetilde{P}) = \lambda_P \widetilde{\pi(P)}$. To reconstruct $\tilde{0}_A$ we need to recover the values of λ_P.

Fix a basis $e_1, \dots, e_g$ of $K_1(\mathcal{L}^\ell)[\ell]$.

- Using the addition law, from $\pi(e_i)$ we may recover all $\pi(P)$, $P \in K_1(\mathcal{L}^\ell)[\ell]$.
- Via the compatibility of Riemann relations with isogenies, we have the following relations on the projective factors: $\lambda_{P+Q}\lambda_{P-Q} = c\lambda_P\lambda_Q$ where c is a constant that is obtained by plugging the differential addition, and $\lambda_{P+Q+R}\lambda_P\lambda_Q\lambda_R = c'\lambda_0\lambda_{Q+R}\lambda_{P+R}\lambda_{P+Q}$ where c' is a constant that is obtained by plugging in the three way addition.
- Normalizing things such that $\lambda_0 = 1$, ie $\tilde{0}_B = \widetilde{\pi}(\tilde{0}_A)$, we get the projective factors are completely determined by $\lambda_i := \lambda_{e_i}, \lambda_{ij} := \lambda_{e_i+e_j}$. (Compare with Example 2.8.6).

But using symmetry, we can say more. Let $P' = \pi(P)$ and $\widetilde{P'} = \widetilde{\pi(P)}$. If $K' = \pi(K_1(\mathcal{L}^\ell))[\ell]$, K' is a maximal isotropic subgroup of $B[\ell]$. We say that $\widetilde{K'} = \{\widetilde{P'}\}$ is an excellent lift of K' if the points in $\widetilde{K'}$ satisfy all Riemann relations from Theorem 2.7.3 that involve only points in $\widetilde{K'}$. In particular, due to the compatibility of Riemann relations with the isogeny $\widetilde{\pi}$, if $\widetilde{K'}$ is given by the $\{\widetilde{\pi}\widetilde{P}\}$, it is an excellent lift of K'. We will soon see that all excellent lifts are of this form. But first we specialize what this definition means for a point and its multiples.

Definition 2.10.1. Let $(B, \mathcal{L})$ be a theta structure of level n, P' a point of ℓ-torsion with ℓ odd and prime to n. Write $\ell = 2\ell' + 1$. We say that an affine lift $\widetilde{P'}$ is an excellent point of ℓ-torsion if $\texttt{diff\_mult}(\ell' + 1, \widetilde{P'}) = -\texttt{diff\_mult}(\ell', \widetilde{P'})$.

We remark that if we replace $\widetilde{P'}$ by $\lambda\widetilde{P'}$, the left hand term is multiplied by $\lambda^{(\ell'+1)^2}$ while the right hand term by $\lambda^{(\ell')^2}$. So if $\widetilde{P'}$ is an excellent point of ℓ-torsion, then the other ones are exactly given by $\zeta\widetilde{P'}$, ζ a ℓ-root of unity.

We apply this to our setup above, the $\widetilde{\pi}(\widetilde{P})$ are all excellent points of ℓ-torsion. So if we want to recover the λ_i, λ_{ij}, plugging in the fact that the affine lift should be excellent lift yields equations of the form $\lambda_i^\ell = c_i, \lambda_{ij}^\ell = c_{ij}$.

Proposition 2.10.2. *Let $e'_1, \dots, e'_g$ be a basis of K'. Choose affine lifts of $\widetilde{e'_i}$, $\widetilde{e'_{i+j}}$ that are excellent points of ℓ-torsion. Use multiaddition to construct affine lifts $\widetilde{P'}$ for all $P' \in K'$. Then $\widetilde{K'}$ is an excellent lift of K', which comes from $\tilde{0}'_A$, a theta null point of level ℓn on $(A, \mathcal{L}^\ell)$.*

Proof. We know that if $\widetilde{K'}$ comes from $\tilde{0}_A$, it is an excellent lift. Since $\widetilde{e'_i}$ is an excellent point of ℓ-torsion, it differs from $\widetilde{\pi}(\widetilde{e_i})$ by a projective factor λ_i with $\lambda_i^\ell = 1$. Same for $\widetilde{e'_i + e'_j}$ and λ_{ij}. But by Remark 2.6.6 there is a symmetric automorphism of $G(\mathcal{L}^\ell)$ that acts exactly by the λ_i and λ_{ij} on $\widetilde{\pi}(\widetilde{e_i})$ and on $\widetilde{\pi}(\widetilde{e_i + e_j})$. Letting $\tilde{0}'_A$ be the image of $\tilde{0}_A$ by this automorphism, our $\widetilde{K'}$ comes from $\tilde{0}'_A$. In particular it is an excellent lift. □

The proof given here is the same proof as in [LR12], [Rob10, §7.2], slightly reformulated to be more general so that it extends readily to abelian schemes (see Section 5.2.2).

Remark 2.10.3.
(i) We may reinterpret Proposition 2.10.2 as follow: the isogeny $f : B \to B/K' = A$ is the ℓ-contragredient isogeny of $A \to B$. So we have a recipe, starting from $(B, \mathcal{M})$ of level n and a totally isotropic subgroup K' of $B[\ell]$, to construct the theta null point of level ℓn on $A = B/K'$ with polarisation given by $\pi^*(\mathcal{M})$ where π is the ℓ-contragredient isogeny $A \to B$. We will of course use this construction in Chapter 4.

(ii) In our previous example, we implicitly fixed a basis of K' on B. This fixes a basis of any symplectic complement of K' in $B[\ell]$, hence abasis of $K = f(B[\ell])$ in $A[\ell]$. The $\ell^{g(g+1)/2}$ choices of $\tilde{0}_A$ then all correspond to a choice of symplectic complement of K in $A[\ell]$.

(iii) The same ideas hold to compute the preimage by π in $(A, \mathcal{L}^\ell)$ of a point $y \in (B, \mathcal{M})$: choose an affine lift $\widetilde{y + e'_i}$ such that the $\widetilde{y + \ell e'_i}$ computed with $\texttt{diff\_multadd}$ is equal to $\tilde{y}$. This determines $\widetilde{y + e'_i}$ up to a factor λ_i with $\lambda_i^\ell = c_i$. Using $\texttt{multi\_add}$ we can compute all $\widetilde{y + P'}$, $P' \in K'$, from which we can recover the preimage $x \in (A, \mathcal{L}^\ell)$ of y. We have ℓ^g choices, as expected from the degree of the isogeny.

(iv) If ℓ is not prime to n, but still odd, we need to start with a basis of a totally isotropic subgroup K_0' of $B[n\ell]$ compatible with $B_1[n]$ (and such that $K' = K_0[\ell]$). It is easy to extend the notion of excellent lift [Rob10, Lemme 6.3.4]: we have that $(\ell'+1)P$ and $-\ell' P$ differ by a point of n-torsion, and we can use the explicit affine translation by points of n-torsion induced by the theta structure of level n on B. So we still have equations of the form $\lambda_i^\ell = c_i$.

(v) If ℓ is even, we have to write $\ell = 2\ell'$, so the equations are of the form $\lambda_i^{2\ell} = c_i$. These extra solutions also comes from automorphisms of the theta structure from Remark 2.6.6, via the symmetric automorphisms that leave the symplectic basis unchanged [Rob10, Théorème 6.3.6]. An equivalent point of view is that we may conjugate the theta structure on $\mathcal{L}^\ell$ by a point $c \in K[2]$; since c is killed by the isogeny f, the conjugate symmetric theta structure is still compatible with the one on $(B, \mathcal{M})$. This gives us the extra 2^g sign choices. From the point of view of isogenies, a difficulty is that the subgroup K_0' of $B[n\ell]$ then needs to be compatible not only with $B_1[n]$ but with the symmetric theta structure on $(B, \mathcal{M})$, hence a symplectic decomposition of $B[2n]$ which induces this symmetric theta structure.
Let us explain how to recover equations giving ℓ-roots of unity. Let K''_0 be a totally isotropic subgroup of $B[2n\ell]$ such that $2K''_0 = K_0'$. If the symmetric theta structure on $(A, \mathcal{L}^\ell)$ is induced by the decomposition $A[2n\ell] = A_1[2n\ell] \oplus A_2[2n\ell]$, we may take $K''_0 = f(A_1[2n\ell])$. The if $P''_i \in K''_0$ is of order $2n\ell$, we may compute an excellent lift $\widetilde{P''_i}$ via an equation of the form ${\lambda''_i}^{4\ell} = {c''_i}$. Then using a differential addition to get an affine lift above $P_i = 2P''_i$, the normalisation factor λ_i for P_i is $\lambda_i = {\lambda''_i}^4$, hence satisfy $\lambda_i^\ell = c_i$.

(vi) For the computation of preimages, once we have fixed a basis $P_1, \dots, P_g$ of K_0', we can define an excellent lift $\widetilde{y+P_i}$ as satisfying $\widetilde{y+\ell P_i} = g_i\tilde{x}$ where g_i is the element of the theta group above ℓP_i (which is of n-torsion). This gives equations of the form $\lambda_i^\ell = c_i$ (even if ℓ is even), hence these equations encode all preimages.

We will need the concept of **??** 2.10.3.(iii) again so we make it into a definition:

Definition 2.10.4. With the notations of Definition 2.10.1, let $\widetilde{P}$ be an excellent lift of a ℓ-torsion point $P \in A[\ell]$. Let $x \in A$. We say that $\widetilde{x+P}$ is an excellent lift of $x+P$ with respect to $\widetilde{P}$ if the affine point $\widetilde{x+\ell P}$ computed via `diff_multadd` is equal to $\tilde{x}$.

If K is an isotropic subgroup of $A[\ell]$ and we fix an excellent lift $\widetilde{K}$ of K, we may extend the definition by saying that $\widetilde{x+K} = \{\widetilde{x+P} \mid P \in K\}$ is an excellent of x with respect to $\widetilde{K}$ if these points respect all Riemann relations only involving points of $\widetilde{K}$ and the point $\widetilde{x+P}$ once in the left hand term. We may construct an excellent lift $\widetilde{x+K}$ of x with respect to $\widetilde{K}$ by first constructing excellent lifts $\widetilde{x+e_i}$ for e_i a basis of K, and then obtaining all other points by multi way additions.

2.10.2 *Raising level on the same variety*

So we know how to change the level structure via an isogeny. We now explain how to change the level by staying on the same abelian variety. Let $(A, \mathcal{L})$ be a symmetric theta structure of level n even, and let assume that ℓ is prime to n as above. Then since ℓ is odd, a symplectic basis of $A[\ell]$ (along with the theta structure on $(A, \mathcal{L})$) completely determines the symmetric theta structure of $(A, \mathcal{L}^\ell)$. Write $A[\ell] = K_1 \oplus K_2$ the induced symplectic decomposition.

By surjectivity of the multiplication map (see Section 2.8.1), generators of $\Gamma(A, \mathcal{L}^\ell)$ are given by the ℓ-fold product of the $\theta_i^{\mathcal{L}}$, $i \in K_1(\mathcal{L})$. Our Recipe 2.5.3 explains how we may reconstruct $\theta_i^{\mathcal{L}^\ell}$ from these generators provided we can explicit the action of $G(\mathcal{L}^\ell)$ on it. By the compatibility of differential addition with the change of level explained in Section 2.8.2 the answer is once again given by excellent affine lifts.

Proposition 2.10.5. *Let $A[\ell] = K_1 \oplus K_2$ be a symplectic decomposition, $x \in A$. Fix an excellent affine lift $\widetilde{K_i} = \{\widetilde{P}\}$ of K_1 and K_2, and an excellent affine lift $\widetilde{x+K_i}$ of x with respect to $\widetilde{K_i}$. Then if $g \in G(\mathcal{L}^\ell)$ is in the image of $g' \in G(\mathcal{L})$ via the embedding $\epsilon_\ell : G(\mathcal{L}) \to G(\mathcal{L}^\ell)$ described in [Mum66], $g(\prod \theta_{i_j}^{\mathcal{L}}) = \prod g' \cdot \theta_{i_j}^{\mathcal{L}}$.*

If g is the unique symmetric lift above $P \in K_1$, or $P \in K_2$ then $g(\prod \theta_{i_j}^{\mathcal{L}}(\tilde{x})) = \prod \theta_{i_j}^{\mathcal{L}}(\widetilde{x+P})$.

More generally, if $g = g_1 g_2$ where g_1 symmetric above $P \in K_1$ and g_2 symmetric above $Q \in K_2$, $g_1 g_2(\prod \theta_{i_j}^{\mathcal{L}})(\tilde{x}) = \prod(\theta_{i_j}^{\mathcal{L}})(x + \widetilde{Q} + P)$ where $x + \widetilde{Q} + P$ is computed by first normalizing $\widetilde{x+Q}$ from $\tilde{x}$ with respect to Q and then normalizing $x + \widetilde{Q} + P$ with respect to P.

Since any $g \in G(\mathcal{L})$ can be written as $g = g_0 g_1 g_2$ with each g_i as above (and g_0 commuting with g_1, g_2 since ℓ is prime to n), this suffices to describe the action of $G(\mathcal{L}^\ell)$ on the generators given by the ℓ-fold product.

Proof. This is immediate from Section 2.8.2. If g is above $P \in K_i$, g is symmetric and $g^\ell = 1$. This induces relation on the $g^n.\tilde{x}$. But by Equation (2.18) the $g^n.\tilde{x}$ can be computed via differential additions from $\tilde{x}$ and $g.\tilde{x}$. We then use the compatibility of differential addition with the change of level (described via the Segre embedding).

We remark that since an excellent lift $\widetilde{x+P}$ is defined up to a ℓ-root of unity, the choice is killed in the ℓ-fold product and hence the action does not depend on these choices, as it should. □

From the point of view of Section 2.9, we can view the choice of an excellent lift as follow: if $P \in A[\ell]$, a choice of lift $\tilde{0}_A$ and $\widetilde{P}$ correspond to trivialisations of $\mathcal{L}$ at 0_A and P respectively. Taking ℓ-th power, we have that $\mathcal{L}^\ell \simeq t_P^*\mathcal{L}^\ell$, and there is a unique isomorphism $g_P \in G(\mathcal{L}^\ell)$ compatible with the chosen trivialisations (and changing the trivialisations of $\mathcal{L}$ by a ℓ-root of unity does not change g_P). If $P \in K_i(\mathcal{L}^\ell)$, we see that $\widetilde{P}$ is an excellent lift whenever g_P is induced by a symmetric theta structure compatible with our symplectic decomposition of $A[\ell]$. The same reasoning hold to interpret excellent lifts $\widetilde{K_i}$ and $x + \widetilde{K_i}$ in terms of trivialisation of $\mathcal{L}^\ell$ at x and translates by K_i which are induced by trivialisations of $\mathcal{L}$ and a symmetric theta structure on $\mathcal{L}^\ell$ respectively.

Remark 2.10.6. We remark that if g_1 is above $P \in K_1$ and g_2 above Q in K_2, then g_1 and g_2 does not commute. This means that first computing an excellent lift $\widetilde{x+P}$ and then an excellent lift $x + \widetilde{P+Q}$ or computing an excellent lift $\widetilde{x+Q}$ and then an excellent lift $x + \widetilde{P+Q}$ does not give the same value. In fact by Proposition 2.10.5 they differ by a factor λ such that $\lambda^\ell = e_{\mathcal{L}^\ell}(P,Q)$. Applying this to $x = 0_A$ shows that fixing arbitrary lifts $\widetilde{P}$, $\widetilde{Q}$ and $\widetilde{P+Q}$ of P, Q and $P+Q$ in $(A,\mathcal{L})$, then if $\widetilde{\ell Q} = \lambda'_Q\tilde{0}_A$, $\widetilde{P+\ell Q} = \lambda_Q\widetilde{P}$, $\widetilde{\ell P} = \lambda'_P\tilde{0}_A$, $\widetilde{\ell P+Q} = \lambda_P\widetilde{P}$, $e_{\mathcal{L}^\ell}(P,Q) = \frac{\lambda_P\lambda'_Q}{\lambda'_P\lambda_Q}$.

In other words we recover the pairing $e_{\mathcal{L}^\ell}$ by working with coordinates in $(A,\mathcal{L})$! We will recover this idea in Chapter 3. The normalisation factors λ'_Q and λ'_P are just there so that the formula work even if we take non excellent affine lifts $\widetilde{P}$, $\widetilde{Q}$.

Remark 2.10.7. As in Remark 2.10.3, we may extend this to ℓ not prime to n. First a symmetric theta structure on $\mathcal{L}^\ell$ is induced by a symplectic decomposition of $A[2n\ell]$, and we want one which is compatible with a symplectic decomposition of $A[2n]$ giving the symmetric theta structure we have on $(A,\mathcal{L})$. If ℓ is prime to $2n$, it suffices to choose a symplectic decomposition of $A[\ell]$. Then we have seen how excellent lifts of $P \in A_i[\ell]$ allows to compute the action of g_P.

If ℓ is odd but not prime to n, we need a symplectic decomposition of $A[n\ell]$ compatible with the one on $A[n]$. Then as in Remark 2.10.3, if eg $P \in A_1[n\ell]$ we may extend the notion of excellent lifts $\widetilde{P}$, $\widetilde{x+P}$ by requiring that $\widetilde{\ell P}$ and $\widetilde{x+\ell P}$ are given by the corresponding action of the element of $G(\mathcal{L})$ induced by our theta structure above $\ell P \in K(\mathcal{L})$.

Finally, if ℓ is even, we need a full symplectic decomposition of $A[2n\ell]$ compatible with one on $A[2n]$ inducing the symmetric theta structure. Then as in Remark 2.10.3, taking an excellent lift of $P' \in A_1[2n\ell]$ such that $2P' = P$, allows to compute an excellent lift $\widetilde{P} = 2\widetilde{P'}$ above P defined up to a projective factor λ which satisfy $\lambda^\ell = C_P$ (rather than $\lambda^{2\ell} = C_P$ if we had normalised P directly), hence do give the explicit action of g_P on ℓ-fold products.

When looking at this Remark and Remark 2.10.3, the general slogan here is that if ℓ is prime to n we can focus on points of ℓ-torsion, if ℓ is not prime to n we need points of $n\ell$-torsion, and if ℓ is even we have equations of the form $\lambda_i^{2\ell} = C_i$ and we need to fix some information on the $2\ell n$ torsion to get back to equations of the form $\lambda_i^\ell = C_i$.

Alternatively, as in Remark 2.10.3, if we are given a symplectic basis of $A[\ell n]$ and we simply want a symmetric theta structure compatible with this basis (without fixing a prior decomposition of $A[2\ell n]$), then the equations $\lambda_i^{2\ell} = C_i$ are enough. Indeed, it suffices to make a choice of λ_i for each element of the symplectic basis e_i of $A[n\ell]$; since we act on ℓ-fold products the action only depend on λ_i^ℓ, ie a different choice only changes the action by a sign. This amount to changing the symmetric element g_{e_i} above e_i to $-g_{e_i}$. But these signs all come from a conjugation of the symmetric theta structure, see Remark 2.6.6. The 2^{2g} points of 2-torsion gives all 2^{2g} choice of signs for the g_{e_i}. The important point here is to fix these signs once and for all for a basis, we cannot fix them independently for each point $P \in A[n\ell]$, as was the case when ℓ was odd. Since we had the exact same kind of consideration in Remark 2.10.3, we may extend the slogan as: when ℓ is even, using the equations $\lambda_i^{2\ell} = C_i$ in the determination of the excellent affine lifts for each point of a basis is enough to implicitly fix the extra information of $A[2n\ell]$ we needed.

Via our Recipe 2.5.3, we can recover $\theta_0^{\mathcal{L}^\ell}$ as the trace under $\widetilde{K_2}$ of $(\theta_0^{\mathcal{L}})^\ell$ (since this function is already invariant under $K_2(\mathcal{L}^\ell)[n]$): $\theta_0^{\mathcal{L}^\ell}(x) = \sum_{\widetilde{Q}\in\widetilde{K_2}} \theta_0^{\mathcal{L}}(\widetilde{x+Q})^\ell$. Then it suffices to apply the action of g_P for $P \in K_1(\mathcal{L}^\ell)$ on this trace to recover the $\theta_i^{\mathcal{L}^\ell}$: $\theta_P^{\mathcal{L}^\ell}(\tilde{x}) = \sum_{\widetilde{Q}\in\widetilde{K_2}} \theta_0^{\mathcal{L}}(x + \widetilde{Q} + P)^\ell$ where the normalisation of $x + \widetilde{Q} + P$ is done with respect to P.

Applying Proposition 2.10.5 to $x = 0$ we may recover the theta null point of $\mathcal{L}^\ell$, and applying it to a generic point x, this gives an algorithm to express the linear change of variable between the $\theta_i^{\mathcal{L}^\ell}$ and the ℓ-fold products $\prod \theta_{i_j}^{\mathcal{L}}$. Thus we get a general multiplication formula, extending the duplication formula to the case ℓ odd. A fully general multiplication formula can then be obtained by combining both cases.

Let us record this result:

Corollary 2.10.8. *Let A/k be an abelian variety defined over a field k, represented by a theta model of even level n. Let ℓ be prime to n. Given a symplectic basis of $A[\ell]$, and the corresponding symplectic decomposition $A[\ell] = A_1[\ell] \oplus A_2[\ell]$ there is a unique symmetric theta structure of level ℓn compatible with this symplectic decomposition and the structure of level n.*

The given the coordinates of $x \in A$ in level n, we can compute the coordinates $x \in A$ of level $n\ell$ in time $O(\ell^{2g})$ as $\theta_P^{\mathcal{L}^\ell}(\tilde{x}) = \sum_{\widetilde{Q} \in \widetilde{K_2}} \theta_0^{\mathcal{L}}(x + \widetilde{Q} + P)^\ell$. In particular, we can compute the theta null point of level ℓn in time $O(\ell^{2g})$.

Proposition 2.10.5 readily yields a quasi-linear isogeny algorithm:

Corollary 2.10.9. *Let $K = K_2(\mathcal{L})[\ell]$, $\widetilde{K}$ the (unique) symmetric subgroup above K, $(B, \mathcal{M})$ the descent of $\mathcal{L}^\ell$ by $\widetilde{K}$ and $f : A \to B$ the corresponding isogeny. Let $x \in A$. Fix an excellent lift $\widetilde{K}$ of K, and $\widetilde{x + K}$ of x with respect to $\widetilde{K}$. Then identifying $K_1(\mathcal{L})$ with $K_1(\mathcal{M})$ via f, we have*

$$\theta_i^{\mathcal{M}}(f(x)) = \sum_{P \in \widetilde{K}} \theta_i^{\mathcal{L}}(\widetilde{x + P})^\ell.$$

Thus we can compute the isogeny f in time quasi-linear (in terms of arithmetic operations in $(A, \mathcal{L})$ in the size of the kernel K.

Proof. This is immediate by applying the isogeny theorem (Theorem 2.5.6) on the above construction of $\theta_i^{\mathcal{L}^\ell}$ or by applying the recipe of Recipe 2.5.8 to Proposition 2.10.5. □

It is interesting to generalize Proposition 2.10.5 as follow. Let $n_1, \dots, n_m$ be integers such that $\sum n_i^2 = \ell$. Let $F : A \to A^m, x \mapsto ([n_i]x)$. Then $F^* \mathcal{L}^{\star,m} = \mathcal{L}^{\sum n_i^2} = \mathcal{L}^\ell$. We thus obtain sections of $(A, \mathcal{L}^\ell)$ by looking at m-folds products of the form $\prod_{j=1}^m \theta_{i_j}^{\mathcal{L}} \circ n_j$. This generalises the ℓ-fold decomposition above (taking $\ell = 1 + \cdots + 1$), except that in our current setting the linear span of these sections is not necessarily the full $\Gamma(\mathcal{L}^\ell)$.

Nevertheless, we want to compute the action of $G(\mathcal{L}^\ell)$ on sections of this form. Adapting the proof of Proposition 2.10.5 yields:

Proposition 2.10.10. *Let $x \in A$. Fix an excellent affine lift $\widetilde{K_i} = \{\widetilde{P}\}$ of K_1 and K_2, and an excellent affine lift $\widetilde{x + K_i}$ of x with respect to $\widetilde{K_i}$. Using* `diff_multadd`*, these induce excellent affine lift $\widetilde{n_j x + K_i}$.*

Then if $g \in G(\mathcal{L}^\ell)$ is in the image of $g' \in G(\mathcal{L})$ via the embedding $\epsilon_\ell : G(\mathcal{L}) \to G(\mathcal{L}^\ell)$ described in [Mum66], $g(\prod_{j=1}^m \theta_{i_j}^{\mathcal{L}})(n_j \tilde{x}) = \prod_{j=1}^m g'(\theta_{i_j}^{\mathcal{L}})(n_j \tilde{x})$.

If g is the unique symmetric lift above $P \in K_1$, or $P \in K_2$ then $g(\prod_{j=1}^m \theta_{i_j}^{\mathcal{L}})(n_j \tilde{x}) = \prod_{j=1}^m (\theta_{i_j}^{\mathcal{L}})(\widetilde{n_j x + P})$.

More generally, if $g = g_1 g_2$ where g_1 symmetric above $P \in K_1$ and g_2 symmetric above $Q \in K_2$, $g_1 g_2(\prod_{j=1}^m \theta_{i_j}^{\mathcal{L}})(n_j \tilde{x}) = \prod_{j=1}^m (\theta_{i_j}^{\mathcal{L}})(n_j x + \widetilde{Q} + P)$ where $n_j x + \widetilde{Q} + P$ is computed by first normalizing $\widetilde{x + Q}$ from $\tilde{x}$ with respect to Q and then normalizing $x + \widetilde{Q} + P$ with respect to P and then computing $n_j x + \widetilde{Q} + P$.

While sections of the form above may not span the full $\Gamma(\mathcal{L}^\ell)$, by irreducibility of the action of $G(\mathcal{L}^\ell)$ we can use Proposition 2.10.10 on any of these sections to get all sections. In particular, this is enough to give another quasi-linear way to compute isogenies by applying our recipe from Recipe 2.5.8:

Corollary 2.10.11. *Let $K = K_2(\mathcal{L})[\ell]$, $\widetilde{K}$ the (unique) symmetric subgroup above K, $(B, \mathcal{M})$ the descent of $\mathcal{L}^\ell$ by $\widetilde{K}$ and $f : A \to B$ the corresponding isogeny. Let $x \in A$. Fix an excellent lift $\widetilde{K}$ of K, and $\widetilde{x + K}$ of x with respect to $\widetilde{K}$, and let $\widetilde{n_i x + K}$ the induced lift. Then identifying $K_1(\mathcal{L})$ with $K_1(\mathcal{M})$ via f, we have*

$$\theta_i^{\mathcal{M}}(f(x)) = \sum_{P \in \widetilde{K}} \prod_{j=1}^m \theta_i^{\mathcal{L}}(\widetilde{n_j x + P}).$$

2.10.3 Descending level

If we have a (symmetric) theta structure of level ℓn on $(A, \mathcal{L}^\ell)$, it canonically induces a (symmetric) theta structure of level n on $(A, \mathcal{L})$. Hence it is natural to ask for a formula to descend level.

If $\ell = m^2$, $\mathcal{L}^\ell \simeq [m]^*\mathcal{L}$ and $A[m]$ is isotropic in $A[\ell]$ so it suffices to apply the isogeny formula from Theorem 2.5.6. In the general case, we can always write $\ell = \sum_{i=1}^r n_i^2$ (a sum of two or four squares), and construct an integer matrix F of size r such that $t_F F = \ell\,\mathrm{Id}$ (using the multiplication matrix of $n_1 + n_2 i + n_3 j + n_4 k$ in the quaternions or the complex numbers). Then we can apply Theorem 2.7.1 to descend $(A^r, \mathcal{L}^\ell \star \cdots \star \mathcal{L}^\ell)$ to $(A^r, \mathcal{L} \star \cdots \star \mathcal{L})$.

This relates r-fold product of theta functions of level n with the r-fold product of theta functions of level $n\ell$, and was used in [CR15]. One problem with this formula is that we take the trace of the functions of level $n\ell$ under the action of (the lift of) the kernel of F in $A_2[\ell]^r$, for a total cost of $O(\ell^{gr/2})$. In particular if $r = 4$, we get a quadratic algorithm $O(\ell^{2g})$.

Using the (generalised) Segre morphism $F : A \to A^r, x \mapsto ([n_i]x)$ instead, since $F^*(\mathcal{L}^\ell \star \cdots \star \mathcal{L}^\ell) = \mathcal{L}^{\ell^2}$, $F(A[\ell])$ is isotropic in $(A^r, \mathcal{L}^\ell \star \cdots \star \mathcal{L}^\ell)$, so we can descend it using the isogeny theorem.

Theorem 2.10.12. *Let $(A, \mathcal{L}^\ell)$ be an abelian variety with a symmetric theta level structure of level $n\ell$. Let $n_1, \dots, n_m$ be integers such that $\ell = \sum n_i^2$. Let $x \in A$ and fix an arbitrary affine lift $\tilde{x}$.*

Then for $i \in A_1[n]$, $\theta_i^{\mathcal{L}}(\tilde{x}) = \sum_{t \in A_1[\ell]} \prod_{j=1}^m \theta_{i+t}^{\mathcal{L}^\ell}(\widetilde{n_j x})$, where $\widetilde{n_j x}$ is computed via `diff_mult`.

The complexity to descend the theta structure is thus $O(\ell^g)$ arithmetic operations in A.

Proof. It suffices to combine the isogeny formula Theorem 2.5.6 with the generalised Segre morphism. Indeed since $A[\ell]$ is isotropic for $(\mathcal{L}^\ell)^\ell$, $F(A[\ell])$ is isotropic for $(A^r, (\mathcal{L}^\ell)^{n_1} \star \cdots \star (\mathcal{L}^\ell)^{n_r})$. Hence we may apply the isogeny formula to compute $A^m/F(A[\ell])$, and recover $(A, \mathcal{L})$ as the descent of $F(A)$ by this isogeny. □

Combining Proposition 2.10.2 and Theorem 2.10.12, we find the same quasi-linear algorithm to compute isogenies as in Corollary 2.10.11:

Corollary 2.10.13. *Let $K = K_2(\mathcal{L}^\ell)[\ell]$, $\widetilde{K}$ the (unique) symmetric subgroup above K, $(B, \mathcal{M})$ the descent of $\mathcal{L}^\ell$ by $\widetilde{K}$ and $f : A \to B$ the corresponding isogeny. Let $x \in A$. Fix an excellent lift $\widetilde{K}$ of K, and $\widetilde{x + K}$ of x with respect to $\widetilde{K}$. Using* `diff_multadd`*, this fixes excellent lifts of $n_i\widetilde{x + K}$. Then identifying $K_1(\mathcal{L})$ with $K_1(\mathcal{M})$ via f, we have*

$$\theta_i^{\mathcal{M}}(f(x)) = \sum_{P \in \widetilde{K}} \prod_{j=1}^m \theta_i^{\mathcal{L}}(n_j\widetilde{x + P}).$$

We give more details about this formula in Section 4.4.3.

Remark 2.10.14. We can hold the same reasoning as in Remarks 2.10.3 and 2.10.7 when ℓ is not prime to n. Let us assume that our symplectic theta structure on $(A, \mathcal{L})$ is induced by a given symplectic decomposition of $A[2n]$. If ℓ is prime to $2n$, f is injective on $A[2n]$, hence the symplectic decomposition descends to $B[2n]$, hence induces a (unique) symmetric theta structure on $(B, \mathcal{M})$.

If ℓ is not prime to $2n$, part of the information on the $2n$-torsion is lost by the isogeny, in other words there are several compatible choices on B. So we need to fix a totally isotropic subgroup K'' of $A[2n\ell]$ such that $K = 2nK''$ and $\ell K'' = A_2[2n]$. Then we get a symplectic decomposition of $B[2n]$ by pushing $A_1[2n] \oplus K''$ via f. Of course if ℓ is odd, it suffices to choose K' totally isotropic in $A[n\ell]$ such that $K = nK'$ and $\ell K' = A_2[n]$, exactly as our slogan of Remark 2.10.7 dictated.

Even if ℓ is even, we can just fix a K' (with the caveat in this case that we need to be careful it is compatible with a decomposition of $A[2n]$ inducing the theta structure on $\mathcal{L}$, ie $[\ell/2]K'' = A_2[2n]$) and then use the equations $\lambda_i^{2\ell} = C_i$ on a given basis of K' to get all possible symmetric theta structure on B compatible with our choice of K'. See the extended slogan of Remark 2.10.7. Indeed, if we have a symmetric theta structure on $(A, \mathcal{L}^\ell)$ compatible with the one on $(A, \mathcal{L})$, then conjugating by a point of 2-torsion c does not change the compatibility (since this conjugates the theta structure on $(A, \mathcal{L})$ by ℓc and ℓ is even), but this changes the induced symmetric theta structure on $(B, \mathcal{M})$ if $c \notin K$. This gives us the 2^g choice of signs.

2.11 RATIONALITY

If $(A, \mathcal{L})$ is an abelian variety defined over a field k, we may ask for conditions on whether there exists a rational symmetric theta structure on it.

Of course there must exists a theta structure over $\overline{k}$ (assume k perfect for simplicity, or use k_s instead and stick with levels prime to the characteristic). This means, if $\mathcal{L}$ is of level n, that $K(\mathcal{L})$ has to be isomorphic as a Galois module to $(\mathbb{Z}/n\mathbb{Z})^g \times \mu_n^g$. So if the level is not prime to p, a theta structure may only exists on an ordinary abelian variety. For a symmetric theta structure, we have seen in Section 2.6 that there is an obstruction to its existence, measured by the value of $e_*^{\mathcal{L}}$ on $K(\mathcal{L})[2]$.

In general, even if it exists, a theta structure on $\overline{k}$ may not descend on k, because $G(\mathcal{L})$ is only a twist (and not isomorphic to) the Heisenberg group $H(n)$ over k. It would be interesting to work out these twists (eg if k is a finite field), and see how to twist the Riemann relations from Sections 2.7 and 2.8 to have the results of Section 2.10 apply over the base field and not only an extension.

We need two conditions for a theta structure $\Theta_{\mathcal{L}}$ to be rational:

- The induced isomorphism $H(n) \to K(\mathcal{L})$ has to be rational (ie Galois equivariant). Equivalently, since $e_{\mathcal{L}}$ is Galois equivariant, it suffices that $A[n]$ contains a maximal totally isotropic subgroup K whose geometric points are rational. (But not all points of $A[n]$ need to be rational if $\mu_n \not\subset k^*$.)
- The level subgroups $\widetilde{K}_i(\mathcal{L})$ have to be rational.

Let us focus on the second condition in the symmetric case (assuming each polarisation is separable for simplicity). Let $x \in K(\mathcal{L})$, and y a point such that $x = 2y$. Then $y \in K(\mathcal{L}^2)$. Let $\pm g_y \in G(\mathcal{L}^2)$ be one of the two symmetric elements above y, then $g = \eta_2(\pm g_y)$ does not depend on the choice of g_y (using the notations of Section 2.6.3 and **??** 2.8.7.(vi)). Hence a symmetric element above x is canonically determined by the choice of y. It remains to see how the g above depends on y.

Lemma 2.11.1. *Let $\mathcal{L}$ be totally symmetric. Let $y \in K(\mathcal{L}^2)$, $x = 2y$, and write $g = \delta_2(\pm g_y)$ the symmetric element above x, where g_y is a symmetric element above y. Let $y + t$ where $t \in A[2](\overline{k})$ be another preimage of x by $[2]$, and let g_t be the induced symmetric element above x. Then $g_t = g e_{\mathcal{L}}(x, t)$.*

Proof. Since $\mathcal{L}$ is totally symmetric, there is a symmetric theta structure on $(\mathcal{L}, \mathcal{L}^2)$, so we can check this on the level of the Heisenberg groups. Let $g_y = (\alpha, y_1, y_2) \in H(2n)$, it is symmetric when $\alpha = \pm\langle y_1, y_2\rangle^{1/2}$, and $\eta_2(g_y) = (\alpha^2, x_1, x_2)$ with $\alpha^2 = \langle y_1, y_2\rangle$. Taking $(y_1 + t_1, y_2 + t_2)$ instead, we get that $\eta_2(g_t) = \gamma\eta_2(g_y)$ with $\gamma = \langle y_1, t_2\rangle\langle t_1, y_2\rangle\langle t_1, t_2\rangle$. Since t is a point of 2-torsion and $2n$ is divisible by 4, we have $\langle t_1, t_2\rangle = 1$. So we find $\gamma = e_{2n}(y, t) = e_n(x, t)$, as required. □

Corollary 2.11.2. *If $\mathcal{L}$ is totally symmetric, a symmetric theta structure $\Theta_{\mathcal{L}}$ on $\mathcal{L}$ is completely determined by a symplectic basis $f_1, \dots, f_g, f_1', \dots, f_g'$ of $K(\mathcal{L}^2)$.*

Another symplectic basis yields the same $\Theta_{\mathcal{L}}$ is and only if

- *the induced symplectic basis $e_i = 2f_i, e_i' = 2f_i'$ of $K(\mathcal{L})$ is the same, ie the new basis is of the form $f_i + t_i, f_i' + t_i'$ with the $t_i, t_i' \in A[2]$;*
- *$e_{\mathcal{L}}(e_i, t_i) = 1, e_{\mathcal{L}}(e_i', t_i') = 1$.*

In particular, $\Theta_{\mathcal{L}}$ is completely determined by a symplectic decomposition $K(\mathcal{L}^2) = K_1(\mathcal{L}^2) \oplus K_2(\mathcal{L}^2)$ and a choice of basis of the induced $K_1(\mathcal{L})$.

Remark 2.11.3. Lemma 2.11.1 also shows that if a symmetric theta structure on $\mathcal{L}$ is induced by a symplectic basis $f_1, \dots, f_g, f_1', \dots, f_g'$, then the action of conjugation by a point $c \in A[2]$ is induced by the symplectic basis $f_i + t_i, f_i' + t_i'$ where if $c = n\left(\sum \epsilon_i f_i + \sum \epsilon_i' f_i'\right)$ is the decomposition of c, $t_i = n\epsilon_i' f_i'$ and $t_i' = n\epsilon_i f_i$. In other words, $t_i = 0$ if $e_{\mathcal{L}^2}(f_i, c) = 1$ and $t_i = nf_i'$ if $e_{\mathcal{L}^2}(f_i, c) = -1$.

In particular, if $f : A \to B$ is an isogeny, Corollary 2.11.2 yields a convenient way to check if two symmetric theta structure on $(A, \mathcal{L})$ and $(B, \mathcal{M})$ are compatible with respect to f: there should be a symplectic decomposition of $K(\mathcal{L}^2)$ which induces $\Theta_{\mathcal{L}}$ on A and $\Theta_{\mathcal{M}}$ (via f) on B, see Lemma 2.6.9.

Corollary 2.11.4. *Let $(A, \mathcal{L})/\mathbb{F}_q$ be a separably polarised abelian variety of even level $n = 2n_0$ over the finite field $\mathbb{F}_q$. Then there exist a rational symmetric theta structure on $\mathcal{L}$ if and only if there exist a symplectic basis $(e_1, \dots e_g, e_1', \dots, e'g)$ with $e_1, \dots, e_g$ rational and such that $e_{T,2}(n_0 e_i, e_i) = 1$ and $e_{T,2}(n_0 e_i', e_i') = 1$ where $e_{T,2}$ denotes the 2-Tate pairing.*

In particular, if $\mu_n \subset \mathbb{F}_q$, this is equivalent to: e_i, e_i' form a rational symplectic basis consisting of elements whose self n-Tate pairing $e_{T,n}(e_i, e_i), e_{T,n}(e_i', e_i')$ is not a primitive n-th root of unity.

Proof. This is clear from Corollary 2.11.2 and the definition of the Tate pairing as $e_{T,2}(n_0 e_i, e_i) = e_{W,2}(n_0 e_i, \pi_q(f_i) - f_i)$ where $e_{W,2}$ is the Weil pairing (or more precisely, if $\mathcal{L} = \mathcal{L}_0^n$ the Weil pairing associated to the principal polarisation $\mathcal{L}_0$, ie $e_{W,2} = e_{\mathcal{L}_0^2}$), where $2f_i = e_i$ and π_q is the Frobenius of $\mathbb{F}_q$. Indeed, $e_{\mathcal{L}}(e_i, \pi_q(f_i) - f_i) = e_{\mathcal{L}_0^2}(n_0 e_i, \pi_q(f_i) - f_i)$, see Chapter 3. □

2.12 ARITHMETIC ON KUMMER VARIETIES

We briefly discuss arithmetic on Kummer varieties, which is the focus of the article [LR16]. We recall that if A is an abelian variety, we define the Kummer variety as $K_A = A/\pm 1$ (beware that some authors define the Kummer variety as the variety $A/\operatorname{Aut}(A)$, thus a quotient of our Kummer; in the generic case $\operatorname{Aut}(A) = \pm 1$ so the two definitions agree.)

There are several related questions:

- What kind of arithmetic is available on a Kummer variety? The addition law does not descend, but it is well known that the multiplication $P \mapsto n.P$ does.
- Using theta functions, the Kummer variety is described by a symmetric theta structure of level $n = 2$ (more precisely, if $\mathcal{L} = \mathcal{L}_{A,1}^n$ with $\mathcal{L}_{A,1}$ principal and $n = 2$, then if $\mathcal{L}_{A,1}$ splits as $(A, \mathcal{L}_{A,1}) = \prod (A, \mathcal{L}_{A,1_i})$ then $\mathcal{L}$ gives an embedding of $\prod K_{A_i}$). Riemann relations are still valid in this case, but not sufficient to give equations for the Kummer nor an addition law. We have seen in Section 2.8 the algorithmic usefulness of the affine version of the addition laws given by Riemann relations. How does this transposes to Kummer varieties?
- How to go back and forth between the abelian variety and the Kummer variety? If $\mathcal{L}_{A,1}$ is a principal polarisation, the Kummer is described by $\mathcal{L}_{A,1}^2$ while the abelian variety by $\mathcal{L}_{A,1}^n$ with $n \geq 3$. In particular the duplication formula from Corollary 2.7.2 relates sections of $\mathcal{L}_{A,1}^4$ with sections of $\mathcal{L}_{A,1}^2$. But $\Gamma(A, \mathcal{L}_{A,1}^4)$ is of dimension 4^g while $\Gamma(A, \mathcal{L}_{A,1}^2)$ of dimension 2^g. Do we really need that many extra functions just to encode a choice of sign?

2.12.1 *Arithmetic of Kummer groups*

In [LR16] we developed the arithmetic of Kummer varieties. But in fact this extends to a general commutative group scheme G, and we used this in [LR20b] to get arithmetic information on the tangent cone of a Kummer variety at points of 2-torsion x. Indeed the tangent cone of K_A at x is isomorphic to $T_x A/\pm 1$, so the arithmetic of the tangent space descends to the tangent cone.

Let G/k be a commutative group scheme, and $K = G/\pm 1$. Let $\pi : G \to K$ the projection. We are interested in what kind of the arithmetic of G descends to K. We denote by P, Q points of G, and $[P], [Q]$ their image in K.

The key definition in [LR16, Definition 2.1] is:

Definition 2.12.1. Given a model of K defined over the field k, a *schematic addition* is an algorithm which provided with two points $[P], [Q] \in K(k)$, outputs equations defining the dimension 0 scheme of degree two $\mathcal{S} = \{[P+Q], [P-Q]\}$. More precisely this algorithm should output a rational parametrisation of $\mathcal{S}$, that is a polynomial $\mathcal{P} \in k[t]$ of degree 2 in one variable, and a rational isomorphism $f : \operatorname{Spec} k[t]/\mathcal{P}(t) \to \mathcal{S}$ together with its inverse f^{-1}.

More generally we may ask for an explicit schematic addition on K not only for k-points but for general R points, R a k-algebra.

Using schematic additions, we get the following arithmetic operations on K:

- If $[T]$ is a point of 2-torsion on K, then the schematic addition of $[P]$ and $[T]$ is a double point, so schematic additions allows to compute translation by points of $[2]$-torsion.
- Given $[P], [Q], [P-Q]$ we can compute $[P+Q]$ as the point of $\mathcal{S}$ which is not $[P-Q]$. So we recover differential additions from schematic additions. (One should be careful not to conflate this differential addition, which on a Kummer variety would be a differential addition on projective coordinates, with the differential addition we have defined for affine lifts in Section 2.8 and which we will see for Kummer below).

From differential additions it is easy to derive a scalar multiplication $[P] \mapsto n[P]$, using an addition chain. A standard method is to use a double and add algorithm, keeping $(m[P], (m+1)[P])$ at each step and computing $(2m[P], (2m+1)[P])$ or $((2m+1)[P], 2(m+1)[P])$ via a doubling and a differential addition according to whether our current bit is 0 or 1. Likewise we can compute multiscalar multiplications $n[P] + m[Q]$ given $[P], [Q], [P+Q]$, or more generally $\sum n_i[P_i]$ given all $\sum \epsilon_i[P_i]$, $\epsilon_i \in \{0, 1\}$.

- Let $P, Q, R, S \in G(k)$ be such that $P + Q = R + S$ and $P - Q \neq R - S$, $P - Q \neq S - R$. Then the point $[P+Q] = [R+S]$ of K is well defined from the knowledge of $[P], [Q], [R], [S]$ and can be computed as the intersection of the output of the two schematic additions: $\{[P+Q], [P-Q]\} \cap \{[R+S], [R-S]\}$. This is called a *compatible addition* in [LR16, § 2.1].

  In practice the two schematic additions will output two degree two polynomials $P_1 = X^2 + aX + b$ and $P_2 = X^2 + cX + d$ in $k[X]$ parametrizing the two schemes $\{[P+Q], [P-Q]\}$ and $\{[R+S], [R-S]\}$. Then P_1 and P_2 have a common root if and only if $(ad - bc)(c - a) = (d - b)^2$ and in this case this root is $(d - b)/(a - c)$.

  Applications of compatible addition to multiscalar multiplication is given in [LR16, § 2.3].

- If $[P_0] \in K(k)$ is not a point of 2-torsion, then from $[P_1], \dots, [P_n] \in K(k)$ and $[P_0 + P_1], \dots, [P_0 + P_n] \in K(k)$, one can compute the *multi way additions* $[P_1 + \cdots + P_n]$ and $[P_0 + P_1 + \cdots + P_n]$ using $2(n-1)$ compatible additions [LR16, Corollary 2.9].

  The idea behind multi-way additions is that giving the points $[P_0 + P_i]$ on K "fixes" the sign of P_i relatively to P_0, so that we can compute $[P_1 + \cdots + P_n]$ and $[P_0 + P_1 + \cdots + P_n]$. So in particular, to compute the multiscalar multiplication $\sum n_i P_i$ we only need the data of $[P_0 + P_i]$.

- If $[P_0] \in K(k)$ is not a point of 2-torsion, we have an injection $\alpha_{P_0} : G(k) \to K(k) \times K(k), P \mapsto ([P], [P + P_0])$. It is easy to check if a tuple $([P], [Q])$ lies in the image of α_{P_0} (just check if $[Q]$ is in $\{[P + P_0], [P - P_0]\}$, and to do arithmetic in this model of G.

  Note that P_1 is another point not of 2-torsion, it is easy to switch from the model given by α_{P_0} to the model α_{P_1} if we are given the point $[P_0 + P_1] \in K_A$, in particular if we are given the coordinates of P_1 in the α_{P_0} model. (Otherwise if we only have $[P_1] \in K_A$ we need to make a schematic addition and choose a root, hence get either $[P_0 + P_1]$ or $[P_0 - P_1]$.)

  We remark that we do not need to represent $[P + P_0]$ fully, we just need the root x of the polynomial $P(X)$ representing $\{[P + P_0], [P - P_0]\}$ corresponding to $[P + P_0]$. Furthermore, while the standard addition on this model is quite slow (since it requires two compatible additions and a differential addition, and a compatible addition requires two schematic addition), scalar multiplication in G can be done using differential additions in K, to compute $([nP], [nP + P_0])$. In fact, we just need to compute $[(n-1)P], [nP]$ in K and recover $[nP + P_0]$ at the end as a compatible addition $nP + P_0 = (n-1)P + (P + P_0)$.

  So if K has an efficient model, we get a rather efficient model of G using just one extra coordinate. We note that, if G is smooth, α_{P_0} is injective not only on k-points but also on $k[\epsilon]$-points, ie on tangent vectors, so if we have a projective embedding of K in $\mathbb{P}^n$, α_{P_0} induce a projective embedding of G in $\mathbb{P}^n \times \mathbb{P}^n$ (once again we see the usefulness of the Segre embedding).

The above very simple idea shows that we can compute *some* additions on $\mathcal{K}_A$.

2.12.2 Riemann relations in the theta model of level 2

We now specialize the above results to the model of K_A given by a symmetric theta structure of level $n = 2$. Looking back at Section 2.8.1, we see that we may recover $(\underbrace{\sum_{t \in Z(\bar{2})} \chi(t)\theta_{i+t}(\tilde{x}_1)\theta_{j+t}(\tilde{y}_1)}_{t \in Z(\bar{2})})$ from the other coordinates in Equation (2.10) provided that we can find k, l such that $U^{\mathcal{L}^2}_{\chi,k}(\tilde{u}) U^{\mathcal{L}^2}_{\chi,j}(\tilde{v}) \neq 0$. But as we saw, this is related to the surjectivity of the multiplication map. Here we need the following refinement of the Mumford-Koizumi-Kempf theorem due to Kempf in [Kem88].

Theorem 2.12.2 (Mumford-Koizumi-Kempf). *Let A be an abelian variety, $\mathcal{P}$ the Poincare line bundle on $A \times \widehat{A}$, $\mathcal{P}_x$ the pullback of $\mathcal{P}$ by $\mathrm{Id} \times x$ for $x \in \widehat{A}$ (ie the line bundle on A represented by $x \in \widehat{A} \simeq \mathrm{Pic}^0(A)$). Let α be the multiplication map*

$$\alpha : \Gamma(A, \mathcal{L}^n \otimes \mathcal{P}_x) \otimes \Gamma(A, \mathcal{L}^m \otimes \mathcal{P}_y) \to \Gamma(A, \mathcal{L}^{n+m} \otimes \mathcal{P}_{x+y}).$$

Then:

- *If $n \geq 2$ and $m \geq 3$, α is surjective for all $x, y \in \widehat{A}$;*
- *If $n = m = 2$, then fixing any $u \in \widehat{A}$, α is surjective for $x = -v$ and $y = u + v$ for v in an open dense dense subset of $\widehat{A}$.*
- *If $p \neq 2$, $\mathcal{L}$ is principal and is represented by a symmetric divisor Θ, the rank of*

$$\alpha : \Gamma(A, 2(\Theta + x)) \otimes \Gamma(A, 2(\Theta + y)) \to \Gamma(A, 4(\Theta + z)),$$

where $x + y = 2z$ is given by the number of points of 2-torsion $t \in A[2](\bar{k})$ such that $z + t \notin \Theta + x$.

From Theorem 2.12.2 we get that we may generically compute Riemann relations on K_A. However, differential additions involve the non annulation of the $U^{\mathcal{L}^2}_{\chi,k}(0)U^{\mathcal{L}^2}_{\chi,j}(0)$. But $U_{\chi,i}(-z) = \chi(2i)U_{\chi,i}(z)$, so for the odd theta functions $U^{\mathcal{L}^2}_{\chi,i}$ we have $U^{\mathcal{L}^2}_{\chi,j}(0) = 0$ (recall that analytically the $U^{\mathcal{L}^2}_{\chi,i}$ corresponds to the $\theta\left[\begin{smallmatrix} a/2 \\ b/2 \end{smallmatrix}\right](2z, \Omega)$).

This is reflected by the fact that the multiplication map $\Gamma(A, \mathcal{L}) \otimes \Gamma(A, \mathcal{L}) \to \Gamma(A, \mathcal{L}^2)$ cannot be surjective since $\Gamma(A, \mathcal{L})$ consists only of even functions, so the image by the multiplication map consists of even functions.

Let us study this map in more details. Since we have a symmetric theta structure on $\mathcal{L}$, $\mathcal{L}$ has to be totally symmetric, so it descends to the Kummer variety K_A. Decomposing $\Gamma(A, \mathcal{L}^m)$ into even and odd functions $\Gamma(A, \mathcal{L}^m) = \Gamma(A, \mathcal{L}^m)^+ \oplus \Gamma(A, \mathcal{L}^m)^-$, we have that $\Gamma(K_A, \mathcal{L}^m) = \Gamma(A, \mathcal{L}^m)^+$. So since $\Gamma(A, \mathcal{L}) = \Gamma(A, \mathcal{L})^+$, the map $\Gamma(K_A, \mathcal{L}) \otimes \Gamma(K_A, \mathcal{L}) \to \Gamma(K_A, \mathcal{L}^2)$ is surjective precisely when $\Gamma(A, \mathcal{L}) \otimes \Gamma(A, \mathcal{L}) \to \Gamma(A, \mathcal{L}^2)^+$ is surjective. We remark that by Theorem 2.12.2, $\Gamma(A, \mathcal{L}^m)^\pm \otimes \Gamma(A, \mathcal{L}) \to \Gamma(A, \mathcal{L}^{m+1})^\pm$ is surjective whenever $m \geq 2$, so it is the map for $m = 1$ that may pose a problem.

Let us first review standard properties of line bundles related to the multiplication map:

- If $\mathcal{L}$ is ample, $\Gamma(X, \mathcal{L}^n) \times \Gamma(X, \mathcal{L}^m) \to \Gamma(X, \mathcal{L}^{n+m})$ is surjective for $n, m \gg 0$.
- If $\mathcal{L}$ is ample, $\mathcal{L}$ is very ample if and only if $S^n\Gamma(X, \mathcal{L}) \to \Gamma(X, \mathcal{L}^n)$ is surjective for $n \gg 0$; this implies that $\Gamma(X, \mathcal{L}^n) \times \Gamma(X, \mathcal{L}) \to \Gamma(X, \mathcal{L}^{n+1})$ is surjective for $n \gg 0$.
- If $\mathcal{L}$ is ample and $\Gamma(X, \mathcal{L}^n) \times \Gamma(X, \mathcal{L}) \to \Gamma(X, \mathcal{L}^{n+1})$ is surjective for all $n > 0$ (equivalently $\Gamma(X, \mathcal{L}^n) \times \Gamma(X, \mathcal{L}^m) \to \Gamma(X, \mathcal{L}^{n+m})$ is surjective for all $n, m > 0$ or $S^n\Gamma(X, \mathcal{L}) \to \Gamma(X, \mathcal{L}^n)$ is surjective for all $n > 0$), then $\mathcal{L}$ is said to be projectively normal (Mumford calls such a $\mathcal{L}$ normally generated in [Mum69]). The terminology comes from the fact that $\mathcal{L}$ is projectively normal exactly when the homogeneous ring $S(X)$ induced by the projective embedding given by the very ample $\mathcal{L}$ is equal to the global ring of sections $\tilde{S}(X) = \oplus\Gamma(X, \mathcal{L}^n)$, which is normal if X is normal (eg smooth).

For all this and much more see [Mum69], which contains much more precise results also on the kernel of the multiplication map using Casterlnuovo-Mumford regularity.

Going back to our Kummer, using Theorem 2.12.2 we get that the rank of $\Gamma(A, \mathcal{L}) \otimes \Gamma(A, \mathcal{L}) \to \Gamma(A, \mathcal{L}^2)^+$ is the number of non zero theta null $U^{\mathcal{L}^2}_{\chi,i}(0) \neq 0$ (which we will call the even theta-null werte).

Corollary 2.12.3. *The following are equivalent::*

- *$\mathcal{L}$ is projectively normal on K_A;*
- *$\Gamma(A, \mathcal{L}) \otimes \Gamma(A, \mathcal{L}) \to \Gamma(A, \mathcal{L}^2)^+$ is surjective;*
- *The even theta-null werte are all different from zero.*

See also [LR16, § 3].

This will be our running assumption in the following. Note that this excludes both decomposable polarisations or Jacobians of hyperelliptic curves of genus $g \geq 3$ since they are both characterised by the cancellation of suitable theta-nulls [Mum84].

Proposition 2.12.4. *If the even theta-null werte are non zero, in the differential addition formula from $\widetilde{P}, \widetilde{Q}$ we can recover all the $\theta_i(\widetilde{P+Q})\theta_j(\widetilde{P-Q}) + \theta_j(\widetilde{P+Q})\theta_i(\widetilde{P-Q})$.*

Proof. By our assumptions, we can recover all $(\sum_{t \in Z(\bar{2})} \chi(t)\theta_{i+t}(\widetilde{P+Q})\theta_{j+t}(\widetilde{P-Q}))$ for χ such that $\chi(i+j) = 1$, and summing over these characters yield the result, see [Rob10, Proposition 4.8.6]. □

Letting $\kappa_{ij} = \theta_i(\widetilde{P+Q})\theta_j(\widetilde{P-Q}) + \theta_j(\widetilde{P+Q})\theta_i(\widetilde{P-Q})$, it is not surprising that we can recover the values of the κ_{ij} since they are invariant under the action of $[-1]$. It is also clear that they allow to compute the schematic addition of Section 2.12.1. Indeed, in the 2×2^g matrix $(\theta_i(\widetilde{P+Q}), \theta_i(\widetilde{P-Q}))_{i\in Z(\overline{n})}$ take two lines i, j of rank 2. Any other line $\theta_k(\widetilde{P}), \theta_k(\widetilde{Q})$ is recovered from a linear equation involving κ_{ik}, κ_{jk}. We may encode $\{\theta_i(\widetilde{P+Q})\theta_j(\widetilde{P-Q}), \theta_j(\widetilde{P+Q})\theta_i(\widetilde{P-Q})\}$ as the roots of the polynomial equation $X^2 - \kappa_{ij}X + \kappa_{ii}\kappa_{jj}/4$. This gives an algorithm to compute the schematic addition on K_A. See [Rob10, Algorithme 4.8.7] and [LR15a, § 3.3] for more details and the algorithm.

Notice that Proposition 2.12.4 not only gives a projective schematic addition but also an affine schematic addition, with the caveat that the κ_{ij} are homogeneous of degree 2 so the affine schematic addition does not distinguish between $\{\lambda\widetilde{P+Q}, \lambda\widetilde{P-Q}\}$ where $\lambda = \pm 1$. Hence all the arithmetic developed in Section 2.12.1 also exists on the affine lifts (with the caveat above).

In particular, we already knew that we could compute three way additions generically from Theorem 2.12.2. But if $\mathcal{L}$ is projectively normal, we get that we can always compute three way additions (up to a sign) [LR16, Proposition 3.7].

2.12.3 *From level 2 to level 4*

Given a symmetric theta structure of level $n = 2$ on $(A, \mathcal{L})$ and the theta null point of level 4 of a compatible symmetric theta structure on $(A, \mathcal{L}^2)$, we explain how to switch back and forth between the two models.

First it is easy, using the duplication formula from Corollary 2.7.2 to go from the $\theta_i^{\mathcal{L}^2}$ to the $\theta_i^{\mathcal{L}}$:

$$\theta_{i+j}^{\mathcal{L}}(P)\theta_{i-j}^{\mathcal{L}}(P) = \frac{1}{2^g}\sum_{t\in Z(\overline{2})}\theta_{i+t}^{\mathcal{L}^2}(P)\theta_{j+t}^{\mathcal{L}^2}(0).$$

We can then invert the duplication formula using:

$$U_{\chi,i}^{\mathcal{L}^2}(P)U_{\chi,j}^{\mathcal{L}^2}(0) = \sum_{t\in Z(\overline{2})}\chi(t)\theta_{i+j+t}^{\mathcal{L}}(P)\theta_{i-j+t}^{\mathcal{L}}(P).$$

Since the odd theta null values are null, we only recover the coordinates $U_{\chi,i}^{\mathcal{L}^2}$ such that $\chi(2i) = 1$.

But in level 4 we have the explicit translation of points of four torsion, so it is also easy to go from the model of $(A, \mathcal{L}^2)$ to the model $\alpha_T : A \to (K_A, \mathcal{L}) \times (K_A, \mathcal{L}), \theta_i^{\mathcal{L}^2}(P) \mapsto (\theta_i^{\mathcal{L}}(P), \theta_i^{\mathcal{L}}(P+T))$, for T four-torsion.

Going from $[P] \in (K_A, \mathcal{L})$ to $\{[P+T], [-P+T]\}$ requires a schematic addition. A choice of root in the schematic addition then encode the choice of P or $-P$ in A. It remains to explain how to go from the α_T model back to A: since we have the theta null point of level 4 on A, we have all the $([T_i], [T_i + T])$ where T_i is the point of four-torsion corresponding to $i \in Z(\overline{4}) \subset Z(2n)$. From $([P], [P+T])$ we can thus compute the $([P+T_i], [P+T+T_i])$, hence also the $[P-T_i]$ using a differential addition. We then recover all the coordinates of level 4 on P via:

$$U_{\chi,i}^{\mathcal{L}^2}(P)U_{\chi,i}^{\mathcal{L}^2}(T_i) = \sum_{t\in Z(\overline{2})}\chi(t)\theta_{2i+t}^{\mathcal{L}}(P+T_i)\theta_t^{\mathcal{L}}(P-T_i), \tag{2.20}$$

See [LR16, § 4.1] for more details.

Remark 2.12.5.

- Using Section 2.12.2, compressing coordinates of Example 2.8.6 is also valid if we project via the isogeny $\pi : A \to B$ from level ℓn to level $n = 2$, as long as the polarisation of level 2 on B is projectively normal. (More precisely if $n = 2$ we use $\pi : A \to B \to K_B$). See [LR16, Theorem 4.4].
- Likewise if we are only given $y \in \pi(x) \in B$ (more precisely $[y] \in K_B$), we can compute the preimages of y in A as in **??** 2.10.3.(iii). Indeed, provided we have the theta null point of A, and reusing the notations of **??** 2.10.3.(iii), we have all the $[\pi(e_i)], [\pi(e_i + e_j)]$. So once we choose among $\{[y + e_1], [-y + e_1]\}$ via a schematic addition, we get all the others $[y + e_i]$ using compatible additions, and then the exact same computation as in **??** 2.10.3.(iii) gives $\pi^-1(y)$ or $\pi^-1(-y)$. See [LR16, Remark 4.6].
- Given a symmetric theta structure of level $(2, 2, \dots, 2, 2m)$, we get that an affine lift $\tilde{x}$ of a point of A is completely determined by $(\widetilde{\pi}(\tilde{x}), \widetilde{\pi}(\widetilde{x+t}))$ where t is a point of $2m$-torsion (recall that the affine action of translation by t is determined by the theta structure on A). We may even take t be a point of m-torsion if m is odd. But this theta structure of level $(2, \dots, 2, 2m)$ induces an embedding of A. Hence we see that we may represent points of A by a pair of points on a Kummer variety, which differ by the translation of a point not of 2-torsion. This was the basic idea which led to the consideration of the model α_P in Section 2.12.1.

- By Section 2.12.2, the results of Section 2.10 also work with a projectively normal polarisation $\mathcal{L}$ of level 2. Note that, for instance in Corollary 2.10.8, the symplectic basis of $A[\ell]$ has to be computed in A rather than K_A. Typically this will be done by first computing a basis, $[e_1], \dots, [e_{2g}]$, fixing a choice of $[e_1 + e_i]$ using schematic additions, and then computing the rest of the elements of $A[\ell]$ using multiway additions.

2.13 CONCLUSION AND PERSPECTIVES

We have seen the importance of the theta group $G(\mathcal{L})$ in developing the arithmetic of $(A, \mathcal{L})$, if we are able to compute its action directly. We have also seen in Section 2.9 how an explicit description of the theorem of the square allows us to extend an explicit action of $G(\mathcal{L})$ to explicit actions of $G(\mathcal{L}^\ell)$.

All of this is crystallised when we have a symmetric theta structure on $(A, \mathcal{L})$ ($\mathcal{L}$ totally symmetric). The corresponding theta null point $(\theta_i(0))$ encodes: the explicit action of $G(\mathcal{L})$, equations of A (via Riemann relations), and an explicit theorem of the square via differential and three-way additions. This leads to completely explicit formulae for arithmetic and isogenies in the theta model, and also for pairings as we will see in Chapter 3.

Other models

However, the theta model, while extremely convenient (all this information is encoded by the coordinates of just one point!), has the major drawback that it is not rational (see Section 2.11 for precise rationality conditions). This is annoying when working with abelian varieties over finite fields, and redhibitory over number fields (the extension can be of huge degree). There is another model where we can do explicit computations in spirit similar to the theta model: namely when $A = \mathrm{Jac}(C)$ is a Jacobian; by using the tools of [CE14]. But this does not cover all cases. Possible solutions include looking at system of generators induced by the regular representation of $G(\mathcal{L})$, since this is Galois equivariant, unlike the irreducible action which requires to trivialise the action of half of $K(\mathcal{L})$. We could also look, at least over finite fields, for twists of the Heisenberg group so that we can get an isomorphism with $G(\mathcal{L})$ over the base field, and take the corresponding twisted irreducible representations. There is a lot of exciting work to do on other models or even rational (twisted) theta models. An exemple of this (for Jacobians) when $g = 2$ is given by [Fly90; CF+96]. See also [Van98] for twists of theta functions when $A = \mathrm{Jac}(C)$ is the Jacobian of an hyperelliptic curve $C : y^2 = f(x)$, which provide a model defined over the splitting field of f.

Another big drawback of the theta model of level n is that it requires a number of coordinates n^{2g} exponential in g. In practice I have done isogeny computations in dimension up to $g = 4$, and $g = 5$ would be feasible, but moderate dimensions like $g \approx 30$ would be way too costly. To handle greater dimensions, we probably need to abandon projective models, and work with affine coordinates (on affine patches) or even just with birational models. So the question becomes: can we find an efficient encoding of a birational model of an abelian variety A that still allows us (generically of course) to compute the arithmetic of A?

The case $\ell = 2$

Going back to the theta models, we have seen in Section 2.10 two different ways to compute isogenies, which we will explore in more details in Chapter 4. We have outlined how to handle the case of ℓ-isogenies with ℓ even. But works remain to do from an implementation point of view: this case has not yet been implemented in [BCR10]. Indeed, as we have seen in Remark 2.10.14 we need more information than just the kernel. So the implementation should find a way to encode this extra information in the most efficient way possible (eg should we really fix some points of $2\ell n$-torsion above the kernel, or could we choose correct signs as needed?). Also if the kernel K we want to use is not compatible with our current theta structure, we need to act by an isomorphism of the theta group (so that it becomes compatible) first. Likewise, one should consider the best way to compute such an automorphism in practice.

Even testing if K is compatible is not obvious: the explicit definition of δ_2 involves working with $\mathcal{L}^4$. This also holds when we want to go from level n to level $n\ell$ with ℓ even, and we need to assume that the symplectic decomposition of $A[n\ell]$ we take is such that the induced decomposition of $A[2n]$ does give us our symmetric theta structure of level n. My guess is that we can detect if a point P' such that $2P' = P$ is in $A_i[n]$ induces the correct symmetric lift g_P above P whenever all affine relations deduced from differential additions of the form $a\widetilde{P' + bQ}$ for $Q \in A_i[n]$ hold. Ie we need to be able to normalize $\widetilde{P'}$ such that $2\widetilde{P' + Q}$ computed via a differential addition does give $\widetilde{P + Q}$. Here $\widetilde{P' + Q}$, $\widetilde{Q}$ and $\widetilde{P + Q}$ are given by the theta structure of level n.

As a particular case, let us look at the key case of 2-isogenies between abelian varieties with a theta structure of level n even. This is exactly given by the duplication formula, and when $n = 2$ this gives generalisations of the AGM, which occurs in the case $g = 1$. Then the kernel is fixed, this is $A_2[2]$ where $A[2] = A_1[2] \oplus A_2[2]$ is the symplectic decomposition induced by the level $n = 2$ theta structure. The generalised agm involves choice of signs (the duplication formula give the squares of the theta constants, and the signs correspond to taking square roots). These choices correspond to choices of symmetric theta structures on $B = A/A_2[2]$, and when $g > 2$ not all possible choices of signs correspond to a theta structure (this is the case when $g = 2$ too but bad cases are easy to detect). These signs are fixed by some choices of points in $A[4]$ (and also some information from the 8-torsion), but a natural question would be to find compatible signs the most efficient way possible, without computing fully the 4-torsion.

The same kind of consideration holds when increasing the level from n to $2n$, using Remark 2.10.7 (ie the duplication formula again). If we assume that we are given a symplectic basis of the $2n$-torsion, what is the most efficient way to compute a theta structure of level $2n$ compatible with this basis, without computing the full $4n$-torsion?

We remark that when going from level 2 to level 4, then by Section 2.12.3 the difficulty lies only in the choice of square roots of the duplication formula for the theta constants. Once the theta constant of level 4 is chosen, changing level for $P \in A$ does not require any more choices. This is similar from level n to $2n$. Likewise, when computing a 2-isogeny $f : A \to B$ in even level via the duplication formula, once the choice of roots for $\theta_i^B(0_B)$ are done, the duplication formula directly gives the $\theta_i^B(f(P))$ in terms of the $\theta_i^A(P)$. Of course, choosing a level 4 structure on $(A, \mathcal{L}^2)$ encodes the theta null points of level 2 both of $B = A/K_1(\mathcal{L})$ and $C = A/K_2(\mathcal{L})$. Conversely, from these theta null we can reconstruct the theta null of level 4 of A. So the choice of sign in the duplication formula for a 2-isogeny or changing level from 2 to 4 are essentially equivalent.

Thomae's formula

A related consideration: in Remark 2.9.6 we explained how we could use the results of [CE14] to get a generalised Thomae algorithm on Jacobians (given a symplectic basis of the n-torsion). This approach can be seen as a more algorithmic reformulation of [Shec08], with the advantage that we get the theta constants of level n, not simply their power to the $2n^2$; and more importantly this also allows us to compute the theta coordinates of any point on the Jacobian.

Of course, the standard Thomae formula on an hyperelliptic curve gives the fourth power $\theta_i^4(0)$ of the theta of level 4 in term of the Weierstrass point of C (which encode the 2-torsion of $\mathrm{Jac}(C)$); we refer to [Mum84; Cos11] for more details, and [Cel19] for an extension to non hyperelliptic curves. A question is then how to pass from the $\theta_i^4(0)$ to the $\theta_i^2(0)$ and then $\theta_i(0)$. Via the duplication formula, the $\theta_i^2(0)$ of level 4 are essentially the same as the $\theta_i(0)$ of level 2; more precisely we have a linear change of variable between the $\theta_i^{\mathcal{L}^2}(0)$ and the $\theta_i^{\mathcal{L}}(0)\theta_j^{\mathcal{L}}(0)$. In other words, the choice of square roots correspond in terms of moduli to go from the level subgroup $\Gamma(2)$, to $\Gamma(2,4)$ to $\Gamma(4,8)$. The choice of roots from $\theta_i^2(0)$ to $\theta_i(0)$ can thus be seen as a special case of going from level 2 (ie $\Gamma(2,4)$) to level 4 (ie $\Gamma(4,8)$). Even more interesting is the first choice of roots. We could even ask for formula to go from level $\Gamma(1,2)$ to level $\Gamma(2,4)$. Here all our Thomae formulas involve projective theta constant, ie finding the values of $\theta_i(0)/\theta_0(0)$. Analytically we do have modular versions (giving the exact values of $\theta_i(0)$), and a topic of considerable interest is to have algebraic versions of the affine Thomae formula when we fix a basis of differential of A. We give such formulas when $g = 1$ in [KNR+20b]. We will go back to this topic in Section 5.6.5.

Equations for Kummer varieties

A last, but very important topic I want to mention is the following: Riemann equations from Theorem 2.7.3 gives equations for the abelian variety A and for the moduli $\mathcal{A}_{g,n}$ when $n \geq 4$ is even. However they are trivial when $n = 2$. But the case $n = 2$ is very important algorithmically to have efficient algorithms (if only because we have 2^g coordinates rather than 4^g). When $g = 2$, it is well know that Kummer surfaces are given by a quartic equation in $\mathbb{P}^3$. An immediate computation shows that in the theta model of level $n = 2$, we can recover this equation as saying that the equation coming from the differential addition $\widetilde{2x} = \texttt{diff\_add}(\tilde{x}, \tilde{x}, \tilde{0}_A, \tilde{0}_A)$ holds. So, like Riemann equations which were induced by the equation $\tilde{x} = \texttt{diff\_add}(\tilde{x}, \tilde{0}_A, \tilde{x}, \tilde{0}_A)$ we can recover the Kummer equation as some sort of compatibility conditions on differential additions. It would be interesting to extend this analysis to higher dimension g and also to get equations for the moduli of abelian varieties with a symmetric theta structure of level 2.

If $\mathcal{L}$ is symmetric and principal on A, then $\mathcal{L}^2$ descends to a line bundle $\mathcal{M}$ on K_A, where $\Gamma(A, \mathcal{M}^n) = \Gamma(A, \mathcal{L}^{2n})^+$. By surjectivity of the multiplication, $\mathcal{M}^2$ is projectively normal and by a general theorem of Kempf [Kem92], the projective embedding induced by $\mathcal{M}^2$ is described by quadric and cubic relations. So if $\mathcal{M}$ is projectively normal, it is very ample and the projective embedding is described by quartic and sextic relations. The question is then to make them explicit. It would also be nice to extend the results of Section 2.12 on the arithmetic of Kummer varieties when $\mathcal{M}$ is not projectively normal. Note that if $\mathcal{L}$ is indecomposable, $\mathcal{M}$ is very ample by Proposition 2.3.1. But if $\mathcal{L}$ decomposes as $(A, \mathcal{L}) = \prod(A_i, \mathcal{L}_i)$, it is easy to compute the arithmetic of each $(A_i, \mathcal{L}_i)$ separately, so projective normality is not a always a prerequisite. The more interesting case is when $\mathcal{L}$ is indecomposable but $\mathcal{M}$ is not projectively normal (eg $A = \mathrm{Jac}(C)$ is the Jacobian of an hyperelliptic curve of genus $g > 2$). The equations of the Kummer probably give us the extra relations we need to compute the arithmetic of K_A as in Section 2.12.

3 COMPUTING PAIRINGS IN ABELIAN VARIETIES

CONTENTS

3.1 INTRODUCTION

In this Chapter, we give algorithms to compute the Weil and Tate pairings (and related pairings) on an abelian variety. We briefly describe these pairings in Section 3.2, and refer to [Rob21, Chapter 4] for a lot more informations.

The algorithms are described in Section 3.3, where we first give a general overview when having an explicit version of the theorem of the square (as in Algorithmic Hypothesis 2.9.2), and then we specialize to the theta model. Pairings on Kummer varieties are treated in Section 3.4. I use the occasion of writing this document to give in Section 3.5 some results about pairings on elliptic curves which were only available as a Chapter of the book [Rob17], which is not publicly available. Perspectives are in Section 3.6.

3.2 PAIRINGS

3.2.1 *The Weil and Tate pairings*

If $f : A \to B$ is an isogeny, $K = \mathrm{Ker} f$, $\hat{f} : \hat{B} \to \hat{A}$ the dual isogeny, $\widehat{K} = \mathrm{Ker}\hat{f}$, then $\widehat{K}$ is canonically the Cartier dual $\mathrm{Hom}(K, \mathbb{G}_m)$ of K, and the Weil-Cartier pairing is the corresponding pairing $K \times \widehat{K} \to \mathbb{G}_m$. Applying that to the isogeny $\Phi_{\mathcal{L}^\ell}$ induced by a polarisation $\mathcal{L}$ on A, we get a pairing $e_{\mathcal{L}^\ell}$ also denoted $e_{W,\mathcal{L},\ell}$ on $K(\mathcal{L})^\ell = [\ell]^{-1}K(\mathcal{L})$. This is the standard Weil pairing on $A[\ell]$ if $\mathcal{L}$ is principal. Over a finite field there is also the Tate-Cartier pairing associated to an isogeny. We call $e_{T,\mathcal{L}^\ell}$, also denoted by $e_{T,\mathcal{L},\ell}$ the Tate-Cartier pairing associated to $\Phi_{\mathcal{L}^\ell}$. For more details on how to construct these pairings and explicit formula, we refer to [Rob21, Chapter 4].

There is a small mystery here: we will see that the Weil and Tate pairing $e_{W,\mathcal{L},\ell}$ and $e_{T,\mathcal{L},\ell}$ can be computed in time $O(\log \ell)$ in $(A, \mathcal{L})$. However I don't know of an efficient way to compute the Weil-Cartier pairing, ie without taking a preimage to reduce to a Weil pairing computation, see Section 3.6. In fact, my main motivation in writing in [Rob21, Sections 4.1.1 and 4.2] the different variants of the Weil and Weil-Cartier pairings and their relationship (along with the Tate, Tate-Cartier and Tate-Lichtenbaum pairings), was to try to find explanations as to why we had fast reformulations for the Weil (and Tate) pairing, but not yet for the Weil-Cartier pairing.

Let me detail this here. It is customary in cryptography to look at pairings on Jacobians. Then it is not hard to see, if we take for $\mathcal{L}$ the principal polarisation associated to the Theta divisor, that we can define the Weil and Tate pairing using divisors on the curve rather than divisors on the Jacobian, see [Rob21, Propositions 4.1.4 and 4.2.5]. Here Weil's reciprocity is used to replace the definition of the Weil pairing using the divisor $[\ell]^*((P) - (0))$ by the one using the divisor $\ell(P) - \ell(0)$. This allows us to use a fast double and add algorithm to compute a function associated to this divisor via Miller's algorithm described in Section 3.3.

The same method can be applied to a general abelian variety. In this case we work with cycles (and line bundles associated to these cycles) on the abelian variety, and Lang's reciprocity [Lan58] (a generalisation of Weil's reciprocity

to any variety) also allows us to work with the cycle $\ell(P) - \ell(0)$ to compute the Weil and Tate pairings, see [Rob21, Section 4.1.2]. In this case, the version of Miller's algorithm applied to abelian variety simply uses an explicit version of the theorem of the square. We explained how to compute pairings using Miller's algorithm in the theta model in [LR15a] (using of course, differential additions). This raises the question as to whether there is a similar reformulation of the Weil-Cartier pairing using reciprocity that makes it faster to compute.

When working on an abelian variety we also have the interpretation of $e_{W,\mathcal{L},\ell}$ as resulting from the commutator pairing on the theta group $G(\mathcal{L}^\ell)$. This was the original point of view of [LR10]; we reformulated this approach to show that it was equivalent to Miller's original algorithm in [LR15a] in order to apply it to the variants constructed for cryptography: ate pairings, optimal pairings. But see also the discussion after Corollary 3.3.3 as to why the point of view of the theta group naturally recovers directly the variants given by the ate and optimal ate pairings. So this is a second reformulation of the Weil pairing (the first one using reciprocity, the second one using the theta group), that allows for faster computation. Unfortunately this second reformulation does not seem to help either for the Weil-Cartier pairing.

Another motivation for writing [Rob21, Chapter 4] was also to study the restriction of the Weil and Tate pairings to subgroups. In cryptography, it is customary to restrict to the subgroups usually called $\mathbf{G}_1$ and $\mathbf{G}_2$ for elliptic curves, which represent eigenvectors for the Frobenius. It is well known that this still work for abelian varieties (using characteristic spaces), but I could not find an explicit reference, so that was the occasion to write it down, see [Rob21, Sections 4.1.3 and 4.2.3].

Anyway, in this Chapter we detail the explicit computation of the Weil, Tate (and related) pairings, in the case of a polarised abelian variety $(A, \mathcal{L})$ of level n over a finite field $\mathbb{F}_q$. Note that if $\mathcal{L} = \mathcal{L}_{A,1}^n$, $e_{W,\mathcal{L},\ell} = e_{W,\mathcal{L}_{A,1},\ell}^n$ on $A[\ell]$. We assume here that p is prime to $n\ell$. Summing up the discussion in [Rob21, Sections 4.1 and 4.2], if n is prime to ℓ and d is the embedding degree, the Weil pairing $e_{W,\mathcal{L},\ell}$ is a non degenerate pairing $A[\ell] \times A[\ell] \to \mu_\ell$, and the (reduced) Tate pairing is a pairing $e_{T,\mathcal{L},\ell} : A[\ell](\mathbb{F}_{q^d}) \times A(\mathbb{F}_{q^d})/\ell A(\mathbb{F}_{q^d}) \to \mu_\ell$.

We recall from [Rob21, Section 4.1.2] (see also Section 2.9) that if Z_P is the 0-cycle $(P) - (0_A)$ then we have an associated divisor D_P, and the corresponding line bundle associated to this cycle is $t_P^*\mathcal{L} \otimes \mathcal{L}^{-1}$. Furthermore ℓD_P is a principal divisor, and we denote $f_{\ell D_P}$ (or $f_{\ell Z_P}$) a corresponding function. More generally we may define such functions for any 0-cycle Z_P equivalent to $(P) - (0_A)$. Then by [Rob21, Corollary 4.1.3 and Proposition 4.2.5]:

Proposition 3.2.1. *Let Z_P and Z_Q be two cycles equivalent to $(P) - (0_A)$ and $(Q) - (0_A)$ respectively. Then if $P \in A[\ell](\mathbb{F}_{q^d})$ and $Q \in A(\mathbb{F}_{q^d})/\ell A(\mathbb{F}_{q^d})$, the Tate pairing is given by*

$$e_{T,\mathcal{L},\ell}(P,Q) = f_{\ell Z_P}(Z_Q)^{\frac{q^d-1}{\ell}} \tag{3.1}$$

If $P, Q \in A[\ell]$, the Weil pairing is given by

$$e_{W,\mathcal{L},\ell}(P,Q) = (-1)^\ell \frac{f_{\ell Z_P}(Z_Q)}{f_{\ell Z_Q}(Z_P)} \tag{3.2}$$

3.2.2 *Variants of the Tate pairing and twists*

A lot of variants of the Tate pairing have been introduced to speed up pairing computation for elliptic curves (and Jacobians):

- The ate pairing: $\mathbf{G}_2 \times \mathbf{G}_1 \to \mu_\ell$ given by $f_{Z_{\lambda,P}}(Z_Q)$ where $\lambda \equiv q \pmod{\ell}$. (In dimension $g > 1$ we can take $\lambda = q$ and the pairing is already reduced).
- The optimal ate pairing on $\mathbf{G}_2 \times \mathbf{G}_1$, writing a multiple of ℓ as $m\ell = \sum c_i q^i$ with small coefficients c_i and using the Frobenius to compute $f_{m\ell Z_P}(Z_Q)$ from the $f_{Z_{c_i,P}}(Z_Q)$.
- If A is ordinary and has twists of degree f, with $f \mid d$ (d the embedding degree), one can use a twist A' to represent $\mathbf{G}_2$ in $A'(\mathbb{F}_q^e)$ where $e = d/f$. Indeed twisting change the action of π by $\zeta\pi$ where ζ is a d-th root of unity. It suffices to take the twist corresponding to ζ acting as q^e modulo ℓ (since $q^{ed} = 1 \pmod{\ell}$).
- Existence of a twist also allows to define twisted versions of the ate and optimal ate pairings on $\mathbf{G}_1 \times \mathbf{G}_2$ (using the twist to reduce the length of the Miller loop rather than the size of the field representing the points). Indeed on the twist $A'/\mathbb{F}_q^e$ above, by the same reasoning as above, $\mathbf{G}_1$ is sent to $\mathbf{G}_2'$ and we have seen that $\mathbf{G}_2$ is sent to $\mathbf{G}_1'$.

- More generally, twists of degree f allows us to decompose $A(\mathbb{F}_{q^f})$. We recall that twists are in bijection with $H^1(k, \mathrm{Aut}(A))$, and if $k = \mathbb{F}_q$ and A is ordinary, then $\mathrm{Aut}_k(A) = \mathrm{Aut}_{\overline{k}}(A)$. So an element of $H^1(k, \mathrm{Aut}(A))$ is completely determined by the image of the Frobenius, which is a root of unity ζ. Then if $\mathrm{Aut}_k(A)$ contains a primitive f-root of unity ζ, we have a twist A_{ζ^i} (which becomes isomorphic to A over $\mathbb{F}_{q^f}$), and via the isomorphism above on A_{ζ^i} the Frobenius π is twisted by ζ^i. Then if the kernels $\mathrm{Ker}(\zeta^i\pi - 1)$ are disjoints, $A(\mathbb{F}_q^f) = \oplus A_{\zeta^i}(\mathbb{F}_q)$ by the same proof as [HSV06, Theorem 3].

For all this and more details, (eg on when these pairings are non degenerate) we refer to [Rob17, § 3.2.4 and § 3.2.5] and the references therein for elliptic curves, and [LR15a, § 6 and § 7] for abelian varieties.

3.3 MILLER'S ALGORITHM

3.3.1 *Overview of Miller's algorithm in abelian varieties*

Computing pairings thus boils down to computing the functions $f_{\ell Z_P}$ for $P \in A[\ell]$ (or more generally functions $f_{Z_{\lambda,P}}$). There is a standard algorithm, due to Miller for elliptic curves and Jacobians [Mil86; Mil04] but which extends readily to abelian varieties.

The cycle $(P+Q) + (0_A) - (P) - (Q)$ corresponds to a divisor linearly equivalent to 0, and we let $\mu_{P,Q}$ be a function representing this divisor. This function $\mu_{P,Q}$ can be seen as making the theorem of the square $t_{P+Q}^*\mathcal{L} \otimes \mathcal{L} \simeq t_P^*\mathcal{L} \otimes t_Q^*\mathcal{L}$ explicit. Its construction depends on the model of the abelian variety. It is clear that we can compute the function associated to a principal cycle Z (we recall that we define a principal cycle to be a cycle of degree zero such that its realisation $S(Z) = 0_A$) by combinations of the $\mu_{P,Q}$ functions (using the special case $Q = -P$ to replace $(0) - (P)$ by $(-P) - (0)$). But if $Z = \sum n_i(P_i)$, this costs $O(\sum|n_i|)$ operations.

Like for the addition law, we look for a double and add method to compute the function associated to cycles of the form $n(P) + \cdots$. For an arbitrary point $P \in A$, we let $Z_{m,P}$ be the cycle $m(P) - (mP) - (m-1)(0_A)$, and $f_{Z_{m,P}}$ or simply $f_{m,P}$ be the corresponding function (this depends on the choice of a divisor Θ representing the polarisation $\mathcal{L}$, and we denote by $f_{\Theta,m,P}$ this function when we want to make this choice explicit). Its divisor is linearly equivalent to 0, and if $P \in A[m]$, we have that $Z_{m,P} = m[(P) - (0_A)] = mZ_P$. The key insight of Miller's algorithm is the following relation (up to changing one of the representative function by a constant factor):

$$f_{Z_{\ell_1+\ell_2,P}} = \mu_{\ell_1 P,\ell_2 P} f_{Z_{\ell_1,P}} f_{Z_{\ell_2,P}} \tag{3.3}$$

This immediately yields a double and add algorithm to evaluate $f_{Z_{m,P}}(Q)$, using that $f_{Z_{1,P}} = f_{Z_{0,P}} = 1$. As a corollary, provided we have an explicit version of the theorem of the square, we can compute a function associated to a cycle $Z = \sum n_i(P_i)$ in time $O(\sum \log|n_i|)$. In particular, we can compute the Tate $e_{T,\mathcal{L},\ell}$ and Weil $e_{W,\mathcal{L},\ell}$ pairing in time $O(\log \ell)$ (with the caveat that the final exponentiation in the Tate pairing becomes asymptotically dominant).

3.3.2 *Miller's algorithm in the theta model*

In the following, $(A, \mathcal{L})$ is a polarised abelian variety, with a symmetric theta structure of even level n.

If $A = \mathbb{C}^g/\Lambda$ is a complex abelian variety described by the analytic theta functions θ_i of level n (we recall that for $i \in Z(\overline{n})$, $\theta_i(z) = \theta\left[\begin{smallmatrix} 0 \\ i/n \end{smallmatrix}\right](z, \Omega/n)$), we can describe these functions explicitly, see [LR15a]:

Lemma 3.3.1. *Let $z_P \in \mathbb{C}^g$ representing $P \in A$, and $i \in Z(\overline{n})$. For all λ, μ positive integers, we have up to a constant factors (for all z where this is defined):*

$$\mu_{\lambda P,\mu P}(z) = \frac{\theta_i(z+\lambda z_P)\theta_i(z+\mu z_P)}{\theta_i(z+(\lambda+\mu)z_P)\theta_i(z)}. \tag{3.4}$$

More generally: $\mu_{P,Q}(z) = \frac{\theta_i(z+z_P)\theta_i(z+z_Q)}{\theta_i(z+z_P+z_Q)\theta_i(z)}$.

$$f_{\lambda,P}(z) = \frac{\theta_i(z)}{\theta_i(z+\lambda z_P)}\left(\frac{\theta_i(z+z_P)}{\theta_i(z)}\right)^{\lambda}, \tag{3.5}$$

It is easy to give an algebraic interpretation of Lemma 3.3.1

Proposition 3.3.2. *Let $(A, \mathcal{L})$ be a polarised abelian variety with a symmetric theta structure of level n even. Fix once and for all an affine lift $\tilde{0}_A$ of the theta null point.*

- *Let $P, Q, R, S \in A$. Fix arbitrary lifts $\widetilde{P}, \widetilde{Q}$ and $\widetilde{P+Q}$, $\widetilde{R}$, $\widetilde{P+R}$, $\widetilde{Q+R}$, $\widetilde{R+S}$, $\widetilde{P+R}+S$, $\widetilde{Q+R}+S$. Compute $\widetilde{P+Q}+R, P+\widetilde{Q+R}+S$ using* `threeway_add`. *Let $i \in Z(\overline{n})$. Then*

$$\mu_{\Theta_i,P,Q}((S+R)-(R)) = \frac{(\widetilde{R+S}+P)_i(\widetilde{R+S}+Q)_i(\widetilde{R+P}+Q)_i\widetilde{R}_i}{(\widetilde{R+S+P}+Q)_i(\widetilde{R+S})_i(\widetilde{R+P})_i(\widetilde{R+Q})_i}$$

  where Θ_i is the divisor representing $\mathcal{L}$ associated to θ_i.
  In particular, fixing arbitrary lifts $\widetilde{P}, \widetilde{Q}$ and $\widetilde{P+Q}, \widetilde{T}, \widetilde{P+T}, \widetilde{Q+T}$, and computing $\widetilde{P+\widetilde{Q}+T}$ using `threeway_add`,

$$\mu_{\Theta_i,P,Q}((T)-(0)) = \frac{(\widetilde{P+Q})_i(\widetilde{T+P})_i(\widetilde{T+Q})_i\tilde{0}_i}{(\widetilde{P+\widetilde{Q}+T})_i(\widetilde{T})_i(\widetilde{P})_i(\widetilde{Q})_i}.$$

- *Let $P, R, S \in A$. Fix arbitrary lifts $\widetilde{P}, \widetilde{R}, \widetilde{P+R}, \widetilde{R+S}, \widetilde{P+R}+S$, and compute $\widetilde{\lambda P+R}, \widetilde{\lambda P+R}+S$ using* `diff_multadd`. *Then*

$$f_{\Theta_i,\lambda,P}((S+R)-(R)) = \frac{(\widetilde{R+S}+P)_i^{\lambda}(\widetilde{R+\lambda P})_i\widetilde{R}_i^{\lambda-1}}{(\widetilde{R+S+\lambda P})_i(\widetilde{R+S})_i^{\lambda-1}(\widetilde{R+P})^{\lambda}}.$$

  In particular, fixing arbitrary lifts $\widetilde{P}, \widetilde{Q}, \widetilde{P+Q}$ and computing $\widetilde{\lambda P}, \widetilde{\lambda P+Q}$ using `diff_multadd`, *then*

$$f_{\Theta_i,\lambda,P}((Q)-(0)) = \frac{\widetilde{Q}_i.\widetilde{\lambda P}_i}{(\widetilde{Q+\lambda P})_i.\tilde{0}_i}\left(\frac{(\widetilde{P+Q})_i.\tilde{0}_i}{\widetilde{Q}_i.\widetilde{P}_i}\right)^{\lambda}$$

Proof. This is [LR15a, Corollary 1 and Proposition 3]. We check that, due to our normalisations, these values does not depend on the choice of lifts. Over $\mathbb{C}$ we may take lifts given by the $\theta_i(z+z_P)$ and so on for z representing $R+S$ and S, to see that this does define $\mu_{\Theta_i,P,Q}((R+S)-(S))$. Since we evaluate on a cycle of degree zero, this does not depends on the choice of $\mu_{\Theta_i,P,Q}$. The algebraic case follows by a lifting argument, but it can also be proved directly using Section 2.9: differential additions give an explicit version of the theorem of the square. This gives a formula for $\mu_{P,Q}$, and the formula for $f_{\lambda,P}$ follows immediately. □

Corollary 3.3.3. *If $P \in A[\ell], Q \in A$, take arbitrary lifts $\widetilde{P}, \widetilde{Q}, \widetilde{P+Q}$. Compute $\widetilde{\ell P+Q}$ and $\widetilde{\ell P}$ using* `diff_multadd`. *Write $\widetilde{\ell P+Q} = \lambda_1^P\widetilde{Q}$ and $\widetilde{\ell P} = \lambda_0^P\tilde{0}$.*
Let d be the embedding degree, and assume that $P \in A[\ell](\mathbb{F}_{q^d})$, $Q \in A(\mathbb{F}_{q^d})$, and the chosen lifts are in $\mathbb{F}_{q^d}$. Then the (non) reduced Tate pairing is given by

$$e_{T,\mathcal{L},\ell}(P,Q) = \lambda_1^P/\lambda_0^P.$$

If $P, Q \in A[\ell]$, we can also compute $\widetilde{P+\ell Q}$ and $\widetilde{\ell Q}$ using `diff_multadd`, *and recover projective factors $\widetilde{P+\ell Q} = \lambda_1^Q\widetilde{P}$ and $\widetilde{\ell Q} = \lambda_0^Q\tilde{0}$. Then the Weil pairing is given by*

$$e_{W,\mathcal{L},\ell}(P,Q) = \frac{\lambda_1^P\lambda_0^Q}{\lambda_1^Q\lambda_0^P}.$$

Proof. This is an immediate application of plugging the formulae of Proposition 3.3.2 into Proposition 3.2.1, see also [LR15a]. We note that Proposition 3.3.2 shows how the result of the non reduced Tate pairing is defined up to an ℓ-power of an element of $\mathbb{F}_{q^d}^*$. □

I really like this formula, because it really fits into the theme "you could have invented the Weil and Tate pairing" (see [Choo6] for the reference). We take a point of ℓ-torsion P, use differential additions to compute $\widetilde{\ell P+Q}$, we know that we get back $\widetilde{Q}$ up to some factor λ_1^P. Looking at what happen when we change our lifts, we get the following diagram:

$$\begin{array}{cccc}
\tilde{0} & \alpha\widetilde{P} & \alpha^4(2\widetilde{P}) & \dots & \alpha^{\ell^2}(\ell\widetilde{P}) = \lambda'^P_0\tilde{0} \\
\beta\widetilde{Q} & \gamma(\widetilde{P+Q}) & \frac{\gamma^2\alpha^2}{\beta}(2\widetilde{P+Q}) & \dots & \frac{\gamma^\ell\alpha^{\ell(\ell-1)}}{\beta^{\ell-1}}(\ell\widetilde{P+Q}) = \lambda'^P_1\beta\widetilde{Q} \\
\beta^4(2\widetilde{Q}) & \frac{\gamma^2\beta^2}{\alpha}(\widetilde{P+2Q}) & & & \\
\dots & \dots & & & \\
\beta^{\ell^2}(\ell\widetilde{Q}) = \lambda'^Q_0\tilde{0} & \frac{\gamma^\ell\beta^{\ell(\ell-1)}}{\alpha^{\ell-1}}(\widetilde{P+\ell Q}) = \lambda'^Q_1\alpha\widetilde{P} & & &
\end{array}$$

So we get that $\lambda'^P_1/\lambda'^P_0 = \lambda^P_1/\lambda^P_0\left(\frac{\gamma}{\alpha\beta}\right)^\ell$. (The above diagram shows that we don't really need to normalise with λ^P_0 for the Tate pairing, but we get a less symmetric formula.) So we could define $e_T(P,Q)$ directly by the formula $e_T(P,Q) = \lambda^P_1/\lambda^P_0$. It is pretty easy to check directly bilinearity (using compatibility of `diff_add` with `threeway_add`), and that if $Q = \ell Q_0$ with $Q_0 \in A(\mathbb{F}_{q^d})$, $e(P,Q)$ is an ℓ-th power. However, it does not seem easy to prove non degeneracy directly with this formula, without linking it with the Tate pairing.

Likewise, defining directly $e_W(P,Q)$ via the formula $e_W(P,Q) = \frac{\lambda^P_1\lambda^Q_0}{\lambda^Q_1\lambda^P_0}$, the above diagram shows that it depends not on our choice of lifts (here we really need the normalisation factors λ^P_0, λ^Q_0). But it should be clear from the way our computations are done, and by the compatibility of the action of the theta group and differential additions, that we are recovering the commutator pairing on $G(\mathcal{L}^\ell)$. In fact, this is how we first proved this formula in [LR12], before making the link with the standard definition with [LR10; LR15a].

Indeed, we can directly compute the commutator pairing in $(A,\mathcal{L})$ as follow: if $x,y \in K(\mathcal{L})$ and we fix arbitrary lifts $\tilde{x},\tilde{y},\widetilde{x+y}$, let us write $\tilde{x} = (\alpha,i_1,j_1).\tilde{0}_A$ where $(\alpha,i_1,j_1) \in \mathcal{H}(\overline{n})$, and $\tilde{y} = (\beta,i_2,j_2).\tilde{0}_A$, we can also write $\widetilde{x+y}$ in two ways: $\widetilde{x+y} = (\gamma_1,i_1,j_1).\tilde{y}$ and $\widetilde{x+y} = (\gamma_2,i_2,j_2).\tilde{x}$. Then $e_{\mathcal{L}}(x,y) = \frac{\langle i_1,j_2\rangle}{\langle i_2,j_1\rangle}\frac{\gamma_2\alpha}{\gamma_1\beta}$. Then using Key Idea 3, we recover Corollary 3.3.3 for $e_{\mathcal{L}^\ell}$ by descending this formula for $e_{\mathcal{L}^{\ell^2}}$ by the isogeny $[\ell]$. See [Rob10, Théorème 5.4.1]. In fact, Proposition 2.10.5 gives another proof of this, as remarked in Remark 2.10.6.

Remark 3.3.4. So rather than proving Corollary 3.3.3 from the formula of Proposition 3.2.1 and Miller's algorithm, we can as in [LR10] prove the formula directly by considering the pairings as commutator pairings on the theta group. It may seem strange that we can recover the formula from [Rob21, Corollary 4.1.3] without invoking Lang's reciprocity. But Lang's theorem is based upon the theorem of the square [Lan58, § 2], and the theorem of the square is also exactly the theorem that shows that $\Phi_{\mathcal{L}}$ is a polarisation and gives rise to the theta group. Hence it is not surprising that we can recover the same formula by working directly on the theta group.

The same philosophy gives the ate pairing and the optimal ate pairing. We briefly detail this: if $P \in \mathbf{G}_2$ and $Q \in A(\mathbb{F}_q)$, letting $\lambda \equiv q \mod \ell$, we may compare $\lambda\widetilde{P+Q}$ computed using `diff_add` to $\pi_q(\widetilde{P+Q})$, which are both affine lifts of the point $\pi_q(P+Q) = qP+Q$. This gives a projective factor λ^P_1, which we normalise with respect to the projective factor λ^P_0 comparing $\widetilde{\lambda P}$ and $\pi_q(\widetilde{P})$. Then $a_T(P,Q) = \lambda^P_1/\lambda^P_0$ is exactly the ate pairing (except that as usual, since we work with a polarisation of level n we compute $a_{T,\mathcal{L}}(P,Q) = a_{T,\mathcal{L}_0}(P,Q)^n$). Furthermore writing $q^d - 1 = m\lambda$, we can relate the ate pairing with the Tate pairing directly by tracking the projective factors using the differential addition rather than working with divisors as in the standard proofs, see [LR15a, Remark 5].

A similar philosophy apply for the optimal ate pairing: write $m\ell = \sum c_i q^i$, let $P \in \mathbf{G}_2$ and $Q \in A(\mathbb{F}_q)$. Compute the $\widetilde{c_iP}, \widetilde{c_iP+Q}$ using `diff_multadd`, and then apply (powers of) the Frobenius π_q to get $\widetilde{c_iq^iP}$ and $\widetilde{c_iq^iP+Q}$. We then compute an arbitrary lift $c_iq^iP + \widetilde{\sum_{j>i} c_jq^jP}$, and then use `threeway_add` to compute $c_iq^iP + \widetilde{\sum_{j>i} c_jq^jP} + Q$ (changing the first lift by λ change the three way add by λ too, so in the end everything is correctly normalised). We thus get $\sum\widetilde{c_iq^iP}$ and $\sum\widetilde{c_iq^iP} + Q$, and comparing with $\tilde{0}$ and $\widetilde{Q}$ as usual gives exactly the optimal ate pairing (up to the power n). Once again we could relate the optimal ate pairing with the Tate pairing directly by tracking only the factors in the differential additions. We refer to [LR15a, § 6 and § 7] for more details, including a look at the twisted versions.

In summary:

Key Idea 5. *Relating affine lifts computed in different ways using the differential addition naturally recover pairings (in particular the commutator pairing).*

Remark 3.3.5.

- During the execution of Miller's algorithm to compute the Tate and Weil pairing, the intermediate steps introduce extra zeroes and poles, and so the intermediate evaluations of $f_{Z_{\lambda,P}}(Z_Q)$ may not be well defined even if $f_{\ell Z_P}(Z_Q)$ is. That is why we stated Proposition 3.2.1 allowing cycles linearly equivalent

to $(P)-(0)$ and $(Q)-(0)$ rather than working directly with them. This has the inconvenient of giving a non deterministic algorithm where we may need to restart the computation with a different equivalent cycle if we encounter such a situation.

- An alternative to replacing $(Q)-(0)$ by a linearly equivalent cycle would be to define an extended value of a function f at Q which has a pole or zero by fixing uniformisers and looking at the coefficients of the Laurent series. One then need to check that this does not depend on the choice of uniformisers and that this extended evaluation still gives the correct result.

  This is classical for elliptic curves: $f_{\ell((P)-(0))}$ has of course a pole of order ℓ at 0 so is not well defined on the cycle $(Q)-(0)$. But fixing the uniformiser $z=-x/y$ at 0_E, we may define the value of a function f at 0_E as $\left(\frac{f}{z^{v_{0_E}(f)}}\right)(0_E)$, which is well defined. We say that f is normalised at 0_E if the extended value $f(0_E)=1$. The standard definition of the functions $\mu_{P,Q}$ for elliptic curves are normalised at 0_E, so the functions $f_{\ell,P}$ computed via Miller's algorithm are also normalised. Thus Proposition 3.2.1 becomes $e_{T,\ell}(P,Q)=f_{\ell,P}(Q)$ (non reduced) and $e_{W,\ell}(P,Q)=(-1)^{\ell}f_{\ell,P}(Q)/f_{\ell,Q}(P)$.

  For the Weil pairing on elliptic curves, for the computation of the function $f_{\ell,P}$, the intermediate poles and zeroes introduced by Miller's algorithm are all multiple of P. So if the computation fails, we know that Q is a multiple of P, so its Weil pairing with P is 1 anyway.

  But for the Tate pairing, if we get an intermediate zero and pole T, we can compute an extended value at T (generalizing the definition above for $T=0_E$), and at the end we recover the Tate pairing. This may not be too useful in the cryptographic setting where we have lots of rational points, but it is useful to compute Tate pairings on elliptic curves without many rational points. See Section 3.5 for formulae for elliptic curves, which were implemented in Pari/GP by Bill Allombert.

- By contrast there is no problem of this type using Corollary 3.3.3. Indeed, since Q has always a non zero coordinate i, we may always compute the projective factor relating $\widetilde{\ell P+Q}$ and $\widetilde{Q}$ using this coordinate i. This is related to Remark 2.9.4 that our polarisation $\mathcal{L}$ is base point free.

- The pairing $e_{\mathcal{L}^\ell}$ is actually defined on $K(\mathcal{L}^\ell)$, so on $A[\ell n]$ if $\mathcal{L}$ is of level n. It is easy to extend Corollary 3.3.3 as follow: if $P\in A[\ell n]$, we can compute $\widetilde{\ell P+Q}$ and look for the action of the Heisenberg group $g=(\lambda_1^P,i_1,j_1)$ such that $\widetilde{\ell P+Q}=g\widetilde{Q}$, and so on. Then the same formula as in Corollary 3.3.3 gives the commutator pairing on $e_{\mathcal{L}^\ell}$ ([Rob10, Théorème 5.4.1]). Likewise for the Tate pairing $e_{T,\mathcal{L}^\ell}$.

3.4 PAIRINGS ON THE KUMMER VARIETY

The beauty of the approach of Miller's algorithm via differential additions in Section 3.3.2 is that it works equally well for Kummer varieties. In the following we assume that we are away from characteristic two.

Of course the Weil and Tate pairings are not well defined on the Kummer variety K_A, since (using the notations of Section 2.12), we have $e([P],[Q])=e(P,Q)^{\pm 1}$. So to get a well defined pairing, we have to work over $\mathbb{G}_m/\pm 1$ (where -1 acts by $x\mapsto x^{-1}$). This is a good occasion to illustrate the techniques of Section 2.12.1. If $x\in\mathbb{G}_m(\bar{k})$, we represent by $[x]$ its value in $\mathbb{G}_m/\pm 1$. We have a model of $\mathbb{G}_m/\pm 1$ given by $t:[x]\mapsto x+1/x$. The possible values $x^{\pm 1}$ can easily be recovered from $t([x])$ via the equation $X^2-t([x])+1=0$.

Given $[x],[y]\in\mathbb{G}_m/\pm 1$, represented by t_x,t_y, then the schematic addition $\{[xy],[x/y]\}$ is represented by the polynomial $X^2-t_xt_yX+t_x^2+t_y^2-4$, whose roots are t_{xy} and $t_{x/y}$.

Plugging the constructions of Section 2.12.1, we get

- doubling: $t([x^2])=x^2+1/x^2=(x+1/x)^2-2=t_x^2-2$;
- differential addition $t([xy])=xy+1/xy=(x+1/x)(y+1/y)-(x/y+y/x)=t_xt_y-t([x/y])$;
- compatible additions …

In particular, the final exponentiation in Tate's pairing can be done over $\mathbb{G}_m/\pm 1$ too.

Now the Weil, Tate, ate and optimal ate pairings all require, from P,Q to compute an affine point of the form $P\widetilde{+\lambda}Q$. On the Kummer, from $[P]$, $[Q]$, we cannot compute $[P+Q]$, but we can use the schematic addition to represent the pair $\{[P+Q],[P-Q]\}$. This is an affine scheme Spec R (isomorphic to Spec $k\times$ Spec k if P,Q comes

from rational points in A), so we may see this pair as a point of $K_A(R)$. Then proceeding as in Section 3.3.2 to compute the pairings, we get an element of $\mathbb{G}_m(R)$ on which we can use t to get the value of $e(P,Q)+e(-P,Q)$.

For the optimal ate version of the pairing, we also need to compute $c_iq^iQ+\sum_{j>i}c_jq^jQ$ in order to use a three way addition to get $P+c_iq^i\widetilde{Q}+\sum_{j>i}c_jq^j\widetilde{Q}$. But given our $P+Q$ (or rather our $\{[P+Q],[P-Q]\}$), we can compute $c_iq^i\widetilde{Q}$, $\sum_{j>i}c_jq^j\widetilde{Q}$, $P+c_iq^i\widetilde{Q}$, $P+\sum_{j>i}c_jq^j\widetilde{Q}$ and then use a compatible addition to recover $c_iq^iQ+\sum_{j>i}c_jq^jQ$.

3.5 THE WEIL AND TATE PAIRINGS FOR ELLIPTIC CURVES

In writing the Chapter 3 of the book [Rob17], I have thought about proving the non degeneracy of the Weil and Tate pairing on an elliptic curve $E/\mathbb{F}_q$ in the most elementary way, without involving cohomology. This was also a good occasion to give formulae for the extended value of a function $f\in k(E)$ at a point P which may be a pole or zero (see Remark 3.3.5). The idea to use extended values dates back to Miller's original article [Mil86], but to my knowledge no explicit formula was given for elliptic curves. Since the book is unfortunately not publicly available, this document is a good occasion to give the formulae.

First we fix uniformisers π_P at each point P of E. It is customary to take $\pi_P=x-x_P$ if P is not a Weierstrass point, $\pi_P=y$ if P is a Weierstrass point (away from infinity), and $\pi_{0_E}=-x/y$. If $\mathrm{ord}_P:=v_P$ is the valuation on $k(E)$ induced by the rational point P, a uniformiser is a function π_P such that $\mathrm{ord}_P\,\pi_P=1$. Then if $f\in k(E)$ have valuation $\mathrm{ord}_P(f)=m$ at P, the extended value of f at P is defined to be $\left((f/\pi_P^m)(P),m\right)$. There is an obvious group law on this pair, so we can extend the definition to $f(D)$ where D is any divisor. If D is of degree 0, this does not change when multiplying by a constant, so the value $D'(D)$ (denoted by $f_{D'}(D)$) makes sense for a principal divisor D'. We then have the following generalisation of Weil's reciprocity theorem:

Theorem 3.5.1 (Weil's reciprocity theorem)**.** *Let $f,g\in k(E)$. Then*

$$f(\mathrm{Div}(g))=(-1)^{\sum_P \mathrm{ord}_P(f)\,\mathrm{ord}_P(g)}g(\mathrm{Div}(f)).$$

Expressing the above equation in terms of divisors, we get the following reformulation: Let D_1 and D_2 be two degree 0 divisors and define $\epsilon(D_1,D_2)=(-1)^{\sum_P \mathrm{ord}_P(D_1)\,\mathrm{ord}_P(D_2)}$. If D_1 and D_2 are principal, then

$$f_{D_1}(D_2)=\epsilon(D_1,D_2)f_{D_2}(D_1).$$

Proof. See [Ser75, p. 44–46]. □

We then deduce the following versions of the Weil and Tate pairings:

Theorem 3.5.2. Weil*: Let $E/\mathbb{F}_q$ be an elliptic curve, ℓ a prime different from p and P and Q two points of ℓ-torsion on E. Let D_P be a divisor linearly equivalent to $[P]-[0_E]$ and D_Q be a divisor linearly equivalent to $[Q]-[0_E]$. Then*

$$e_{W,r}(P,Q)=\epsilon(D_P,D_Q)^{\ell}\frac{f_{\ell D_P}(D_Q)}{f_{\ell D_Q}(D_P)} \tag{3.6}$$

is well defined, does not depend on the choice of uniformisers nor on the choice of D_P and D_Q. In particular, $e_{W,r}=(-1)^{\ell}\frac{f_{((P)-(0))}((Q)-(0))}{f_{((Q)-(0))}((P)-(0))}$, and if the functions are normalised at 0_E, $e_{W,r}=(-1)^{\ell}\frac{f_{((P)-(0))}(Q)}{f_{((Q)-(0))}(P)}$.

Furthermore the application $E[\ell]\times E[\ell]\to\mu_\ell:(P,Q)\mapsto e_{W,\ell}(P,Q)$ is the Weil pairing. The pairing $e_{W,\ell}$ is an alternate pairing, which means that $e_{W,\ell}(P,Q)=e_{W,r}(Q,P)^{-1}$.

Tate*: Let ℓ be an integer dividing $\#E(\mathbb{F}_q)$, d be the embedding degree, $P\in E[\ell](\mathbb{F}_{q^d})$ a point of ℓ-torsion defined over $\mathbb{F}_{q^d}$ and $Q\in E(\mathbb{F}_{q^d})$ a point of the elliptic curve defined over $\mathbb{F}_{q^d}$. Let D_P be a divisor linearly equivalent to $[P]-[0_E]$ and D_Q be a divisor linearly equivalent to $[Q]-[0_E]$. Then*

$$e_{T,\ell}(P,Q)=\left(f_{\ell D_P}(D_Q)\right)^{\frac{q^\ell-1}{\ell}} \tag{3.7}$$

is well defined, does not depend on the choice of uniformisers nor on the choice of D_P and D_Q. Furthermore the application $E[r](\mathbb{F}_{q^d})\times E(\mathbb{F}_{q^d})/rE(\mathbb{F}_{q^d})\to\mu_r:(P,Q)\mapsto e_{T,\ell}(P,Q)$ is the Tate pairing.

Proof. This is proven in [Rob17, Theorem 3.10 and Theorem 3.11], except for the case $d = 1$ in the Tate pairing, so let us give more details.

We use Weil's reciprocity from Theorem 3.5.1 to show that the definition of $e_{W,\ell}$ does not depend on the equivalence class of D_P. Since it is obviously alternate, this holds for D_Q too. Using $D_P = (P) - (0_E)$ it is easy to see that the result does not depend on the choice of uniformisers. Using Weil's reciprocity again, we show that the Weil pairing is equivalent to the other definition using the function $g_{\ell,P}$: $e_{W,\ell}(P,Q) = g_{\ell,P}(x+Q)/g_{\ell,P}(x)$ where $g_{\ell,P}$ has for divisor $[\ell]^*((P)-(0))$. Bilinearity on the right is then immediate. Likewise, if it was degenerate on the right, the function $g_{\ell,P}$ would be invariant by translation by a point $Q \in E[\ell]$, so would be of the form $h \circ [\ell]$. But h would have for divisor $(P) - (0_E)$ hence h would give an isomorphism $E \simeq \mathbb{P}^1$, which is absurd since these curves don't have the same genus. Using that $e_{W,\ell}$ is alternate, we get bilinearity and non-degeneracy on the left too.

If $d > 1$, we may also define $\mathbf{G}_1$ and $\mathbf{G}_2$ as in [Rob21, Section 4.1.3]. We have $E[\ell] = \mathbf{G}_1 \oplus \mathbf{G}_2$ (since we have two distinct eigenvalues 1 and q) and these are 1-dimensional, and the Weil pairing is trivial on $\mathbf{G}_1 \times \mathbf{G}_1$ and $\mathbf{G}_2 \times \mathbf{G}_2$, so it is non trivial on $\mathbf{G}_1 \times \mathbf{G}_2$ and $\mathbf{G}_2 \times \mathbf{G}_1$, or more generally on $\mathbf{G}_1 \times \mathbf{G}_3$ for any supplement of $\mathbf{G}_1$ and so on. (The whole situation is a lot simpler than in [Rob21, Section 4.1.3].)

All this is classical. The case of the Tate pairing is more fun. It is immediate that it does not depend on the class of D_P, and we use Weil's reciprocity again to show that it does not depend on the class of D_Q. It is also straightforward to check that it does not depend on the choice of uniformisers. We show the alternative definition $e_{T,\ell}(P,Q) = e_{W,\ell}(P,(\pi^d-1)Q_0)$ where $\ell Q_0 = Q$ (since ℓQ_0 is rational, $(\pi^d-1)Q_0$ is a point of ℓ-torsion): $e_{W,\ell}(P,(\pi^d-1)Q_0) = \frac{g_{\ell,P}(\pi^d Q_0)}{g_{\ell,P}(Q_0)} = g_{\ell,P}(Q_0)^{q^k-1} = (g_{\ell,P}^{\ell})^{(q^d-1)/\ell} = f_{\ell,P}(Q)^{(q^d-1)/\ell}$, using that π^d commutes with $g_{\ell,P}$ if $P \in E[\ell](\mathbb{F}_{q^d})$, and that $g_{\ell,P}^{\ell} = f_{\ell,P} \circ [\ell]$.

This shows that $e_{W,\ell}(P,(\pi^d-1)Q_0)$ does not depends on the choice of Q_0. We can prove it directly as follow: this is obvious if $E[\ell]$ is rational over $\mathbb{F}_{q^d}$ (for instance if $d > 1$), since another choice Q_0' satisfy $Q_0' = Q_0 + T$ with $T \in E[\ell](\mathbb{F}_{q^d})$, so $(\pi^d-1)(Q_0) = (\pi^d-1)(Q_0')$. The remaining case is $d = 1$ and $\mathbf{G}_1 = E[\ell](\mathbb{F}_q)$ of rank 1. Then the matrix of the Frobenius π acting on $E[\ell]$ is given by $\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$. But then $(\pi-1)E[\ell] = \mathbf{G}_1$, so since the Weil pairing is trivial on $\mathbf{G}_1 \times \mathbf{G}_1$ we get an alternative proof that the result does not depend on Q_0. Bilinearity is then obvious from the bilinearity of the Weil pairing.

It remains to show non degeneracy. We first treat the case where all the ℓ-torsion is defined over $\mathbb{F}_{q^d}$. Then we have seen that $\alpha = \frac{\pi^d-1}{\ell} : E(\mathbb{F}_{q^d}) \to E[\ell]$ is well defined and is an endomorphism. Its kernel is $\ell E(\mathbb{F}_{q^d})$. We have $E(\mathbb{F}_{q^d}) = \mathbb{Z}/a\mathbb{Z} \oplus \mathbb{Z}/b\mathbb{Z}$ with $a \mid b$, and since $E[\ell] \subset E(\mathbb{F}_{q^d})$, $\ell \mid a$, so $E(\mathbb{F}_{q^d})/\ell E(\mathbb{F}_{q^d}) \simeq (\mathbb{Z}/\ell\mathbb{Z})^2$. This implies that $\alpha(E(\mathbb{F}_{q^d})) = E[\ell](\mathbb{F}_{q^d})$, which proves non degeneracy since the Weil pairing is non degenerate.

Also, if $d > 1$, and assuming that $E(\mathbb{F}_{q^d})$ does not contain a point of ℓ^2-torsion so we may identify $E[\ell] = \mathbf{G}_1 \oplus \mathbf{G}_2$ with $E(\mathbb{F}_{q^d})/\ell E(\mathbb{F}_{q^d})$, then since α commutes with π, it stabilizes $\mathbf{G}_1$ and $\mathbf{G}_2$, so we get from the above case of the Weil pairing that $e_{T,\ell}$ is trivial on $\mathbf{G}_1 \times \mathbf{G}_1$ and $\mathbf{G}_2 \times \mathbf{G}_2$ and non degenerate on $\mathbf{G}_1 \times \mathbf{G}_2$ or $\mathbf{G}_2 \times \mathbf{G}_1$, or $\mathbf{G}_1 \times \mathbf{G}_3$…

It remains to check non degeneracy for $d = 1$. We have $E[\ell](\mathbb{F}_q) = \mathbf{G}_1$ so we need to check that there is a $\pi Q_0 - Q_0$ not in $\mathbf{G}_1$ for a $Q \in E(\mathbb{F}_q)$. Taking $Q = P'$ where $P' \in E(\mathbb{F}_q)$, $\ell^m P' \in \mathbf{G}_1$ for some $m \geq 0$ but P' does not have a rational preimage by $[\ell]$ gives the sought after point. Indeed let $P = \ell^m P' \in \mathbf{G}_1$ and assume that $\pi Q_0 - Q_0 = \lambda P \in \mathbf{G}_1$. Then taking $Q_1 = Q_0 - \lambda P$, we have $\ell Q_1 = \ell Q_0 = P'$ and $\pi Q_1 - Q_1 = \lambda P - \lambda P = 0$, so Q_1 is a rational preimage of P' by $[\ell]$.

Working a bit more we could extend this approach to a non prime ℓ, but at this point, comparing this "by hand" proof with the beautiful cohomological proof of [Rob21, Proposition 4.2.1], one cannot but admire the beauty of conceptual proofs. □

It remains to give the explicit formula for the extended value of $\mu_{P,Q}$ on a point [Rob17].

Lemma 3.5.3 (Evaluating $\mu_{P,Q}$). *Let $E : y^2 = f(x) = x^3 + ax + b$ be an elliptic curve. Let $P = (x_P, y_P)$, $Q = (x_Q, y_Q)$, and $R = (x_R, y_R)$ be points on E, with P, Q and $P+Q$ all different from 0_E. Then $\mu_{P,Q} = \frac{l_{P,Q}}{v_{P,Q}}$ where $l_{P,Q} = y - \alpha x - \beta$ with $\alpha = \frac{y_P - y_Q}{x_P - x_Q}$ when $P \neq Q$ and $\alpha = f'(x_P)$ when $P = Q$, $\beta = y_P - \alpha x_P = y_Q - \alpha x_Q$ and $v_{P,Q} = x - x_{P+Q}$ with $x_{P+Q} = \alpha^2 - x_P - x_Q$.*

The extended value of $v_{P,Q}(R)$ is given by the following cases (taking into account that $\mathrm{Div}(v_{P,Q}) = [P+Q] + [-P-Q] - 2[0_E]$):

- *If R is different from $P+Q$, $-P-Q$ or 0_E, then R is not in the support of $\mathrm{Div}\, v_{P,Q}$ and we have a value with valuation 0: $v_{P,Q}(R) = x_R - x_{P+Q}$;*

- *If $R = 0_E$ then we have a value with valuation -2. By definition, since the uniformiser at 0_E is the function y/x:*
$$v_{P,Q}(0_E) = \frac{x - x_{P+Q}}{(y/x)^{-2}}(0_E) = \frac{x^2(x - x_{P+Q})}{y^2}(0_E) = 1$$
because $y^2 = x^3 + ax + b$;

- *If $R = P + Q$ or $R = -P - Q$ but $P + Q \neq -P - Q$ (or in other words $P + Q$ is not a point of two torsion), then we have a value with valuation 1. The uniformiser is $x - x_R$ because $f(x_R) \neq 0$ since R is not a point of 2-torsion, and the value is*
$$v_{P,Q}(R) = \frac{x - x_{P+Q}}{x - x_R}(x_R) = 1$$
because in this case $x_R = x_{P+Q}$;

- *If $R = P + Q$ and $P + Q$ is a point of 2-torsion, then this time we have a value with valuation 2. Since $f(x_R) = 0$ the uniformiser is y, so we have*
$$v_{P,Q}(R) = \frac{x - x_{P+Q}}{y^2}(x_R) = \frac{1}{f'(x_{P+Q})}.$$
Indeed if we write $f(x) = (x - x_{P+Q})g(x)$, then since $y^2 = f(x)$ we have $\frac{x - x_{P+Q}}{y^2}(x_R) = \frac{1}{g(x_{P+Q})}$, and we compute $f'(x) = (x - x_{P+Q})g'(x) + g(x)$ so that $f'(x_{P+Q}) = g(x_{P+Q})$.

The extended value of $l_{P,Q}(R)$ is given by the following cases (taking into account that $\mathrm{Div}(l_{P,Q}) = [P] + [Q] + [-P - Q] - 3[0_E]$):

- *If R is different from P, Q, $-P - Q$ or 0_E, then R is not in the support of $\mathrm{Div}\, l_{P,Q}$ and we have a simple value with valuation 0: $l_{P,Q}(R) = y_R - \alpha x_R - \beta$;*

- *If $R = 0_E$ then we have a value with valuation -3 and*
$$l_{P,Q}(0_E) = \frac{y - \alpha x - \beta}{(x/y)^{-3}}(0_E) = \frac{(y - \alpha x - \beta)x^3}{y^3}(0_E) = 1;$$

- *If $R = P$ or $R = Q$ or $R = -P - Q$ but $l_{P,Q}$ is not tangent to E at R, then we have a value with valuation 1. If R is not a point of two torsion then the uniformiser is $t_R = x - x_R$ and the value is*
$$l_{P,Q}(R) = \frac{y - \alpha x - \beta}{x - x_R}(R) = \frac{y - y_R - \alpha(x - x_R)}{x - x_R}(R) = \frac{y - y_R}{x - x_R}(R) - \alpha = \frac{f'(x_R)}{2y_R} - \alpha.$$
If R is a point of two torsion, then the uniformiser is $t_R = y$ and the value is
$$l_{P,Q}(R) = \frac{y - \alpha x - \beta}{y}(R) = 1 - \alpha\frac{x - x_R}{y}(R) = 1.$$

- *If $R = P$, $R = Q$ or $R = -P - Q$, and $l_{P,Q}$ is tangent to E at R but is not an inflection point, then we have a value of valuation 2. In this case R cannot be a point of two torsion so the uniformiser is $t_R = x - x_R$. To compute the value we must compute the formal series corresponding to y in the completion of $K[E]$ along $x - x_R$ up to order 2: $y = y_R + \alpha(x - x_R) + \alpha_2(x - x_R)^2 + O(x - x_R)^3$. We have $\alpha_2 = \frac{f''(x_R)/2 - \alpha^2}{2y_R}$, so the value is*
$$l_{P,Q}(R) = \frac{y - y_R - \alpha(x - x_R)}{(x - x_R)^2}(R) = \alpha_2.$$

- *Finally when R is an inflection point of f, so that $R = P = Q = -P - Q$ (and in particular is a point of 3-torsion), then we have a value with valuation 3. We compute the formal series corresponding to y in in the completion of $K[E]$ along $x - x_R$ up to order 3: $y = y_R + \alpha(x - x_R) + 0(x - x_R)^2 + \alpha_3(x - x_R)^3 + O((x - x_R)^4)$. We have $\alpha_3 = \frac{1}{2y_R}$ and*
$$l_{P,Q}(R) = \frac{y - y_R - \alpha(x - x_R)}{(x - x_R)^3}(R) = \alpha_3.$$

Combining these values we can now compute the extended value of $\mu_{P,Q}(R)$ *(taking into account that* $\mathrm{Div}(\mu_{P,Q}) = [P] + [Q] - [P+Q] - [0_E]$*):*

- *When* R *is not equal to* P, Q, $P+Q$, $-P-Q$ *or* 0_E *then the valuation is* 0 *and we have a simple value:*

$$\mu_{P,Q}(R) = \frac{y_R - \alpha x_R - \beta}{x_R - x_{P+Q}}. \tag{3.8}$$

  (If $R = -P-Q$ *and* R *is not in the support of* $\mathrm{Div}(\mu_{P,Q})$ *then the valuation is also* 0 *but Equation* (3.8) *is not well defined so to compute the value we need to look at the particular cases above);*

- *When* $R = 0_E$ *the valuation is* -1 *and we have*

$$\mu_{P,Q}(0_E) = 1. \tag{3.9}$$

  Since the value is 1 *we see that the function* $\mu_{P,Q}$ *is indeed normalised at* 0_E*;*

- *For all the other cases we refer to the study of the special cases done for* $v_{P,Q}$ *and* $l_{P,Q}$ *above.*

Finally, when $P = -Q$ *(but* $P \neq 0_E$*) so that* $P + Q = 0_E$*, then* $\mu_{P,Q} = x - x_P$ *and the extended value of* $\mu_{P,Q}$ *at* R *is given by the same formulae as the study of* $v_{P,Q}(R)$ *above.*

3.6 Conclusion and perspectives

We have seen how the reformulations of the Tate and Weil pairing on Jacobians which are better suited for computations are valid on a general abelian variety using Lang's reciprocity (Proposition 3.2.1). The natural generalisation of Miller's algorithm then depends on an explicit version of the theorem of the square.

For an abelian variety given by a theta model, this explicit version is encoded by the differential addition (and Riemann relations), and Miller's algorithm gives in this case a very streamlined and deterministic algorithm, which readily adapts to other versions (ate, optimal ate, twisted ate, …). Furthermore these algorithms naturally adapt on a model of level $n = 2$, hence on Kummer varieties.

There is a big challenge remaining however: the formulation of Proposition 3.2.1 is valid only when computing the Weil or Tate pairing related to the isogeny $[\ell]$. For a general isogeny $f : A \to B$, the explicit definitions of [Rob21, Sections 4.1.1 and 4.2.1] given by [Rob21, Equation (4.2)] using the function $g_{f,P}$ are way too slow to be used if f is of large degree. For the Tate-Cartier pairing, this is not a problem, because by [Rob21, Remark 4.2.2] it does not really depends on the isogeny. So computing the Tate-Cartier pairing $e_{T,f}$ on $B(k)/A(k) \times \widehat{K}(k)$ is the same as computing it using $e_{T,\ell}$ on $B(k)/\ell B(k) \times \widehat{B}[\ell]$ via the obvious maps (here f is an isogeny of exponent ℓ).

Computing the Weil-Cartier pairing e_f is much more challenging. Of course, if $\tilde{f}$ is the contragredient isogeny, and $\hat{\tilde{f}}$ its dual, then by compatibility of the Weil-Cartier pairing with isogenies we have that $e_f(P,Q) = e_\ell(P, Q_0)$ for any $Q_0 \in \widehat{A}[\ell]$ such that $\hat{\tilde{f}}(Q_0) = Q$, but computing Q_0 is too expensive. Looking at divisors, we have that $f_{\ell,P} \circ f = g^\ell_{f,P}$, but of course taking the ℓ-th power of $g_{f,P}$ kills the pairing and it is not obvious how we could take a ℓ-th root of $f_{\ell,P} \circ f$.

We could try to follow Corollary 3.3.3 and compute an affine version of the isogeny f and its dual using Chapter 4. But it is not clear how we could relate the point $P \in \mathrm{Ker}\, f$ and $Q \in \mathrm{Ker}\, \hat{f}$ in this manner.

So this is an exciting problem, in which I am interested because I think fast computation of the Weil-Cartier pairing could have many applications for isogeny related cryptosystems, like SIDH or the ones giving VDF. For instance, pairings have already been used to compress SIDH keys [CJL+17; ZSP+18; NR19], but maybe this could be further improved.

An intermediate interesting problem is the following: if β is a totally positive real element, we also have the β-pairing on $A[\beta]$. Of course, if $\ell = \beta\beta'$, we can compute $e_\beta(P,Q)$ as $e_\ell(P',Q)$ where $\beta' P' = P$, but we should be able to compute the β-pairing directly.

Finally, there is an exciting proposal for a trilinear pairing [Hua18; Hua19], which is essentially $e : \ell NS(A) \times A[\ell] \times A[\ell] \to \mu_\ell$, $(\mathcal{L}, P, Q) \mapsto e_{\mathcal{L}}(P,Q)$. It would be interesting to use the methods of this Chapter to improve the computation this map.

4 ISOGENIES

CONTENTS

4.1 INTRODUCTION

In this Chapter, we extend the two methods we have seen for computing isogenies in the theta model in Section 2.10 to give a general isogeny framework, whenever we have an explicit version of the theorem of the square (see Section 4.2).

We have seen in Sections 2.1 and 2.9 that this is the case in the theta model and the Jacobian model. A specific case is given by the various models of elliptic curves, namely each time we have an explicit pairing algorithm in this model, we have an explicit version of the theorem of the square, hence an isogeny algorithm. But elliptic curves are special, in that we can simply use translate of divisors by the points of the kernel and descend affine coordinates by taking their trace under translation, see Example 4.2.1.

The two methods are then developed for ℓ-isogenies in Sections 4.3 and 4.4. Adapting these methods to compute cyclic isogenies is described in Section 4.5, and a modular interpretation of the isogeny formula is given (for the theta model) in Section 4.6.

In Section 4.7 we describe an alternative strategy based on differential equations, and give details for Jacobians of hyperelliptic curves of genus 2. Finally, Section 4.8 give some perspectives.

4.2 A GENERIC FRAMEWORK FOR ISOGENIES

Let A be an Abelian variety of dimension g over the field k. Let K be a finite subgroup scheme of A. A classical theorem [Mum70a] guarantees the existence of $B = A/K$, the isogeny $f : A \to B$ is then faithfully flat. We refer to [Rob21, Sections 2.2.3 and 2.3.4] for more details. The object of this section is to give a general framework to compute B and the isogeny $f : A \to B$. We will restrict to the case of étale kernels K (ie separable isogenies). Of course, it is necessary to make precise how A and K are given.

As we want to make explicit computations with A considered as an algebraic variety, we consider that A is given with a very ample line bundle $\mathcal{L}_A$, hence a projective embedding $i : A \to \mathbb{P}^m_k$. For the kernel K, it will typically

be given by its geometric points $i(K(\bar{k})) \subset \mathbb{P}^m_k$ or by equations in $\mathbb{P}^m_k$. Likewise the output B will be (at least implicitly) represented by an embedding into a projective space, so in particular by a very ample bundle $\mathcal{L}_B$ and global sections of $\mathcal{L}_B$. An explicit algorithm then explains how, starting from the coordinates of $P \in A$ given by $\mathcal{L}_A$, to compute the coordinates of $f(P) \in B$ given by $\mathcal{L}_B$.

By descent theory (Theorem 2.5.1), $(B, \mathcal{L}_B)$ is completely determined by $f^*\mathcal{L}_B$ and the global sections of $\mathcal{L}_B$ corresponds to sections of $f^*\mathcal{L}_B$ invariant by translation by K.

It remains to see how we can endow B with such a polarisation $\mathcal{L}_B$. We discuss two possibilities. We warn that given the level of generalities we place ourselves here, the discussion in the following paragraphs has to be somewhat generic. Concrete equations and algorithms will depend on the model of A and B we choose to work with. In the rest of this Chapter, we will give such concrete algorithms when A and B are represented by their theta model. The point of the following discussion is to show that there is a broad approach to isogeny computations, generically valid in every model, and that it is only a small subset of this approach that depends on the concrete models chosen.

The *first possibility* is to construct from $\mathcal{L}_A$ a line bundle $\mathcal{L}'_A$ on A which descends to B. From Theorem 2.5.1 we know this will be the case if K is isotropic for $e_{\mathcal{L}'_A}$. The descents of $\mathcal{L}'_A$ to line bundles $\mathcal{L}_B$ on B then corresponds to lifts of K into the theta group $G(\mathcal{L}'_A)$. If we ask $\mathcal{L}_B$ to be symmetric, we restrict to symmetric lifts of K. If K contains points of 2-torsion, then these don't always exist for $\mathcal{L}'_A$ but we may always translate $\mathcal{L}'_A$ such that symmetric lifts exists. If K does not contains points of 2-torsion, then not only a symmetric lift exists, it is unique, so the symmetric $\mathcal{L}_B$ is canonical from the choice of $\mathcal{L}'_A$. We refer to Section 2.6 for more details.

Let us switch here to the language of divisors to make the difference between isomorphic and equal line bundles clearer, so that we can illustrate the following key point. Let Θ_A be a divisor representing $\mathcal{L}_A$. From Θ_A we wish to construct a divisor Θ'_A such that K is isotropic for $e_{\Theta'_A}$ so Θ'_A descends to a divisor Θ_B on B. Then the pullback $\Theta''_A = f^*\Theta_B$ is a divisor which is invariant by translation by K. However, Θ'_A is only linearly equivalent to Θ''_A, not equal, and need not be invariant by translation. Here we see the crucial importance of the theta group $G(\Theta'_A)$: lifting K in $G(\Theta'_A)$ amount to choosing sections of $t_P^*\Theta'_A - \Theta'_A$ in a coherent way.

Example 4.2.1. A very important example is the case of elliptic curves. Isogeny formulae were given by Vélu in [Vél71]. If E is an elliptic curve and 0_E its neutral point, the standard Weierstrass model is given by the sections x, y of the divisor $3(0_E)$, where x is of valuation -2 at 0_E and y of valuation -3.

Let K be a cyclic (étale) kernel of $E[\ell]$, and $f : E \to E' = E/K$ be the isogeny. Then the divisor $f^*(0_{E'})$ is equal to $D_K = \sum_{T \in K(\bar{k})}(T)$. The coordinates x and y are sections of $3D_K$, and the action of $\widetilde{K}$ on them is simply given by the translations by T (ie, with the notations of **??** 4.3.2.(iii), we choose $g_P = 1$ for all P, this is obviously symmetric). Hence traces under $\widetilde{K}$ are simply $X(P) = \sum_{T \in K(\bar{k})} x(P+T)$ and $Y(P) = \sum_{T \in K(\bar{k})} y(P+T)$, these give Weierstrass coordinates on E'. (For normalisation reasons, Vélu translates $X(P)$ by $-\sum_{T \in K(\bar{k}), T \neq 0_E} x(T)$ and similarly for $Y(P)$). Vélu then recovers the equation of E' by developing X and Y along the uniformizer $z = -x/y$ at 0_E.

We note the usual difference between the odd case and even case: D_K is always algebraically equivalent to $\ell(0_E)$, but is only linearly equivalent to it if ℓ is odd.

If we start with an elliptic curve given by some model (not necessarily Weierstrass), and want to compute $E' = E/K$ in some other model corresponding to sections of a divisor D', it suffices to recover them as invariant sections of the divisor $D_K = f^*D'$. This may be done using Miller's algorithm to find sections of D_K and then taking traces under K of these sections. This reduces to constructing the function $\mu_{P,Q}$ with divisor $(P) + (Q) - (P+Q) - (0_E)$ on our model E (eventually normalised at 0_E). Since pairings have been worked out of a lot of models of elliptic curves, we can derive isogeny algorithms for these models.

It is tempting to extend this strategy for abelian varieties: namely given a divisor Θ on A, and K maximal isotropic for $e_{\ell\Theta}$, the divisor $\Theta_K = \sum t_P^*\Theta$ certainly descends to $B = A/K$. And we can get invariant sections of Θ_K by taking trace of sections of Θ under translation by K. Unfortunately Θ_K is algebraically equivalent to $\ell^g\Theta$, so if Θ' is a divisor on B such that $f^*\Theta'$ is algebraically equivalent to $\ell\Theta$, the descent of Θ_K to B is algebraically equivalent to $\ell^{g-1}\Theta'$. So we do not get a divisor of the same level we started with, except if $g = 1$.

So instead we will descend $\ell\Theta$, but this divisor is not invariant by translation by points of K (only linearly equivalent to its translate by points of K), so it appears that whenever $g > 1$ we always need to make explicit the action of the theta group $G(\ell\Theta)$, while we could hide it in the case of elliptic curves.

The *second method*, if $K \subset A[\ell]$ is to look at the ℓ-contragredient isogeny $\tilde{f} : B \to A$. Then $\mathcal{L}'_B = \tilde{f}^*\mathcal{L}_A$ is a line bundle on B. Furthermore if g is a section of $\mathcal{L}_A$, then $g \circ f$ is a section of $\mathcal{L}'_B$, so it is easy to construct sections. Since $f = [\ell] \circ \tilde{f}^{-1}$, we may then try to descend $\mathcal{L}'_B$ to a line bundle $\mathcal{L}_B$ of smaller level along the isogeny $[\ell]$.

Remark 4.2.2. For the first method, from Example 4.2.1 we saw that constructing isogenies is linked to the explicit action of $G(\mathcal{L}_A^\ell)$. For instance, if we have a basis of sections of $\mathcal{L}_A^\ell$ and the action of $\widetilde{K}$, then taking the traces give generators of the sections of $\mathcal{L}_B$. If, supposing n is prime to ℓ for simplicity (the same methods as in Remarks 2.10.3, 2.10.7 and 2.10.14 works for the general case), we also have the explicit action of $G(\mathcal{L}_A^\ell)$ above $A[n]$, then given one non zero section s invariant under $\widetilde{K}$ (eg constructed via a trace), we can use Recipe 2.5.8 to construct (given a symplectic basis of $A[n]$) a basis of theta functions of level n on $(B, \mathcal{L}_B)$. Finally of course given the full action of $G(\mathcal{L}_B^\ell)$, we can use Recipe 2.5.3 to build a basis of theta functions of level $n\ell$. Then we can apply Mumford's isogeny theorem (Theorem 2.5.6) to construct the isogeny. When $(A, \mathcal{L})$ is already given by its theta model of level n, this is exactly the strategy of Section 2.10.2, where we give the action of $G(\mathcal{L}^\ell)$ on product of sections of $\mathcal{L}$.

Likewise, in the second method, we have seen that it is easy to build some sections of $\mathcal{L}'_B$, hence if we have the action of $G(\mathcal{L}'_B)$ we can construct theta functions of level $n\ell$ on $(B, \mathcal{L}'_B)$, which we then need to descend to theta functions of level n on $(B, \mathcal{L}_B)$. When $(A, \mathcal{L})$ is given by its theta model of level n, we can also reformulate the strategy of Section 2.10.1 from this point of view: when $P \in K$, the excellent lift $\theta_i(\widetilde{y+P})$ correspond to the (symmetric) action above a preimage P' of P by the contragredient isogeny $\tilde{f}$ in $B[\ell]$ on the section $\theta_i \circ f$. The action of $G(\mathcal{L}'_B)$ above a point P'_0 in $B[n]$ is directly given by the action above $P_0 = f(P'_0)$. Since $\theta_0 \circ f$ is invariant by $B_2[n\ell]$, these actions are sufficient to construct the basis of theta functions of level $n\ell$ on $(B, \mathcal{L}'_B)$. In particular, given $Q' \in \mathrm{Ker}\tilde{f} = B_2[\ell]$, the action of the symmetric lift above Q' on $\theta_i(\widetilde{y+P})$ is given by the multiplication by $e_{\mathcal{L}_B^\ell}(Q', P')$.

The only non canonical choice lies in the choices of the P' for the $P \in K$, which is implicit in the choice of the affine lift $\widetilde{P}$. On A, they are completely determined by a choice of $P'' \in A[\ell^2]$ such that $P = \ell P''$, via $P' = f(P'')$. Taking an excellent lift $\widetilde{P''}$ of P'', there are ℓ^2 choices, which all give the same excellent lift $\widetilde{P}$ of P.

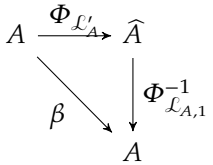
4.3 Descending line bundles on A to line bundles on B

We first look at the strategy to find a line bundle $\mathcal{L}'_A$ on A which is equal, or simply isomorphic, to a line bundle of the form $f^*\mathcal{L}_B$.

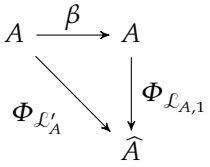
In general there is no hope that $\mathcal{L}_A$ itself is of this form. Indeed, as mentioned in Chapter 2, we usually take $\mathcal{L}_A := \mathcal{L}_{A,n} = \mathcal{L}_{A,1}^n$ with $n = 3, 4$ in order to work with as few coordinates as possible (or even $n = 2$ if working with the Kummer variety is sufficient), so $\deg \mathcal{L}_{A,n} = n^g$. But $\deg f^*\mathcal{L}_B = \deg f \deg \mathcal{L}_B$, so if f is an ℓ-isogeny and we want $\mathcal{L}_B$ to be of the form $\mathcal{L}_{B,n} := \mathcal{L}_{B,1}^n$ where $\mathcal{L}_{B,1}$ is principal, we need to look for a line bundle $\mathcal{L}'_A$ of level ℓn, hence of degree $(\ell n)^g$. An obvious candidate when ℓ is prime to n is to look for $\mathcal{L}'_A = \mathcal{L}_{A,n}^\ell$, and assume that K is totally maximal isotropic for the Weil pairing induced by $\mathcal{L}_{A,1}^\ell$ on $A[\ell]$ (or $\mathcal{L}_{A,n}^\ell$ restricted to $A[\ell]$). If ℓ is not prime to n it suffice of course to take a power $\ell/n \wedge \ell$.

4.3.1 *Constructing other line bundles*

Since we want to deal with more general type of isogenies than ℓ-isogenies, let us first work out the type of polarisations we can construct on A. For simplicity, we assume that $(A, \mathcal{L}_A)$ is of level n, hence comes from the n-th power of a principal polarisation $\mathcal{L}_{A,1}$ (see Section 2.3). Let $\mathcal{L}'_A$ be another polarisation, and $\beta = \Phi_{\mathcal{L}_{A,1}}^{-1} \circ \Phi_{\mathcal{L}'_A} : A \to A$ the endomorphism making the following diagram commute:



By construction β commutes with the Rosati involution induced by $\mathcal{L}_{A,1}$, and is totally positive since $\mathcal{L}'_A$ is ample. Conversely, if β is a totally positive real element of $\mathrm{End}(A)$, we can construct a morphism $\Phi_{\mathcal{L}'_A} : A \to \widehat{A}$ as follow:



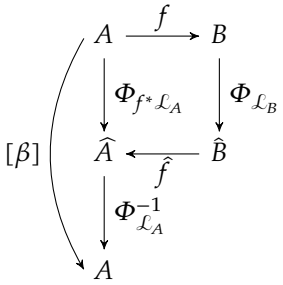
Since the pairing induced by $\Phi_{\mathcal{L}'_A}$ is skew symmetric, the map $\Phi_{\mathcal{L}'_A}$ is indeed a polarisation, ie comes from a line bundle $\mathcal{L}'_A$. Indeed, if $\mathcal{L}''_A = (\mathrm{Id}\times\Phi_{\mathcal{L}'_A})^*\mathcal{P}$, where $\mathcal{P}$ is the Poincare bundle on $A\times\widehat{A}$, it is easy to check using skew symmetry that $\Phi_{\mathcal{L}''_A} = 2\Phi_{\mathcal{L}'_A}$, hence $\mathcal{L}''_A$ descends to a line bundle $\mathcal{L}'_A$ (see [Mil91, Proposition 16.6]).

We call this line bundle $\mathcal{L}^{\beta}_{A,1}$, by analogy with the case $\beta = [n]$ where the corresponding line bundle is $\mathcal{L}^{n}_{A,1}$. Be careful that $\beta^*\mathcal{L}_{A,1} \simeq \mathcal{L}^{\beta^2}_{A,1}$, and that $\mathcal{L}^{\beta}_{A,1}$ is only defined up to algebraic equivalence class. We can rigidify our choice by requiring $\mathcal{L}^{\beta}_{A,1}$ to be symmetric. In practice we will rather work with line bundles of the form $\mathcal{L}^{\beta}_{A,n}$ with $\mathcal{L}_{A,n}$ of even level n (we can define $\mathcal{L}^{\beta}_{A,n}$ either as ${\mathcal{L}^{\beta}_{A,1}}^{n}$ or as induced by the polarisation $\Phi_{\mathcal{L}_{A,n}}\circ[\beta]$), so we can even rigidify the isomorphism class of $\mathcal{L}^{\beta}_{A}$ as to be the only totally symmetric line bundle in its algebraic equivalence class. Alternatively, still with even level, fixing once and for all $\mathcal{L}_{A,n/2}$ such that $\mathcal{L}^{n}_{A} = \mathcal{L}^{2}_{A,n/2}$ (eg $\mathcal{L}_{A,n/2} = {\mathcal{L}_{A,1}}^{n/2}$), we may even define $\mathcal{L}^{\beta}_{A}$ uniquely via the formula $\mathcal{L}^{\beta}_{A} = (\mathrm{Id}\times\Phi_{\mathcal{L}_{A,n/2}}\circ\beta)^*\mathcal{P}$

To sum up: there is a bijection between algebraic equivalence class of ample line bundles on A and totally positive real elements in $\mathrm{End}^s(A)$ where $\mathrm{End}^s(A)$ denotes the endomorphisms invariant under the Rosati involution. More generally there is a bijection between $\mathrm{End}^s(A)$ and $NS(A)$.

If $\mathcal{L}_A$ is not principal, there is still an action of β on it which defines $\mathcal{L}^{\beta}_{A}$ up to algebraic equivalence.

Definition 4.3.1. We say that an isogeny $f : (A,\mathcal{L}_A)\to(B,\mathcal{L}_B)$ is a β-isogeny if $f^*\mathcal{L}_B \simeq \mathcal{L}^{\beta}_{A}$. This implies that K is isotropic for $e_{\mathcal{L}^{\beta}_{A}}$, and the following diagram commutes:



If A has a principal polarisation $\mathcal{L}_{A,1}$, we also say that an isogeny $f : A\to B$ is a β-isogeny if its kernel $K = \mathrm{Ker}\, f$ is maximal totally isotropic for the β-Weil pairing $e_{\mathcal{L}^{\beta}_{A,1}}$ on $A[\beta]$. Indeed in this case $\mathcal{L}^{\beta}_{A,1}$ descends to a principal polarisation $\mathcal{L}_{B,1}$ and $(A,\mathcal{L}^{\beta}_{A,1})\to(B,\mathcal{L}_{B,1})$ is a β-isogeny.

4.3.2 *The algorithm*

So this suggests the following strategy, for a β isogeny, where we assume for simplicity that the level n of $\mathcal{L}_A = \mathcal{L}_{A,n}$ is prime to the degree of β.

Algorithm 4.3.2. (i) Starting from $\mathcal{L}_A$ and its sections g_i, compute a basis of sections t_i of $\mathcal{L}^{\beta}_{A}$;
(ii) Compute the action of $G(\mathcal{L}^{\beta}_{A})$ on this basis;
(iii) Find a lift $\widetilde{K}\subset G(\mathcal{L}^{\beta}_{A})$ of K;
(iv) Identify a basis of invariant sections s_i by $\widetilde{K}$; This defines a descent $\mathcal{L}_B = \mathcal{L}_{B,n}$ of $\mathcal{L}_A$ to B.
(v) Compute the equations of f, by expressing the s_i as rational functions in the g_i. Possibly compute equations of B in the s_i coordinates too.

We detail these steps, when $\beta = \ell$:

- for **??** 4.3.2.(i) we may use a Miller type algorithm to construct sections on $\mathcal{L}_A^\ell$: see Sections 2.9 and 3.3. Alternatively, since the multiplication map $\Gamma(\mathcal{L}_A)^\ell \to \Gamma(\mathcal{L}_A^\ell)$ is surjective whenever $n \geq 3$, we may simply construct the t_i as ℓ-fold products of g_i. (When working with $n = 2$, ie on the Kummer variety K_A, the multiplication map is surjective into even sections, that is sections invariant by the action of $[-1]$ whenever K_A is projectively normal, see Section 2.12. But since all sections of $\mathcal{L}_{B,n}$ are even if $n = 2$, these even sections are enough to recover the sections invariant by $\widetilde{K}$.)

  This gives us several constructions of sections (see Section 2.9 for more details):

  1. Using Miller's algorithm to construction sections associated to the cycles $Z = \ell(P) - \ell(0)$ for $P \in A[\ell n]$. If ℓ is prime to n it suffices to consider $P \in A[\ell]$ (along with the sections of $\mathcal{L}$ we have).
  2. Use $Z = b(aP) + a(-bP) - \ell(0)$ for $a + b = \ell$ and P a random point.
  3. Use ℓ-fold products of sections of $\mathcal{L}$. This can eg be seen as a particular case of Miller's algorithm applied to cycles of the form $Z = \sum_{i=1}^{\ell}(P_i) - \ell(0)$, $P_i \in A[n]$, since $(P_i) - (0)$ corresponds to a section of $\mathcal{L}$.
  4. More generally use $\prod_{i=1}^{k} g_i(n_i x)$ where g_i are sections of $\mathcal{L}$ and $\sum n_i^2 = \ell$. Again, these can be seen as functions associated to the cycles $\sum [n_i]^*((P_i) - (0))$.

- We explained how to do **??** 4.3.2.(ii) in Section 2.9. Essentially we use the explicit version of the theorem of the square to generate sections g_P of $\ell t_P^* \Theta - \ell\Theta$ for $P \in K$, which induce elements g_P of $G(\mathcal{L}_A^\ell)$ via $\mathsf{g}_\mathsf{P} \cdot s = x \mapsto g_P(x)s(x - y)$.

- For **??** 4.3.2.(iii), we normalize these sections g_P so that they generate a lift $\widetilde{K}$, so in particular $\mathsf{g}_\mathsf{P}{}^\ell = 1$ which determines g_P up to a ℓ-th root of unity ζ. If $\mathcal{L}$ is symmetric, we can rigidify the choice of lifts by considering only symmetric lifts, that is lifts g_P of P such that $[-1]^* g_P = (g_P \circ t_P)^{-1}$. Then when ℓ is odd, there is a unique symmetric lift g_P of ℓ-torsion, hence there is a canonical lift. This is because the other symmetric lift is $-g_P$, which cannot be of order ℓ if g_P is of order ℓ. If ℓ is not odd, we have two possibilities for each P, and making a choice for a basis $(P_1, \dots, P_g)$ of K then yield a symmetric lift $\widetilde{K}$ by the same reasoning as in Section 2.9. This is a crucial difference with the odd case: in the odd case we can normalize each section g_P independently since the normalisation is unique, we don't need to take a basis. See Section 4.4.4 for some consequences.

  Once we have chosen a lift $\widetilde{K}$, the other lifts are given by the conjugation action by $Q \in A[\ell n] = K(\mathcal{L}_A^\ell)$, which is acting on a lift g_P by $e_{\mathcal{L}_A^\ell}(P, Q)$. Lifts of K thus form a torsor under $A[\ell n]/K^\perp$, where $K^\perp$ is the orthogonal of K under $e_{\mathcal{L}_A^\ell}$ (see Theorem 2.5.1 for more details). If n is prime to ℓ, then $A[\ell n]/K^\perp = A[\ell]/K$. If $\widetilde{K}$ is symmetric, the conjugation action by Q is symmetric if and only if $Q \in A[2]$.

- Then **??** 4.3.2.(iv) can be done by linear algebra. In fact it is often convenient to simply take a random section $t \in \Gamma(\mathcal{L}_A^\ell)$ and take the trace s of t under $\widetilde{K}$. As long as ℓ is prime to the characteristic of k, we obtain all invariant sections this way. This is convenient when we want to evaluate an invariant section at a point. A slight problem is that to take the trace requires taking the list of all geometric points of K. But if K is only given by equations, we might as well work on the formal point of K (this is the generic point if K/k is irreducible), and compute the trace via a resultant. Concrete details of this step depends on the model of A and the representation of K. When A is given by a theta model it is easy to adapt the methods of [LR15b], see Section 4.4.4.

  We note that $G(\mathcal{L}_B) = Z(K)/\widetilde{K}$, so once we have an invariant section s, we may compute the action of $g \in G(\mathcal{L}_B)$ on it via the action of any representative in $Z(K) \subset G(\mathcal{L}_A^\ell)$. In particular, we may apply Recipe 2.5.3 to compute theta functions for B. This also shows that once we have the invariant section s, the action $g \cdot s$ for g a representative of $Z(K)/\widetilde{K}$ give generators of $\Gamma(B, \mathcal{L}_B)$.

  More generally, under the algorithmic hypothesis of Section 2.9 we also have Algorithmic Hypotheses 2.9.2.(i) and 2.9.2.(ii) for $(B, \mathcal{L}_B)$. Indeed, if $Z = \sum(Q_i)$ is a 0-cycle on B lineary equivalent to 0, then $f^* D_{\Theta_B, Z} \simeq D_{\ell\Theta_A, f^{-1}Z}$ where $\ell\Theta_A \simeq f^*\Theta_B$ (where the linear equivalence is given by the choice of $\widetilde{K}$), and $f^{-1}Z = \sum(P_{i,j})$ with $f(P_{i,j}) = Q_i$. Then the realisation $S(f^{-1}Z)$ is in K if $S(Z) = 0_B$ or is in $K^\perp \subset K(\mathcal{L}_A^\ell)$ if $S(Z) \in K(\mathcal{L}_B)$. We may thus compute a section for $D_{\ell\Theta_A, f^{-1}Z}$ and keeping track of the linear equivalence between $\ell\Theta_A$ and $f^*\Theta_B$ (induced by $\widetilde{K}$), transform it into a section of $f^* D_{\Theta_B, Z}$, which descends to B. From the point of view of the dual abelian varieties, we simply use the fact that $\Phi_{\mathcal{L}_A^\ell} = \hat{f} \circ \Phi_{\mathcal{L}_B} \circ f$.

- **??** 4.3.2.(v) is trickier than it looks if one wants a quasi-linear algorithm. Indeed, to get equations for the isogeny f, it suffices to evaluate the sections s_i on the generic point of A (expressed in the g_i coordinates). But they are constructed as traces under the action of $\widetilde{K}$ on sections $t_i \in \Gamma(A, \mathcal{L}_A^\ell)$, so evaluating such a section s requires summing ℓ^g functions. If we work with projective coordinates, we are summing polynomial functions of total degree ℓ in terms of the g_i, and the complexity of evaluating the sum depends on how we generated our section t. For instance the situation is nicer if we generate sections of $\mathcal{L}_A^\ell$ as products of sections of $\mathcal{L}_A$, since this gives a compact representation of t. Working with affine coordinates, we would sum rational functions of total degree $O(\ell)$ in terms of the g_i/g_0 (for instance). Furthermore, computing this trace on the generic point may not be that easy unless we have an efficient representation of $k(A)$.

  We may also use an interpolation technique (or a rational interpolation in the affine setting), but it may not be possible to choose the interpolation points such that a fast interpolation technique is available, so provided we already have a basis of $\Gamma(A, \mathcal{L}_A^\ell)$ expressed as polynomials in the basis of $\Gamma(A, \mathcal{L}_A)$, this requires solving a linear system of dimension $O(\ell^g)$. And we require at least $O(\ell^g)$ points of interpolation, each evaluation costing $O(\ell^g)$ due to the trace.

  A trick due to [CE14] is to instead evaluate the s_i on one fat point $P \in A(k[\epsilon])$, with $\epsilon^2 = 0$. We may then use this evaluation to get the matrix relating $f^*\omega_B$ with ω_A. Plugging the expression of f in terms of the g_i, $(s_i) \circ f = R(g_i)$, and expressing both $f^*\omega_B$ and ω_A in terms of the dg_i, this yields a differential equation satisfied by R. Picking g uniformisers $x_1, \dots, x_g$ at 0_A, we solve this differential equation in the completion $\widehat{O}_{A,0_A} \simeq k[[x_1, \dots, x_g]]$, to recover f has a formal series via a Newton approach[1]. A rational reconstruction then allows to recover R. When done with fast algorithms, this can be done (heuristically) in time $O(\ell^g)$. See for example Section 4.7.

Example 4.3.3.

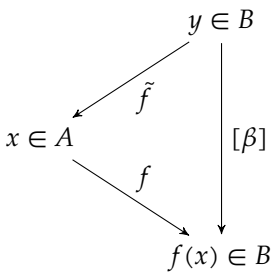
- The isogeny formula from Theorem 2.5.6 can be reinterpreted as follow: given a basis of theta functions for $\Gamma(\mathcal{L}_A^\ell)$, taking the trace of $\widetilde{K}$ for this basis gives functions of $\Gamma(\mathcal{L}_B)$. In fact Theorem 2.5.6 may be recovered by taking the trace of $\theta_0^{\mathcal{L}_A^\ell}$ and then acting by $Z(K)/\widetilde{K}$ on this trace.
- Corollary 2.10.9 can thus be seen as an application of the method of this section to the theta model: we generate sections of $\mathcal{L}_A^\ell$ via multiplication, and the differential addition readily gives the action of the theta group, hence in particular of $\widetilde{K}$. As we have seen, taking traces is then the same as applying the isogeny formula from Theorem 2.5.6 directly.
- The article [CE14] apply this method when A is a Jacobian (see also Section 2.9 for an extension of their method to compute theta coordinates).

Now if we replace ℓ by an endomorphism β, the general strategy is the same, except that it is less obvious to do **??** 4.3.2.(i), namely producing sections of $\beta\Theta$. We will go back to this in Section 4.5, and we will see that we can for instance get sections whenever we can write $\beta = \sum \gamma_i \overline{\gamma_i}$. This is a generalisation of the method that uses the decomposition $\ell = \sum n_i^2$.

4.4 Descending line bundles on B via the descent formula

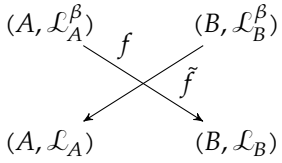
4.4.1 *The contragredient isogeny*

An alternative approach to compute a β-isogeny f, is to get a line bundle on B via the use of the contragredient isogeny $\tilde{f}$. We recall that if K is a totally maximal isotropic subgroup of $A[\beta]$, $\tilde{f} : B \to A$ is the isogeny whose kernel is $K' = f(A[\beta])$, so that $\tilde{f} \circ f = [\beta]_A$ and $f \circ \tilde{f} = [\beta]_B$. (Since $\beta(K) = 0 \subset K = \operatorname{Ker} f$, β descends to B).



[1] Eventually increasing the precision by computing the image of a point in $A(k[\epsilon]/\epsilon^m$ if needed to bootstrap the Newton process.

If we let $\mathcal{L}_B$ be a descent of $\mathcal{L}_A^\beta, f : (A, \mathcal{L}_A^\beta) \to (B, \mathcal{L}_B)$ and $\tilde{f} : (B, \mathcal{L}_B^\beta) \to (A, \mathcal{L}_A)$ are β-isogenies:



In the case that $\mathcal{L}_A = \mathcal{L}_{A,1}$ is principal and K is maximal, $\mathcal{L}_{B,1} = \mathcal{L}_B$ is a principal line bundle. Then by definition of $\mathcal{L}_{A,1}^\beta$, we see that via the identification of A and $\widehat{A}$, B and $\hat{B}$ given by $\Phi_{\mathcal{L}_{A,1}}, \Phi_{\mathcal{L}_{B,1}}$ respectively, $\tilde{f}$ corresponds to the dual isogeny $\hat{f}$. Like in Section 4.3, in practice we work with $\mathcal{L}_{A,n}$ on A, so we rather look at $\tilde{f}^* \mathcal{L}_{A,n} = \mathcal{L}_{B,n}^\beta$.

The strategy works in two steps:

1. Find sections of $\mathcal{L}_B^\beta$;
2. Descend to $\mathcal{L}_B$.

We detail these steps. For simplicity we assume for now that $\beta = \ell$ and we explain how to adapt this for more general endomorphisms in Section 4.5. We also assume that ℓ is prime to the level n of $\mathcal{L}_A$.

4.4.2 *Finding sections on the pullback*

There is an easy way to get sections of $\mathcal{L}_B^\ell$, namely the sections of the form $g \circ \tilde{f}$ where $g \in \Gamma(A, \mathcal{L}_A)$ is a section of $\mathcal{L}_A$. By Theorem 2.5.1, this gives us all sections in $\Gamma(B, \mathcal{L}_B^\ell)^{\widetilde{K'}}$, where $\widetilde{K'}$ is the lift giving $\mathcal{L}_A$. Write a symplectic decomposition $B[\ell] = K' \oplus K''$, where K'' is any symplectic supplement of K'. Then for any lift $\widetilde{K''}$ of K'' into the theta group $G(\mathcal{L}_B^\ell)$, the action of $\widetilde{K''}$ on $\Gamma(B, \mathcal{L}_B^\ell)^{K'}$ gives us the remaining sections by Theorem 2.5.1. More precisely: if $(g_1, \dots, g_{n^g})$ is a basis of $\Gamma(A, \mathcal{L}_A)$, then a basis of $\Gamma(B, \mathcal{L}_B^\ell)$ is given by $(h \cdot g_i \circ \tilde{f})_{h \in \widetilde{K''}, i \in \{1,\dots,n^g\}}$.

Concretely, if $h_{Q''}$ is the element of $\widetilde{K''}$ above $Q'' \in K''$, and $P \in B$, then in projective coordinates, $h_{Q''} \cdot (g_i \circ \tilde{f}(P)) = (g_i(\tilde{f}(P + Q'')))$.But the points $\tilde{f}(Q'')$, for $Q'' \in \widetilde{K''}$ correspond bijectively to the points $Q \in K$.

So having chosen $\widetilde{K''}$, computing coordinates given by $\mathcal{L}_B^\ell$ of $R \in B$ then corresponds to lifting (ie picking affine coordinates) in a coherent way the ℓ^g projective points $(g_i(\tilde{f}(R + Q''))) = (g_i(\tilde{f}(R) + Q))$ in A for $Q'' \in K''$ or $Q \in K$. Once again, since ℓ is odd, there is a unique choice which corresponds to $\widetilde{K''}$ symmetric. Of course there are still several choices of basis of sections corresponding to the choice of $\widetilde{K''}$ itself. We can reinterpret this in terms of trivialisation as follow: let $P = \tilde{f}(R)$. Fixing a trivialisation of $\mathcal{L}_A$ at P then gives trivialisations at $P + T$ for $T \in A_i[n]$ via the theta structure. This is not enough if $Q \in K$, because $Q \notin K(\mathcal{L}_A)$ so there is no trivialisation at $P + Q$. However via the isomorphism $\tilde{f}^* \mathcal{L}_A \simeq \mathcal{L}_B^\ell$ induced by $\widetilde{K'}$, we can fix a trivialisation of $\mathcal{L}_B^\ell$ at R, which then gives a trivialisation at $R + Q''$ for $Q'' \in K''$, which descends to a trivialisation of $\mathcal{L}_A$ at $P + Q$. Of course this choice of trivialisation at $P + Q$ depends on R. And the choice of $\widetilde{K''}$ itself is reflected in the trivialisations at the $0_A + Q, Q \in K$ induced by a trivialisation at 0_B.

We fix these trivialisations in the same manner as in Section 2.10.1, using the symmetry of $\mathcal{L}$ to fix suitable trivialisations at these points. (Namely, fixing a basis e_i of K, we fix trivialisations at the e_i, $e_i + e_j$ and $P + e_i$ that satisfy the same compatibility conditions than in Section 2.10.1, these can be determined using Algorithmic Hypothesis 2.9.2 directly). This gives compatible affine lifts of the $(g_i(\tilde{f}(R) + Q))$ for all Q at once, ie describe a basis of sections of $\Gamma(\mathcal{L}_B^\ell)$ evaluated on the point R above $P = \tilde{f}(R)$ fixed by our choices of trivialisations.

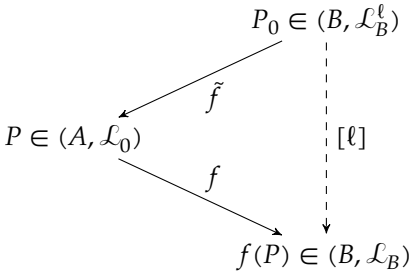
This is the strategy, formulated a bit differently that was exploited in [LR12]. It relied on an explicit theta model on A and B, but as discussed above this whole strategy works equally well with other models, under Algorithmic Hypothesis 2.9.2.

4.4.3 *Descent formula*

The remaining problem is that we want sections of $\mathcal{L}_B$ rather than sections of $\mathcal{L}_B^\ell$. More precisely, given a point P on A, we want to compute the coordinates of $f(P)$ given by the sections of $\mathcal{L}_B$. We have just seen how to get the coordinates of $f(P)$ given by sections of $\mathcal{L}_B^\ell$, these are the

$$(g_i(\tilde{f}(f(P)) + Q))_{i \in 1 \dots n^g, Q \in K} = (g_i(\ell P + Q))_{i \in 1 \dots n^g, Q \in K}. \tag{4.1}$$

So the idea is to use the multiplication by ℓ on B to relate coordinates $u \circ [\ell](P) = u(\ell P)$ with $u \in \Gamma(B, \mathcal{L}_B)$ with the coordinates $v(P)$, $v \in \Gamma(B, \mathcal{L}_B^\ell)$. Assume that we have such a formula $u \circ [\ell] = S(v)$ with S a multivariate rational function. Then we can express the isogeny $f : A \to B$ as follow: if $P \in A$, and $P_0 \in B$ is any point such that $\tilde{f}(P_0) = P$, then $\ell P_0 = f(P)$, and $u(f(P)) = u \circ [\ell](P_0) = S(v(P_0))$. And so taking for v the basis of Equation (4.1) above, this yields $u(f(P)) = S((g_i \tilde{f}(P_0) + Q))_{i \in 1 \dots n^g, Q \in K} = S((g_i(P + Q))_{i \in 1 \dots n^g, Q \in K}$. Furthermore, since we are applying $[\ell]$ when descending, our choice of P_0 (induced by our choices of trivialisations at P) does not matter.



It remains to explain how to find the rational function S. Unfortunately, $[\ell]^* \mathcal{L}_B = \mathcal{L}_B^{\ell^2}$, so we cannot use Theorem 2.5.6 directly (that's why the arrow $[\ell] : (B, \mathcal{L}_B^\ell) \to (B, \mathcal{L}_B)$ is dashed, it does not respect the polarisations).

Koizumi's formula

In [CR15], we used the following trick of Koizumi [Koi76] instead: write $\ell \operatorname{Id} = F\,{}^tF$ for an $r \times r$-matrix $F \in \mathrm{Gl}_r(\mathbb{Z})$. Concretely, since $\ell = a^2 + b^2 + c^2 + d^2$ is always a sum of four squares, we can always find such an F with $r = 4$ as the multiplication matrix of the quaternion $a + bi + cj + dk$, and whenever ℓ is a sum of two squares we can find F such that $r = 2$. Then it is easy to check that $F^* {\mathcal{L}_B}^{\star r} \simeq {\mathcal{L}_B^\ell}^{\star r}$. So on B^r, we can descend ${\mathcal{L}_B^\ell}^{\star r}$ to ${\mathcal{L}_B}^{\star r}$ along the multiplication by $[\ell]$ by first descending it along F, and then simply applying ${}^tF$ (which just amount to some arithmetic), or conversely. So we put again our good friend the Segre embedding to good usage.

In the theta model, Koizumi's formula is given via Theorem 2.5.6 by

$$F^* \theta_{(i_1,\dots,i_r)}^{\mathcal{L}_B \star \cdots \star \mathcal{L}_B} = \sum_{\substack{(j_1,\dots,j_r) \in Z(\overline{\ell n})^r \\ F(j_1,\dots,j_r) = (i_1,\dots,i_r)}} \theta_{(j_1,\dots,j_r)}^{\mathcal{L}_B^\ell \star \cdots \star \mathcal{L}_B^\ell} \tag{4.2}$$

If $X' = (X'_1, \dots, X'_r)$ is in $(B^r, \mathcal{L}_B^{\ell, \star r})$ and $Y = F(X') \in (B^r, \mathcal{L}_B^{\star r})$, then

$$\theta_{i_1}^{\mathcal{L}_B}(Y_1) \cdot \cdots \cdot \theta_{i_r}^{\mathcal{L}_B}(Y_r) = \sum_{\substack{(j_1,\dots,j_r) \in Z(\overline{\ell n})^r \\ F(j_1,\dots,j_r) = (i_1,\dots,i_r)}} \theta_{j_1}^{\mathcal{L}_B^\ell}(X'_1) \cdot \cdots \cdot \theta_{j_r}^{\mathcal{L}_B^{\star r}}(X'_r), \tag{4.3}$$

But in Equation (4.3) we can use $\tilde{f}$ to interpret the theta coordinates of level ℓn on X' in B^r of index $(j_1 + t_1, \dots, j_r + t_r)$ where $j \in Z(\overline{n})^r$ and $t \in Z(\overline{\ell})^r$ as theta coordinates of level n in A^r of $(X_1 + P_{t_1}, \dots, X_r + P_{t_r})$, where $X = \tilde{f}(X')$, using the identification $Z(\overline{\ell}) \to K$. We can rewrite Equation (4.3) as

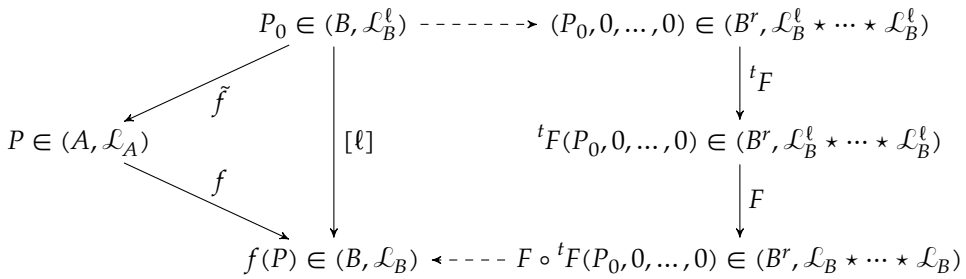
$$\theta_{i_1}^{\mathcal{L}_B}(Y_1) \cdot \cdots \cdot \theta_{i_r}^{\mathcal{L}_B}(Y_r) = \sum_{\substack{(t_1,\dots,t_r) \in Z(\overline{\ell})^r \\ F(t_1,\dots,t_r) = (0,\dots,0)}} \theta_{j_1}^{\mathcal{L}_A}(X_1 + P_{t_1}) \cdot \cdots \cdot \theta_{j_r}^{\mathcal{L}_A}(X_r + P_{t_r}). \tag{4.4}$$

where $j = F^{-1}(i) \in Z(\overline{n})^r$.

In particular, if $i \in Z(\overline{n})$ and $(j_1, \dots, j_r) \in Z(\overline{n})^r$ is the unique preimage of $(i, 0, \dots, 0)$ by F, P_0 a preimage of P by $\tilde{f}$, $X' = {}^tF(P_0, 0, \dots, 0) \in B^r$ and $X = {}^tF(P, 0, \dots, 0) \in A^r$ we get, since $Y = FX' = (f(P), 0, \dots, 0)$:

$$\theta_i^{\mathcal{L}_B}(f(P)) \cdot \cdots \cdot \theta_0^{\mathcal{L}_B}(0) = \sum_{\substack{(t_1,\dots,t_r) \in Z(\overline{\ell})^r \\ F(t_1,\dots,t_r) = (0,\dots,0)}} \theta_{j_1}^{\mathcal{L}_A}(X_1 + P_{t_1}) \cdot \cdots \cdot \theta_{j_r}^{\mathcal{L}_A}(X_r + P_{t_r}). \tag{4.5}$$

We will call this change of level method the Koizumi change of level formula. These three steps are summarized in the diagram below:



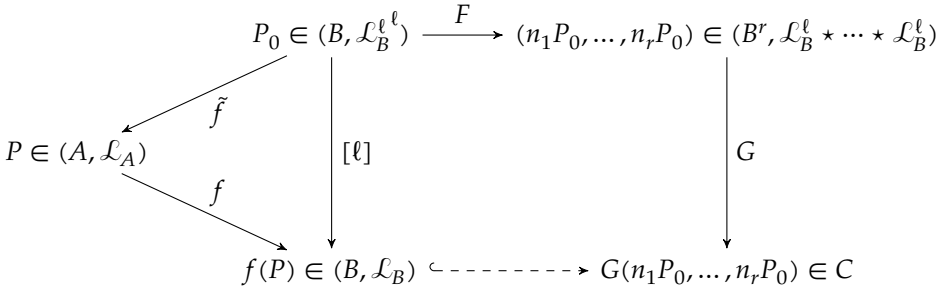
Here the computation of ${}^tF$ is done in $\mathcal{L}_B^\ell$ while we use Koizumi's formula from Equation (4.5) to compute the action of F in order to go back to level n. We could also first descend and then compute ${}^tF$.

A faster descent formula

Here lies a complexity problem: descending $\mathcal{L}_B^{\ell^{\star r}}$ along F amount to taking some traces along the kernel of F, which is of size $O(\ell^{rg})$. Using a theta model on $\mathcal{L}_B^{\ell^{\star r}}$, half of the trace action is trivial, hence need not be computed. Indeed, rather than computing the trace under the full $\operatorname{Ker} F$, we only need to compute it under $B_1[\ell]^r \cap \operatorname{Ker} F$. This reduces the complexity to $O(\ell^{rg/2})$ [CR15]. Unfortunately this still yields a quasi-quadratic algorithm whenever $r = 4$.

In Section 2.10.3, we introduced an alternative change of level using the (generalised) Segre embedding. Write $\ell = \sum_{i=1}^r n_i^2$, and $F : (B, \mathcal{L}_B^{\ell^\ell}) \to (B^r, \mathcal{L}_B^{\ell^{\star r}}), x \mapsto (n_i x)$ the generalised Segre morphism. Let $G : B^r \to C$ be the quotient of B^r by $F(B[\ell])$, so that $G \circ F : (B, \mathcal{L}_B^{\ell^\ell}) \to (B, \mathcal{L}_B)$ is simply $[\ell]$. We call this map F because this is exactly the map F from Koizumi's formula composed with the diagonal embedding $\Delta : B \to B^r$. In other words, the only difference with Koizumi's formula is that we do not quotient by the full kernel of F, but by the kernel of F intersected with the image of the diagonal embedding. This quotient behaves the same as the full quotient on $\Delta(B)$.

The whole point is that we only need the coordinates for $\mathcal{L}_B^\ell$ to express points in the image of F.



This strategy applied to the theta model gives exactly Corollary 2.10.13.

4.4.4 *Isogenies from equations of the kernel*

Another complexity problem is that the trace requires taking the geometric points of $\operatorname{Ker} F$. In [LR15b] we explain how to circumvent this when given equations of K by working with formal points. (The main difficulty, answered in this article, lies in how to compute the action of the lift $\widetilde{K}$" given only the equations of K.) It is straightforward to extend this approach to our new change of level formula.

We detail this a bit. While in Section 2.10 it was convenient to compute an excellent lift $\widetilde{K}$ of K by lifting a basis of K and then using the differential additions and three way additions to lift the other points, it actually suffices (if ℓ is odd) to compute an excellent lift $\widetilde{P}$ of P for each $P \in K$ independently. Indeed from the point of view of computing sections of $\mathcal{L}_A^\ell$ we have seen in Section 2.10.2 that the ℓ choices of excellent lifts all induce the same symmetric lift on $G(\mathcal{L}_A^\ell)$ (as expected by unicity of a symmetric lift), or from the point of view of Section 2.10.3 the different choices are killed by the descent formula. Computing $\widetilde{P}$ only involves computing the multiples of P given by $(\ell' + 1)P$ and $\ell' P$, hence can be done formally over the full kernel if we have equations for it.

Of course for this computation to be efficient we need to have a nice model of the kernel K, for instance a triangular representation. In the best cases, the kernel is parametrized by a univariate polynomial $E(X)$ (otherwise

do a random change of variable, possibly in a small extension, to separate points). All our isogenies formulae we have seen in this Chapter are given by traces, ie of the form $\sum_{P\in K(\bar{k})} W(P)$ where W is a function defined on K (typically computed via the formal normalisation of $\widetilde{P}$ above, see eg Equation (4.5)). Expressing W as $W(X) \in k[X]/E(X)$, we can compute this trace as $\sum_{P\in K(\bar{k})} W(P) = T(0)$ where T is the remainder of the euclidean division of XWE' by E.

Unfortunately I don't know how to do this when ℓ is even. We have seen in Section 2.10 that we have more choices of excellent affine lifts because there are several ways to descend $\mathcal{L}_A$ to a symmetric line bundle, since the isogeny kills part of our information on the n-level structure. This is not a problem when we have an explicit basis of K because each choice on a basis of K is induced by a symmetric theta structure on $\mathcal{L}_B$. But this does not work if we make a choice for each point of K individually. So we need to specify these choice somehow, eg encode efficiently K_0 such that K_0 is totally isotropic and $2K_0 = K$ (see Remark 2.10.14), and in this case the equation of the kernel alone is not sufficient.

4.4.5 *Summary*

We summarize the algorithm induced by this second approach:

1. Fix (implicitly) a symplectic decomposition $B[\ell] = K' \oplus K''$, where $K' = f(A[\ell]) \simeq A[\ell]/K$. Alternatively K'' can be specified from A as $f(K_\ell)$ where K_ℓ is a maximal isotropic subgroup of $A[\ell^2]$ such that $\ell K_\ell = K$;
2. Determine the action of the canonical symmetric lift $\widetilde{K''}$ on the basis of sections $\Gamma(B, \mathcal{L}_B^\ell)^{\widetilde{K'}} = \{g \circ f, g \in \Gamma(A, \mathcal{L}_A)\}$. This amount in making a coherent symmetric choice of affine lifts of the projective points $(g_i(P+Q))_{i\in 1\dots n^g}$ for $Q \in K$, where $(g_i)_{i=1\dots n^g}$ is a basis of $\Gamma(A, \mathcal{L}_A)$. We refer to Section 2.10.1 for more details in the theta model (see also Section 5.2.2).
3. Use a change of level formula to descend these sections into sections of $\mathcal{L}_B$, like for instance the formula given in Section 2.10.3.

4.5 Extending the isogeny computation to isogenies induced by real multiplication

If $\ell = \sum n_i^2$, we can use this decomposition for the first method to produce sections of $\Gamma(A, \mathcal{L}_A^\ell)$ as $\prod s_i(n_i x)$, $s_i \in \Gamma(A, \mathcal{L})$, and in the second method to descend the line bundle $\mathcal{L}_B^\ell$. We now extend this to the case of $\beta \in O_{\mathcal{F}}^{++}$ a totally positive real element, assuming for simplicity that $O_{\mathcal{F}} = \mathrm{End}^s(A)$.

Now if we replace ℓ by a totally positive β in order to compute (for instance) cyclic isogenies, we need to look at a decomposition $\beta = \sum \alpha_i^2$, with α_i totally real. In fact, we could even use a decomposition $\beta = \sum_{i=1}^r \alpha_i\overline{\alpha_i}$ for general endomorphisms α where $\overline{\alpha}$ denotes the Rosati-Involution. Our sections of $\mathcal{L}_A^\beta$ are then of the form $\prod s_i(\alpha_i x)$. From the point of view of generating sections via cycles, these corresponds to cycles of the form $\sum \alpha_i^*((P_i) - (0))$, with $P_i \in K(\mathcal{L}_A)$.

Likewise, we can use this decomposition to descend $\mathcal{L}_B^\beta$ by the isogeny $B[\beta]$ by looking at the generalised Segre morphism $F : B \to B^r, x \mapsto (\alpha_i x)$. The pullback of $\mathcal{L}_B^{\beta^{\star r}}$ by F is then $\mathcal{L}_B{}^{\beta^2}$, hence $F(B[\beta])$ is indeed isotropic.

More generally, we could look for a morphism $F : (B, \mathcal{L}_B^\beta) \to (C, \mathcal{L}_C)$ (not respecting polarisations) such that $F(B[\beta]) \subset K(\mathcal{L}_C)$ and is isotropic, ie $B[\beta]$ is isotropic for $F^*\mathcal{L}_C$ (eg $F^*\mathcal{L}_C = \mathcal{L}_B{}^{\beta^2}$). We can then apply Mumford's isogeny theorem to $F(B[\beta])$. For instance, if we know how to compute an α_i-isogeny $(B, \mathcal{M}^{\alpha_i}) \to (C_i, M_i)$ while working with coordinates from $(B, \mathcal{M})$, we could use the map $F : B \to \prod C_i, x \mapsto (\alpha_i(x))$ when $\beta = \sum \alpha_i$. The decomposition above is the special case where $\alpha_i = \gamma_i\overline{\gamma_i}$ and the α_i-isogeny is simply given by the endomorphism γ_i. This opens the path to a recursive algorithm to compute β isogenies.

Anyway, let us go back to decomposing β as a sum of squares of real endomorphism: $\beta = \sum \beta_i^2$. First, since β is totally positive, by a theorem of Siegel such a decomposition always exist with $\beta_i \in K$. If $\beta_i = \gamma_i/N$ with $\gamma_i \in O_{\mathcal{F}}$, then to evaluate $\beta_i(P)$ one needs to compute $\gamma_i(Q)$ with $P = NQ$. The result depends on Q, but at the end of the isogeny computation this does not matter since β is an endomorphism, so $N\beta$ is zero on the N-torsion.

Otherwise, the rest of the isogeny algorithm proceed the same as in the case of $\beta = \ell$. We just need to be able to compute affine lifts $\beta_i(\widetilde{x+t})$ where x is a point and t an element of the kernel. We can then either use the Frobenius and Verschiebung as in [DJR+17] or decompose β_i into sum of isogenies which we know how to compute on affine lift (in a compatible way), as in [Rob13, § 4.1]. For instance when $g = 2$, $\sqrt{d}$ is a standard d-isogeny, so we can use the formula of Section 2.10.

Remark 4.5.1. We conclude by several remarks:

- When applying the algorithm on a polarised abelian variety $(A, \mathcal{L})$ with a symmetric theta structure, it is important to take a theta structure induced by a symplectic decomposition $K(\mathcal{L}^2) = K_1(\mathcal{L}^2) \oplus K_2(\mathcal{L}^2)$ where $K_i(\mathcal{L}^2)$ is stable under the real multiplication. This way real endomorphisms respect the decomposition.

- In [DJR+17] we did not have the fast descent method yet, so we used a generalisation of Koizumi's formula. This requires finding a matrix F such that ${}^tFF = \beta$ from the decomposition $\beta = \sum_{i=1}^{r} \beta_i^2$. By standard results on Clifford Algebra, this is always possible if $r = 2^d$ is a power of two. So there are two drawbacks: first we may need to increase r, hence get a worse complexity, secondly the construction of the matrix F may introduce denominators (even if β_i are endomorphisms).

- Likewise, using a matrix F to descend $\mathcal{L}_B^{\beta^{\star r}}$ on B^r may not give a theta structure on $(B^r, \mathcal{L}_B^{\star r})$ which is a product theta structure. So in [DJR+17] we had to do a costly linear algebra (which is $O(1)$ asymptotically but very expensive in practice) to recover a product theta structure (in order to find back $(B, \mathcal{L}_B)$) via the theta transformation formula. By contrast the new descent formula descends $(B, \mathcal{L}_B^{\beta})$ directly.

- In [DJR+17], we use the Frobenius to compute the action of the real endomorphisms. We make the simplifying assumption, when computing the image of a point x by the isogeny, that x is rational and of order m prime to the denominators that appear. This allows us to ensure that $\beta_i x = \lambda_i x$ for some integer λ_i, and we can compute $\beta_i \tilde{x} = \lambda_i \tilde{x}$ via differential additions. But as explained above, we can relax these conditions, since it is easy to extend the method of [Rob13, § 4.1] which uses multiway additions to compute $\beta_i \tilde{x}$ whenever we can express β_i as a multivariate polynomial of endomorphisms we already know how to compute affinely (eg the Frobenius, or $\sqrt{d}$ in dimension 2).

- The optimal decomposition of β (reducing the size of the denominator) then becomes a number theory optimisation problem.

- From the point of view of finding sections of $\beta\Theta$, when $\beta = \sum \alpha_i$, we may use the following approach. The real multiplication allows to make sense of a line bundle associated to a cycle with coefficients given by real endomorphisms, rather than just integers (at least for a totally symmetric line bundle). If $P \in A[\beta] = \mathrm{Ker}(\mathcal{L}^{\beta})$, we would like to construct a section associated to the cycle $\beta(P) - \beta(0)$. If $\beta = \sum \alpha_i$, we can try to find sections associated to $\alpha_i(P) - (\alpha_i P) - (\alpha_i - 1)(0)$, and then use the theorem of the square to combine these to a section of $(\sum \alpha_i)(P) - ((\sum \alpha_i)P)$.

  The core difficulty of our situation, is that while for cycles with integer coefficients (linearly equivalent to 0), we may only use the theorem of the square to ultimately reduce to the zero-cycle, in our case, Algorithmic Hypothesis 2.9.2 is not enough, and we have to use the real multiplication somehow (via eg the above decompositions of β).

  For instance, I do not know yet know how to generalise the method from [CE14] which uses cycles of the form $Z = b(aP) + a(-bP) - \ell(0)$. Here of course we would instead use real endomorphisms a, b such that $a + b = \beta$, but it is not obvious how to generate the sections. Perhaps the tools used in [Hua18; Hua19] can be useful.

4.6 MODULAR INTERPRETATION OF THE ISOGENY FORMULA

When using the isogeny algorithm in the theta models, all computations have an affine version. In other words, once we have chosen rigidifications at 0_A and 0_B (ie affine points $\tilde{0}_A$ and $\tilde{0}_B$), if $f(P) = Q$, and we choose a trivialisation of $\mathcal{L}_A^{\ell}$ at P giving the affine lift $\widetilde{P}$, it is immediate to compute the affine lift $\widetilde{Q}$ giving the compatible trivialisation of $\mathcal{L}_B$ at Q.

But we have seen in Section 2.9.3 how the trivialisation of $\mathcal{L}_A$ at 0_A is induced by a choice of basis of differentials w_A of A. So a natural question is to find which basis w_B on B induce the affine lift $\tilde{0}_B$ of B induced by the isogeny and the choice of $\tilde{0}_A$. In this Section, we assume that A and B are endowed with compatible symmetric theta structures of even level.

We have the following results.

Lemma 4.6.1. *In the isogeny formula Theorem 2.5.6, if we take $\lambda = 1$ (ie $\tilde{f}(\tilde{0}_A) = \tilde{0}_B$), then if $\tilde{0}_A$ is induced by w_A, $\tilde{0}_B$ is induced by w_B such that $f^* w_B = w_A$. In other words Theorem 2.5.6 with $\lambda = 1$ naturally gives the values of the theta constant as modular forms for the normalised isogeny.*

Proof. We give an analytic proof, see also [Kem91, §5.3]: by the proof of Theorem 2.5.6, it suffices to treat isogenies of the first type. We take $A = \mathbb{C}^g/(\mathbb{Z}^g + \Omega\mathbb{Z}^g)$ and for kernel $\frac{1}{\ell}\Omega\mathbb{Z}^g/\Omega\mathbb{Z}^g$, so B corresponds to the abelian variety $B = \mathbb{C}^g/(\mathbb{Z}^g \oplus \frac{1}{\ell}\Omega\mathbb{Z}^g)$ and f is the normalised isogeny : $z \mapsto z$. The theta null point of level ℓn on A is given by $\theta\left[\begin{smallmatrix} 0 \\ \frac{i}{\ell n} \end{smallmatrix}\right](0, \frac{\Omega}{\ell n})$, while the one on B is given by $\theta\left[\begin{smallmatrix} 0 \\ \frac{j}{n} \end{smallmatrix}\right](0, \frac{\Omega/\ell}{n})$, so the normalised isogeny corresponds to compatible (affine) theta null points. □

Lemma 4.6.2 (Affine Segre)**.** *If $\tilde{0}_A$ is an affine lift of the theta null point of A given by w_A and $\tilde{0}_B$ of B given by w_B, then the Segre embedding $\theta_i(\tilde{0}_A)\theta_j(\tilde{0}_B)$ of $(\tilde{0}_A, \tilde{0}_B)$ corresponds to the differential basis (w_A, w_B) of $A \times B$.*

Proof. Analytically, this follow from the equation: $\theta\left[\begin{smallmatrix} a_1 \\ b_1 \end{smallmatrix}\right](z_1, \Omega_1)\theta\left[\begin{smallmatrix} a_2 \\ b_2 \end{smallmatrix}\right](z_2, \Omega_2) = \theta\left[\begin{smallmatrix} a_1 a_2 \\ b_1 b_2 \end{smallmatrix}\right]\left((z_1, z_2), \begin{pmatrix} \Omega_1 & 0 \\ 0 & \Omega_2 \end{pmatrix}\right)$. □

From these two results, it is easy to follow the differential basis along the steps of the isogeny algorithm, using the version of Section 4.4.3 with Koizumi's descent. Denote by $(\theta_i^A(0, \sqrt{w_A}))_{i \in Z(\overline{n})}$ the affine theta null point induced by w_A (see Section 2.9.3 for why we use this notation and [Can16] for an algebraic meaning of this notation).

Theorem 4.6.3. *Let w_A be a basis of k-rational regular differentials on A and $(\theta_i^A(0, \sqrt{w_A}))_{i \in Z(\overline{n})}$ be a modular lift. Finally, let $r = 1, 2$ or 4 depending on ℓ being a square, a sum of two square or not. Then the affine isogeny formula yields the products $(\theta_{i_1}^B(0, \sqrt{w_B}) \times \cdots \times \theta_{i_r}^B(0, \sqrt{w_B}))_{i_1, \dots, i_r \in Z(\overline{n})}$ where w_B is such that $f^* w_B = w_A$. Note that the product is uniquely defined except if $r = 1$ in which case we get all constants up to a common sign.*

Proof. This is [KNR+20b, Theorem 4.5]. □

We will apply this in Section 5.6.5 to descend algebraic modular forms along isogenies.

Unfortunately, I do not have yet such a modular result for the faster approaches of the isogeny computation. Using the first strategy (see Section 4.3), this would involve determining the affine constant (ie the modular form) relating theta constants of level ℓn and the products of theta constants of level n: $\prod_{i=1}^r \theta_i(0)$ (the number depending on the version of the generalised Segre morphism we use). Similarly, from the point of view of Section 4.4, this would involve determining the affine (ie modular) version of the morphism $B \to C$. With Koizumi's descent formula, this is easy because $B \to C = B^r$ is given by the Segre embedding, but in the faster version $C = B^r/B[\ell]$, so the theta constants are mixed.

An alternative strategy would be to have an algebraic way to evaluate a given modular form. By algebraic I mean given a modular form $\mathfrak{g}$, a representation of A and a basis of differential w_A, to evaluate $\mathfrak{g}(A, w_A)$. Then given the explicit isogeny $f : A \to B$, we can compute the normalised differential w_B (ie such that $f^* w_B = w_A$), and evaluate $\mathfrak{g}(B, w_B)$, this gives the modular version of the isogeny. From this point of view, Theorem 4.6.3 can be seen as a way to evaluate the theta constants (seen as modular forms) algebraically along isogenies, we will use that in Section 5.6.5. As we will see in Section 5.4, because of the Kodaira-Spencer isomorphism, a particularly important case for algebraic evaluation is when $\mathfrak{g} = dJ$, the differential of a modular invariant.

4.7 ISOGENIES FROM DIFFERENTIAL EQUATIONS

We now explain how, given the tangent map, we can compute the isogeny $f : A \to B$ by solving a differential equation. The tangent map M itself may be computed by computing the image of a fat point $P_\epsilon \in A(k[\epsilon])$. We will see another method using modular polynomials in Section 5.4.

4.7.1 *Elliptic curves*

Using differential equations to compute isogenies between elliptic curves is due to Elkies. Over a finite field $\mathbb{F}_q$, using [BMS+08], once we have M the differential equation allows to recover the ℓ-isogeny $f : E_1 \to E_2$ (equivalently its kernel) in quasi-linear time $\widetilde{O}(\ell \log q)$, provided that $p > 8\ell - 5$ (with slightly more information they only need $p > 2\ell - 1$).

Indeed, writing $E_1 : y^2 = x^3 + ax + b$, $E_2 : Y^2 = X^3 + AX + B$ (assuming $p > 3$ for simplicity), then since $f^*dX/Y = \frac{1}{M}dx/y$ by assumption, we know that $f(x, y) = (u(x), Myu'(x))$ where $u = g/h$ is a rational function whose denominator is $h(x) = \prod_{P \in \mathrm{Ker} f 0_E}(x - x(P)) = x^{\ell-1} + \sigma x^{\ell-2} + \dots$. Given the kernel K, hence the denominator $h(x)$, u is explicitly given by $u(x) = \ell x - \sigma - (3x^2 + A)h'(x)/h(x) - 2(x^3 + Ax + B)(h'(x)/h(x))'$ [Koh96]. (If ℓ is odd we can work with the square root of h). Furthermore plugging the equation of E_2 shows that u satisfy the differential equation

$$M^2(x^3 + ax + b)u'(x)^2 = u(x)^3 + Au(x) + B. \tag{4.6}$$

To solve this singular differential equation, it is convenient to express f along the uniformisers $z = x/y$ and $Z = X/Y$ at 0_{E_1} and 0_{E_2}, ie writing $Z = v(z)$. Via this change of variable, this gives a non singular differential equation satisfied by $v(z) \in k[[z]]$, which can be solved by Newton iterations to some precision, from which a rational reconstruction gives $v(z) \in k(z)$ in quasi-linear time and we then recover $u(x) \in k(x)$ in time $O(M(\ell)\log \ell)$ operations in $\mathbb{F}_q$ where $M(\cdot)$ is the complexity of the multiplication, ie in $\widetilde{O}(\ell \log q)$ binary operations. In [BMS+08, § 4.3] the authors use the change of variable $u = \frac{1}{S(\frac{1}{\sqrt{x}})^2}$ instead, so that $1/\sqrt{X} = S(1/\sqrt{x})$. This is essentially the same as the change of variable above since $1/x = z^2 + O(z^6)$. The Newton step requires $p > 8\ell - 5$, but if $\sigma = \sum_{P \in \mathrm{Ker} f 0_E} x(P)$ is also known, Elkies give an algorithm to recover $u(x)$ directly without rational reconstruction from $v(z) \in k[[z]]$, for a total complexity of $O(M(\ell))$ operations in $\mathbb{F}_q$ and we only need $p > 2\ell - 1$ (see [BMS+08, Theorem 2]).

In the setting of [BMS+08], M is recovered from the derivatives of the modular polynomial Φ_ℓ evaluated at j_{E_1} and j_{E_2}, and σ can be recovered for the second derivatives (see the formulae in [Sch95, § 7]). When given the kernel, Vélu's formula [Vél71] gives the normalised isogeny $f : E_1 \to E_2$. Finally, if the isogeny is given via a "black box" allowing to evaluate $f(P)$ for points P in E_1, we can compute equations for E_2 by evaluating on a few points, then compute M by evaluating at a fat point. For instance, since $x = z^{-2} + O(z^2)$ and $y = z^{-3} + O(z)$, $dx/y = (-2 + O(z))dz$, then we take the fat point induced by $z \mod z^2$ above 0_E, to recover $Z \circ f = Mz$ modulo z^2. If needed, σ could also be recovered by working modulo z^4 and looking at the coefficient of z^3.

4.7.2 *Hyperelliptic curves of genus* 2

For hyperelliptic curves of genus 2, as far as I am aware the idea to use differential equations too to compute isogenies (even when given its kernel, rather than in a modular context as in Elkie's method) is from [CE14]. Differential equations are also used in [CMS+17] to compute endomorphism rings of Jacobians.

The map M determines f if the characteristic is large enough, because there is at most one ℓ-isogeny $A \to B$ with tangent map M and $\ell \leq N$, where $4N < \mathrm{char}\, k$ (or $N = \infty$ if $\mathrm{char}\, k = 0$), see [KPR20, Lemma 5.1]. We will assume f separable (this is the case under the hypothesis above), so that M is inversible.

If A is a Jacobian, $A = \mathrm{Jac}(C)$, taking a base point P of C, it suffices to describe the map $C \to A \to B$. This allows to solve the differential equation on the completion of a point of C rather than of A, ie work with a power series ring of dimension 1 rather than g.

We give a bit more details when A, B are the Jacobians of genus 2 hyperelliptic curves C, C', and refer to [KPR20, § 5] for a more treatment. We look at the compositium

$$C \xhookrightarrow{Q \mapsto [Q-P]} \mathrm{Jac}(C) \xrightarrow{f} \mathrm{Jac}(C') \dashrightarrow C'^{2,\mathrm{sym}} \xrightarrow{m} \mathbb{A}^4$$

where P is any rational point on C, and m is the rational map given by

$$\{(x_1, y_1), (x_2, y_2)\} \mapsto \Big(x_1 + x_2,\ x_1x_2,\ y_1y_2,\ \frac{y_2 - y_1}{x_2 - x_1}\Big).$$

This compositum is a tuple of four rational fractions $s, p, q, r \in k(u, v)$ that we call the *rational representation of f at the base point P*.

Taking if needed a degree 2 étale extension to desymmetrize, we solve the power series x_1, x_2, y_1, y_2 of f_P, that satisfy the differential system given by

$$\begin{cases} \dfrac{x_1\,dx_1}{y_1} + \dfrac{x_2\,dx_2}{y_2} &= (m_{1,1}x + m_{1,2})\dfrac{dx}{y} \\ \dfrac{dx_1}{y_1} + \dfrac{dx_2}{y_2} &= (m_{2,1}x + m_{2,2})\dfrac{dx}{y} \\ y_1^2 = E_{C'}(x_1) \\ y_2^2 = E_{C'}(x_2), \end{cases} \tag{S}$$

where we consider the coordinates x, y on C as elements of $k[[z]]$, z a uniformizer of C around P.

The initialisation is given by the image of P in ${C'}^{\mathrm{sym},2}$ which is of the form $\{Q, \iota(Q)\}$, where ι is the hyperelliptic involution. Now if we computed the tangent matrix via the image of P_ϵ, we already know Q. If we computed it by using modular polynomials as in Section 5.4, then we need to use the geometry of the curves to recover Q, see [KPR20, Proposition 5.4].

A small difficulty is that the differential system is singular of valuation 1 [KPR20, Lemma 5.6]. so we need to use the geometry of the curves again [KPR20, § 5.2] to find the first few terms in the series before switching to Newton iterations [KPR20, Proposition 5.8].

Once we have enough precision, we do a rational reconstruction. The degrees of s, p, q, r as morphisms from C to $\mathbb{P}^1$ can be computed as intersection degrees of the divisor $f_P(C)$ on $\mathrm{Jac}(C')$, and the polar divisors of s, p, q and r as functions on $\mathrm{Jac}(C')$ (or alternatively as intersection degrees of C and the pullback of the polar divisor). For an ℓ-isogeny we get bounds of the form $O(\ell)$, and for a β-isogeny, we get bounds of the form $O(\mathrm{Tr}_{K/\mathbb{Q}}(\beta))$, see [KPR20, § 5.3].

In summary:

Theorem 4.7.1 ([KPR20]). *Let $f : A \to B$ be an ℓ-isogeny between $A = \mathrm{Jac}(C), B = \mathrm{Jac}(C')$, two Jacobians of hyperelliptic curves of genus 2 such that we have the tangent matrix of f (eg on the canonical differential basis $(dx/y, xdx/y)$ induced by the equations of the hyperelliptic curves). Assume that $p > 8\ell + 7$ (or $p = 0$). Then we can compute the representation $f : C \to \mathrm{Jac}(C) \to \mathrm{Jac}(C')$ which is given by rational functions of total degree $O(\ell)$ in time $\widetilde{O}(\ell)$ operations in an extension k'/k of degree at most 4 and $O(1)$ square roots in k'.*

The same holds in the Hilbert case for a β-isogeny. In this case, if $p = 0$ or $p > 4\,\mathrm{Tr}(\beta) + 7$, the representation $f : C \to \mathrm{Jac}(C) \to \mathrm{Jac}(C')$ is given by rational functions of total degree $O(\mathrm{Tr}(\beta))$ and can be computed in quasi-linear time in k'/k along with $O(1)$ square roots in k'.

Of course, a similar algorithm works when A or B is a product of elliptic curves.

Proof. The Newton algorithm to solve the differential system is quasi-linear, as is the rational reconstruction.

A first quadratic extension is needed to compute a generic base point P (in the sense of [KPR20, Lemma 5.2]) to define the mapping $C \to \mathrm{Jac}(C)$. If we start the algorithm with a curve with a base point P already, then P will generically be generic, so no extensions are needed (generically). The last quadratic extension is because for simplicity we desymmetrize the system to work on the completion of ${C'}^2$ rather than ${C'}^{\mathrm{sym},2}$, in other words to express $f((P') - (P))$ as $(P_1) + (P_2) - (Q) - (i(Q))$ where $\{Q, i(Q)\}$ is the image of P in ${C'}^{\mathrm{sym},2}$ rather than as a divisor written in Mumford coordinates. □

Remark 4.7.2. The assumption on the characteristic to be able to solve the differential system is essentially harmless in practice: as explained in [KPR20, § 1], if p is too small compared to ℓ, we can use the idea of [LS08] to lift the isogeny to $\mathbb{Z}_q/p^m\mathbb{Z}_q$ with m large enough and solve the differential system there. Once the lifting is done, this still gives a quasi-linear algorithm if solving the differential system does not loose to much p-adic precision. When $g = 1$ this was proved in [LV16; CEL20], and when $g = 2$ this was recently proved in [Eid20]. For the complexity of the lifting itself, see Section 5.4.3.

Remark 4.7.3. More generally, when we have an ℓ-isogeny $f : (\mathrm{Jac}(C), \ell\Theta) \to (A, \mathcal{L})$, where Θ is the Theta divisor on $\mathrm{Jac}(C)$. We can look at the composition $C \xrightarrow{i} \mathrm{Jac}(C) \xrightarrow{f} A$, assuming that C has a rational point for simplicity. Then we may compute this map by solving a suitable differential equation.

Then by the push-pull formula, if D is a divisor on $\mathrm{Jac}(C)$, $i_*(i^*[D]) = [i(C)] \cdot D$. By Poincaré's adjunction formula, $\deg([i(C)] \cdot [\Theta]) = g$. This is also a consequence of Riemann's inversion of the Jacobi map: if E is an effective divisor of degree g on C, $(C + (K_C - E)) \cap \Theta$ is linearly equivalent to E (after pulling back to C).

Thus if s is a section of $\mathcal{L}^m$ on A, if D_s is the polar divisor of s, then since $f^*D_s \simeq m\ell\Theta$, i^*f^*s has for polar divisor a divisor of degree $m\ell g$. This allows to bound the degree of pullback of sections of A restricted to C.

In particular, we can apply this to study endomorphisms $\alpha : \mathrm{Jac}(C) \to \mathrm{Jac}(C)$. For instance, when $C : y^2 = h(x)$ is hyperelliptic, we can look at $C \xrightarrow{i} \mathrm{Jac}(C) \xrightarrow{[\ell]} \mathrm{Jac}(C)$ (here there is no need to compute the action on tangent space at 0 since we know that it is given by $[\ell]$). Cantor's polynomials [Can94] give the expression of this map in terms of the Mumford coordinates (u, v), this is the generalisation of the ℓ-division polynomial from elliptic curves. The coordinates of u and v are given by rational functions whose polar divisors are $O(1)$ multiple of Θ where Θ is the theta divisor, (we can determine these constants explicitly if needed).

Since $\ell^*\Theta \simeq \ell^2\Theta$, we deduce that the Cantor polynomials are of degree $O(1)g\ell^2$ on C, hence of degree $O(1)g\ell^2$ when expressed in terms of x, y. This answers a conjecture of [AGS19b, § 6].

This can be generalised to a version of Cantor polynomials for a real endomorphism α. If the real multiplication field is given by $K = \mathbb{Q}(\eta)$, and $\mathfrak{l}$ is a prime above a totally split prime ℓ, we can find $\alpha = \sum_{i=1}^{g} a_i\eta^i$ in $\mathfrak{l}$ with $a_i = O(\ell^{1/g})$ (constants depending on K) [Abe20, Lemma 5]. So using the bound above on the a_i-multiplication, we get that the α-Cantor polynomial have coefficients of degree $O(\ell^{2/g})$. This also solves a conjecture in [Abe20, § 2] which allows to improve the exponent bound for point counting on RM hyperelliptic curves, from an exponent $c = 9$ to $c = 7$.

If we have a totally positive real endomorphism β, I conjecture that $\deg([\beta\Theta] \cdot [i(C)]) = \mathrm{Tr}(\beta)$. This is true when $g = 2$, using a formula of Kani [Kan19a, Remark 16], see [KPR20, Proposition 5.13]. Applying the above conjecture to $\beta = \alpha^2$, this gives us the bound $O(\ell^{2/g})$ directly. If α is totally positive and $f : \mathrm{Jac}\, C \to B$ is an α-isogeny, then this conjecture would also prove that the restriction $f \mid C$ is of degree $O(\ell^{1/g})$.

This approach works to bound the degree of division polynomial on a curve C expressed in terms of coordinates on $\mathrm{Jac}(C)$ whose polar divisors are bounded multiples of Θ. This also suggests the alternative strategy to compute Cantor like division polynomials by solving a differential equation rather than via a recursive formula. This paves the way to extend [AGS19b; Abe20] to non hyperelliptic curves.

Remark 4.7.4. When computing an isogeny of the form $f : A = \mathrm{Jac}(C) \to B$, when we have a rational point P on C, we have seen that it is convenient to use the representation of the isogeny as $f : C \to B$. Conversely, to construct back the map $\mathrm{Jac}(C) \to B$, it suffices to use elimination theory to build $\sum f(P_i)$ where $D = \sum((P_i) - (P))$. Of course when $g = 2, C : y^2 = f(x)$ with $\deg f = 5$ and we use $P = \infty$ the point at infinity, we can use Mumford's representation $D = (u, v)$, where if $D = (P_1) + (P_2) - 2(P)$, $u(x) = (X - x(P_1))(X - x(P_2)) = X^2 - ax + b$ and $v(x(P_i)) = y(P_i)$. To recover $f(D)$, it suffices to compute $f(X, v(X)) + f(a - X, v(a - X))$ modulo $u(X)$. We can also recover equations of the kernel by asking that $f(X, v(X)) = -f(a - X, v(a - X))$. See Section 5.5 for more details on this step.

4.7.3 *Compressing isogenies*

A separable isogeny $f : A \to B$ over a finite field $\mathbb{F}_q$ is completely determined by its kernel $K = \mathrm{Ker}\, f$. Representing this kernel requires $O(\deg f \log q)$ bits. But we can compress this by using the representation of f via the differential equations as above. For simplicity we treat the case of elliptic curves, the case of abelian surfaces (or abelian varieties) being an immediate generalisation.

So we have a kernel K of degree ℓ on an elliptic curve E over $\mathbb{F}_q$, which we can assume cyclic. We can represent it via the polynomial of degree $\ell - 1$ over $\mathbb{F}_q$: $h(x) = \prod_{T \in K0_E}(x - x(T))$ (or its square root if ℓ is odd), for a total size of $O(\ell \log q)$ bits. If $K = \langle T \rangle$ and $T \in \mathbb{F}_q$, then T is of course enough to characterize K, and only require $O(\log q)$ bits. But in general T will live in an extension of degree ℓ, so specifying T requires $O(\ell \log q)$ bits, and we do not gain compared to directly giving the equation of K.

We now explain, if $q = p^d$, how there is always a representation that use only $O(d \log \ell + \log q)$ bits. We remark that if T is rational, then this already imply $\log \ell = O(\log q)$ using the Hasse-Weil bounds since T is of ℓ-torsion. Hence there is always a representation of the kernel as compact as when T is rational (whenever d is fixed).

One way to achieve this representation is to encode f by a point $P \in E$ and the value $f(P)$. This is sufficient to represent f provided that P is a point of order $N \geq \ell$ (the exact value of N is not needed), so this imply to work in an extension $\mathbb{F}_{q^m}$ of degree $m = O(\log_q \ell)$, so $(P, f(P))$ take size $O(\log \ell + \log q)$. There are several drawback to this representation however: first we need to find a point P of sufficiently large order, and secondly reconstructing f amount to a rational fraction interpolation between the points $[i]P$ and $[i]f(P)$ for $i = 1, \ldots, \ell$. For elliptic curves this is not a problem because f is essentially given by a rational function $u(x) = g(x)/h(x)$ in x only, so can be computed in quasi-linear time, but for higher dimension since we don't control the interpolation points, hence we cannot use fast interpolation.

A solution to both of these problems is to take for P a fat point (say above 0_E), ie $P \in E(\mathbb{F}_q[\epsilon])$. If P is not trivial it will behave like a point of order p. Furthermore, by the computations of Section 4.7.1, to give such a P and its value $f(P)$ is the same as giving equations of E and $E' = E/K$ and the action of f on the canonical differentials dx/y, or alternatively to give equations of E' such that f is normalised.

Then recovering f, hence K amount to solving the differential equation from Equation (4.6), which we have seen can be done in quasi-linear time. In other words the decompression is quasi-linear. Conversely, given K, Vélu's formula give the normalised E', so the compression is linear. This strategy works as long as p is sufficiently large compared to ℓ. In small characteristic, we need to lift to $\mathbb{Z}_q/p^m\mathbb{Z}_q$ with $m = O(\log_p \ell)$, and give the normalised equations of E and E' there. Thus in general the compressed representation takes $O(m \log q) = O(d \log \ell + \log q)$ bits. The decompression is still quasi linear, but for the compression we need to lift E arbitrarily to p-adic precision m, and then lift K and then compute the isogeny. Once K is lifted, computing the isogeny using Vélu's formula take time $O(\ell m \log q) = O(\ell(d \log \ell + \log q))$, hence is quasi-linear if d is fixed.

We will see methods for lifting K in Chapter 6. We can lift K by lifting $h(x)$ as a factor of the ℓ-division polynomial ψ_ℓ, since it is prime to its cofactor, by lifting a Bézout relation. Since ψ_ℓ is of degree $O(\ell^2)$, this costs $\widetilde{O}(\ell^2 m \log q)$ using fast arithmetic. Alternatively, we can lift a generator T of K, this involves working over an extension of degree $O(\ell)$, hence costs $\widetilde{O}(\ell m \log q)$. Indeed the multiplication by $[\ell]$ can be computed in time $O(\log \ell)$ operations in the base ring, so lifting T is quasi-linear by Remark 6.2.3. Furthermore, since all the non trivial geometric points of K live in the same extension of degree $e \mid \ell - 1$, we can use an equal degree factorisation algorithm (after $\log \ell$ gcds and modular compositions) to find T in time $\widetilde{O}(\ell \log q)$ operations in $\mathbb{F}_q$ by [KU11].

In summary:

Proposition 4.7.5. *An ℓ-isogeny $f : E \to E'$ can be compressed into $O(1)$ elements in $\mathbb{Z}_q/p^m\mathbb{Z}_q$ with $m = O(\log_p \ell)$, ie in $O(m \log q) = O(d \log \ell + \log q)$ bits, and the compression and decompression are quasi-linear, ie take time $\widetilde{O}(\ell m \log q) = \widetilde{O}(\ell(d \log \ell + \log q))$.*

We will also see in Section 5.4.1 and Remark 5.4.4 that to give normalised equations for E and E' with respect to the isogeny f is also equivalent to give the values $j(E), dj(E, w_E), j(E'), dj(E', w_{E'})$ such that $f^* w_{E'} = w_E$. This is interesting because these values are encoded by the ℓ-modular polynomial ϕ_ℓ, see Section 5.3. In particular, given $j(E)$, and $j(E')$, one can compute $dj(E', w_{E'})$ from $dj(E, w_E)$ and the derivatives $\partial/\partial X \phi_\ell(j(E), j(E'))$ and $\partial/\partial Y \phi_\ell(j(E), j(E'))$. (Lifting $j(E)$ to $\mathbb{Z}_q/p^m\mathbb{Z}_q$ first if needed when $\mathbb{F}_q$ is of small characteristic.) This has two interesting consequences: the first is that we may recover f only from the derivatives of ϕ_ℓ at $j(E), j(E')$, without needing the kernel K (we will go back to this in Section 5.4), and the second that the evaluated polynomials $\phi_\ell(j(E), Y)$ and $\partial/\partial X \phi_\ell(j(E), Y)$ which are of size $O(\ell m \log q)$ give a compact representation of all ℓ-isogenies with domain E.

We will see in Section 5.3.8 that if $q = p^d$, we can evaluate these polynomials in time $\widetilde{O}(\ell^2(m \log q + d^2))$. This gives our third method to lift to precision m (with the advantage of not needing the kernel). We will also use this in Section 5.4.1 to study the problem of recovering the isogeny given only E and E'.

4.8 CONCLUSION AND PERSPECTIVES

We have described a general framework to compute isogenies in quasi-linear time. This framework encompasses the isogenies computations done in the theta model [LR12; CR15; LR15b] and in Jacobians [CE14; Mil20]. There is still exciting work to do, notably implementation wise for cyclic isogenies (speed up the implementation in the theta models and adapt the methods to the Jacobian models). It should not be too hard to work out isogenies in the projective rational model for abelian surfaces of Cassels and Flynn [Fly90; CF+96], by working out Algorithmic Hypothesis 2.9.2 there. Having fast isogenies is an important tool that we will use in Chapters 5 and 7.

But the most exciting challenge is whether we could dream of an ℓ-isogeny algorithm in time $\widetilde{O}(\log \ell)$ (even polynomial in $\log \ell$ would be a breakthrough). Of course for this to possibly work we have to assume that the kernel K has rational points, and we are given a basis $P_1, \dots, P_g$ of K/k rather than equations of the kernel. Indeed, since the kernel is of degree ℓ^g, just writing equations would be too costly.

When $\ell = \ell_0^n$ is the power of a small prime, there is such an algorithm, by decomposing the ℓ-isogeny as a composition of n ℓ_0-isogenies. To get a quasi-linear algorithm is not trivial however, and is the main work of [JD11; DJP14]. Indeed, let P be a generator of $K \subset E[\ell]$. We may compute $P_1 = \ell_0^{n-1} P$, and if f_1 is the corresponding ℓ_0-isogeny with kernel generated by P_1, compute f_1 and iterate. We could also compute all multiples $\ell_0^i P$ and push them through f_1, then f_2 and so on. The best strategy depends on whether computing the ℓ_0-multiple of a point or pushing a point via an isogeny is faster, but both are in $O(n^2)$ (as a dependency on n). An intermediate strategy

is as follow: assume $n = m^2$. Then we compute the $\ell_0^{mi}P$, and then P_1, and we push them through f_1 and so on. This gives a complexity of $O(n^{3/2})$. A clever recursion on this strategy then gives a quasi linear algorithm. This of course readily extends to abelian varieties.

Of course, to compute the multiplication by $[\ell]$, the double and add algorithm gives a complexity in $\widetilde{O}(\log \ell)$, which do not depends on the prime decomposition of ℓ. For ℓ-isogenies, the above strategy does not work if ℓ is prime. However there was a very recent breakthrough to compute ℓ-isogenies between elliptic curve using Vélu's formula in time $\widetilde{O}(\sqrt{\ell})$ [BDL+20].

This raises the exiting prospect that maybe there do exists an algorithm polynomial in $O(\log \ell)$. Maybe such an algorithm would also help in computing the Weil-Cartier pairing (see Section 3.6). Meanwhile, a more reasonable task is to extend the $\widetilde{O}(\sqrt{\ell})$ algorithm to abelian varieties, using the formula of this Chapter.

Part II

ALGORITHMS FOR MODULAR SPACES

5 MODULAR CORRESPONDANCES

CONTENTS

5.1 INTRODUCTION

The main goal of this Chapter is to compute modular (aka Hecke) correspondances on Siegel $\mathfrak{H}_g$ and Hilbert $\mathfrak{H}_1^g$ moduli spaces. More details on these moduli and modular correspondances are in [Rob21, Chapter 5].

The Siegel moduli space parametrizing principally polarised complex abelian varieties is given by $\mathcal{A}_{g,\mathbb{C}} = \mathfrak{H}_g/\operatorname{Sp}_{2g}(\mathbb{Z})$ [Rob21, Section 5.4]. If $\Gamma_0(\ell) \supset \Gamma(\ell)$ is the standard level subgroup, we have a modular correspondance $\mathcal{A}_{g,\Gamma_0(\ell),\mathbb{C}} := \mathfrak{H}_g/\Gamma_0(\ell) \to \mathcal{A}_{g,\mathbb{C}} \times \mathcal{A}_{g,\mathbb{C}}$ defined by $\tau \mapsto (\tau, \tau/\ell)$ and parametrizing ℓ-isogenies.

Modular invariants then give (Siegel) modular polynomials which describe this correspondance. If $g = 1$ the standard modular invariant is the j-invariant, if $g = 2$ several choice of Igusa j_1, j_2, j_3 invariants have been used in the literature. We will be using Streng's version [Str10, § 2.1] which gives smaller coefficients. These modular invariants can be defined in terms of the covariants I_2, I_4, I_6, I_{10} of hyperelliptic curves of genus 2, or in terms of

the modular forms $\psi_4, \psi_6, \chi_{10}, \chi_{12}$ or in term of theta constants. See [Igu60; GL12, § 2; Str10, § 2.1 and § 7.1], [KPR20, § 2.2 and § 3.3][§ 2 and § 3]DRcanonicalliftg2 for more informations and formulae; a summary is in [Rob21, Section 8.2]. Igusa invariants are only birational invariants on $\mathcal{A}_{g,\mathbb{C}}$ when $g = 2$, so the modular polynomials only give a birational version of the modular correspondance (equivalently: they describe the modular correspondance at the generic point of $\mathcal{A}_{g,\Gamma_0(\ell),\mathbb{C}}$; they describe the field extension $\mathbb{C}(\mathcal{A}_{g,\Gamma_0(\ell),\mathbb{C}}) = \mathbb{C}(j_i(\tau/\ell))$ over $\mathbb{C}(\mathcal{A}_{g,\mathbb{C}}) = \mathbb{C}(j_i(\tau)), i = 1, \dots, 3$). For instance, for abelian surfaces, Streng's invariants are defined when $\chi_{10} \neq 0$, which is exactly the locus of Jacobians of hyperelliptic curves of genus 2. By [Igu60] this locus is isomorphic to $\mathbb{C}[y_1, y_2, y_3, y_4]^{\mu_5} = \mathrm{Proj}[I_2, I_4, I_6, I_{10}][I_{10}^{-1}]$ (where the grading is such that I_{2i} is of weight $2i$). Streng's invariants generate the coordinate ring $\mathcal{A}_{g,\mathbb{C}}[\psi_4^{-1}, \chi_{10}^{-1}]$ hence are modular invariants on this affine open. Using [Igu60] we may extend this to characteristic p, except when $p = 2$ where Streng invariants have bad reduction, so we have to use invariants constructed from $J_2, J_4, J_6, J_8, J_{10}$ which have good reduction everywhere, see [MR21].

Complex abelian varieties with real multiplication by $O_{\mathcal{F}}$ ($\mathcal{F}$ a totally real number field of degree g) are parametrized by the Hilbert moduli space $\mathfrak{H}_1^g / \mathrm{Sl}_2(O_{\mathcal{F}} \oplus \partial_{\mathcal{F}}^{-1})$ [Rob21, Section 5.5]. If $\beta \gg 0$ is totally positive, the level subgroup $\Gamma_0(\beta)$ allows to define the moduli space $\mathfrak{H}_1^g/\Gamma_0(\beta)$ paremetrizing β-isogenies. Hilbert modular polynomials are then defined using Hilbert modular invariants. For abelian surfaces we either use Gundlach invariants [Gun63; Gun65] (in small discriminant), or pullback of Igusa invariants. We have also computed modular polynomials (both in the Hilbert and Siegel case) using level 2 theta constants [Mil15a], [MR20b].

These modular correspondances extend over $\mathbb{Z}$ and behave well (the moduli stacks are smooth and the maps of the correspondance are representable finite étale) over $\mathbb{Z}[1/\ell]$, ie for étale isogenies, see [Rob21, Sections 5.7 and 5.8], By contrast, fibers of the p-modular correspondance in caracteristic p are not even quasi-finite when $g > 1$, for instance there are fiber components of strictly positive dimension corresponding to kernels isomorphic to α_p^g above a supersingular abelian variety, see [CN90].

From the algebraic point of view, (scalar) modular forms can be seen as sections of the Hodge line bundle on $\mathcal{A}_g$ (which is ample, see [FC90, Chapter V]). If $\pi : X_g \to \mathcal{A}_g$ is the universal abelian stack with universal section ϵ, the Hodge vector bundle is $\mathcal{H} := \pi_* \Omega^1_{\mathcal{A}_g} \simeq \epsilon^* \Omega^1_{\mathcal{A}_g}$. (The isomoprhism comes from [MGE12, Prop. 3.15] and the fact that since an abelian variety as only constant global sections since it is projective, a global differential form is invariant.) If $\rho : \mathrm{Gl}_g \to V$ is a representation, a modular form of weight ρ is a section s of $\rho(\mathcal{H})$ (along with some boundary conditions if $g = 1$), ie a functorial application $(A, w_A) \mapsto \mathfrak{g}(A, w_A)$ where w_A is a basis of differential forms, such that if $\eta : A \to A'$ is an isomorphism, and γ is the matrix of η^* in the bases $w_{A'}, w_A$, then $\mathfrak{g}(A, w_A) = \rho(\gamma)\mathfrak{g}(A', w_{A'})$. Analytically, this corresponds to $\mathfrak{g}(\gamma \cdot \tau) = \rho(c\tau + d)\mathfrak{g}(\tau)$, where $\gamma = \begin{pmatrix} a & c \\ b & d \end{pmatrix}$. This allows to have an algebraic interpretation of the modular polynomials. We refer to [Rob21, Chapter 5] for more details.

A slight difficulty is that modular invariants give coordinates on the coarse moduli space, and on the level of coarse spaces the moduli are no longer smooth everywhere (although they are smooth at points with generic automorphisms). See [KPR20, § 4.2] for a detailed study of the modular correspondance at the level of the coarse spaces, compared to the modular correspondance defined at the level of the algebraic stacks.

In Section 5.2 we describe in more details the modular correspondance in the theta model, and classify its fibers. Modular polynomials, especially modular polynomials for abelian surfaces are described in Section 5.3, both for the Siegel and Hilbert (ie real multiplication) case. We give a general evaluation/interpolation algorithm, describe their size and then give algorithms to compute the evaluated modular polynomials directly. We then give applications of modular polynomials for isogenies computations in Section 5.4, point counting in Section 5.5, and exploring isogeny graphs in Section 5.6. We will see other applications for canonical lifts in Chapter 6 and class polynomials of CM abelian varieties in Chapter 7. Finally Section 5.7 gives some perspectives.

5.2 A GENERAL MODULAR CORRESPONDANCE IN THE THETA MODEL

We construct a modular correspondance $\mathcal{A}_{g,\ell n} \to \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$ in the theta model, which will be used in Chapter 6 to compute canonical lifts.

5.2.1 *Defining the modular correspondance*

In this section, we denote by $\mathcal{A}_{g,n}$ the moduli space over $\mathbb{Z}[1/n]$ of abelian varieties with a symmetric theta structure of level n (with n even), and $\mathcal{X}_{g,n}$ the universal abelian scheme above it, with a totally symmetric normalized relatively ample line bundle (see Remark 2.7.5).

If $n \geq 4$, Riemann's relations define projective schemes $\overline{\mathcal{X}}_{g,n} \to \overline{\mathcal{A}}_{g,n}$ by Theorem 2.7.4, and we denote $(\theta_i)_{i\in Z(\overline{n})}$ the theta coordinates on either $\mathcal{X}_{g,n}$ or $\overline{\mathcal{X}}_{g,n}$ and $(\theta_i(0))_{i\in Z(\overline{n})}$ the theta null point coordinates on either $\mathcal{A}_{g,n}$ or $\overline{\mathcal{A}}_{g,n}$ coming from the section $s : \overline{\mathcal{A}}_{g,n} \to \overline{\mathcal{X}}_{g,n}$ (which restricted to $\mathcal{A}_{g,n}$ corresponds to the zero section of $\mathcal{X}_{g,n}$).

Then the isogeny and change of level formula from Chapters 2 and 4 naturally extend to define a modular correspondance $\pi : \overline{\mathcal{A}}_{g,\ell n} \to \overline{\mathcal{A}}_{g,n} \times \overline{\mathcal{A}}_{g,n}$, induced by $\pi : \overline{\mathcal{X}}_{g,\ell n} \to \overline{\mathcal{X}}_{g,n} \times \overline{\mathcal{X}}_{g,n}$ where the first projection $\pi_1 : \mathcal{X}_{g,\ell n} \to \mathcal{X}_{g,n}$ is the universal change of level structure and the second projection $\pi_2 : \mathcal{X}_{g,\ell n} \to \mathcal{X}_{g,n}$ is the universal isogeny.

In fact if ℓ is prime and prime to n, we may see $\pi_1 : \mathcal{A}_{g,\ell n} \to \mathcal{A}_{g,n}$ as a $\Gamma/\Gamma(\ell) = \mathrm{Sp}_{2g}(\mathbb{Z}/\ell\mathbb{Z})$ (Galoisian) cover were $\Gamma = \mathrm{Sp}_{2g}\,\mathbb{Z}$, and the image $\pi(\mathcal{A}_{g,\ell n}) \subset \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$ then defines a $\Gamma/\Gamma_0(\ell)$ (non Galoisian) cover $\mathcal{A}_{g,n,\Gamma_0(\ell)}$ over $\mathcal{A}_{g,n}$. We will detail the construction of the map π and also explain how to define a $\Gamma/\Gamma_1(\ell)$ (non Galoisian) cover $\mathcal{A}_{g,n,\Gamma_1(\ell)}$ over $\mathcal{A}_{g,n}$. We restrict to ℓ prime to n, but the general case is not harder, see Section 2.10 and [Rob10, Chapitre 6].

On $\overline{\mathcal{X}}_{g,n}$, we have an explicit action λ of the Heisenberg group $\mathcal{H}(\overline{n})$ on $\Gamma(\mathcal{L}_{\overline{\mathcal{X}}_{g,n}})$ [Mum67a, Step 1, p. 84], whose restriction to $\mathcal{X}_{g,n}$ is the one induced by the universal theta structure. Writing $\mathcal{H}(\overline{n}) = \mathbb{G}_m \times Z(\overline{n}) \times \hat{Z}(\overline{n})$ where $\hat{Z}(\overline{n}) \simeq \oplus_{i=1}^{g}\mu_n$ is the Cartier dual of $Z(\overline{n})$, this canonical action is given by $\lambda(i).\theta_j = \theta_{i+j}$ for $i \in Z(\overline{n})$ and $\lambda(i).\theta_j = < i,j > \theta_j$ for $i \in \hat{Z}(\overline{n})$ where $< i,j >$ is the canonical pairing between $Z(\overline{n})$ and its Cartier dual $\hat{Z}(\overline{n})$. Acting on the zero section s gives a canonical basis of n-torsion (for $\mathcal{X}_{g,n}$) and Mumford's isogeny theorem [Mum66] Theorem 5.2.4 describes the universal isogeny (with a descent of level of the theta structure)

$$\pi_2 : \mathcal{X}_{g,\ell n} \to \mathcal{X}_{g,n}, (\theta_i)_{i\in Z(\overline{\ell n})} \mapsto (\theta_i)_{i\in Z(\overline{n})\subset Z(\overline{\ell n})}. \tag{5.1}$$

On $\mathcal{X}_{g,\ell n}$ the level ℓn theta structure induces a symplectic basis of the ℓn-torsion, and in particular a symplectic decomposition $K_1 \oplus K_2$ of the ℓ-torsion. Over a field k, $K_1 = \{(< i,j > \theta_j(0))_{j\in Z(\overline{\ell n})}\}_{i\in\hat{Z}(\bar{\ell})}$ is the kernel of π_2, while $K_2 = \{(\theta_{i+j}(0))_{j\in Z(\overline{\ell n})}\}_{i\in Z(\bar{\ell})}$ is such that $\pi_2(K_2) = \{(\theta_{i+j}(0))_{j\in Z(\overline{n})}\}_{i\in Z(\bar{\ell})}$ is the kernel of the ℓ-contragredient isogeny $\widetilde{\pi_2}$.

We can now describe the modular correspondance as follow.

A MAP INDUCED BY π_2. Denote $\Pi_2 : \mathcal{X}_{g,\ell n} \to \mathcal{X}_{g,n}^{\ell g}, (\theta_i)_{i\in Z(\overline{\ell n})} \mapsto \left(\pi_2(\lambda(i)(\theta_j))_{j\in Z(\overline{\ell n})}\right)_{i\in Z(l)}$, where λ is the action of the Heisenberg group $\mathcal{H}(\overline{\ell n})$ described above. For $j \in Z(\bar{\ell})$ the component Π_2^j of Π_2 is given by

$$\Pi_2^{j\,*}(\theta_i^{\mathcal{X}_{g,n}}) = \theta_{i+j}^{\mathcal{X}_{g,\ell n}}, \quad i \in Z(\overline{n}). \tag{5.2}$$

The image of the restriction of Π_2 to $\mathcal{A}_{g,\ell n}$ (seen as the zero section of $\mathcal{X}_{g,\ell n}$) then describes the moduli scheme $\mathcal{J}_{g,n,\ell}$ of abelian varieties with a level n symmetric theta structure together with the points of an isotropic kernel of the ℓ-torsion. This is the $\Gamma/\Gamma_1(\ell)$ cover $\mathcal{A}_{g,n,\Gamma_1(\ell)}$ of $\mathcal{A}_{g,n}$ we were looking for.

It is easy to see that π_2 extends to a morphism $\overline{\pi}_2 : \overline{\mathcal{X}}_{g,\ell n} \to \overline{\mathcal{X}}_{g,n}$ (or rather a rational map since contrary to π_2, $\overline{\pi}_2$ may not be defined everywhere). Since the action λ is defined on $\overline{\mathcal{X}}_{g,\ell n}$, we can also extend Π_2 to a (rational) morphism $\overline{\Pi}_2 : \overline{\mathcal{X}}_{g,\ell n} \to \overline{\mathcal{X}}_{g,n}^{\ell g}$. Let $\overline{\mathcal{J}}_{g,n,\ell}$ be the image of $\overline{\mathcal{A}}_{g,\ell n}$. By construction $\mathcal{J}_{g,n,\ell}$ embeds into $\overline{\mathcal{J}}_{g,n,\ell}$ and since we have explicit equations for $\overline{\mathcal{A}}_{g,\ell n}$, it is easy to derive equations for $\overline{\mathcal{J}}_{g,n,\ell}$.

DEFINING π_1. Likewise, we may define a descent of level formula. We introduced several version of these in Section 2.10.3, and they all easily extend to the universal abelian scheme. Since we work projectively here, they are all equivalent, and we choose to detail the Koizumi-Kempf descent of level formula (because we have a modular interpretation of it).

So let $r = 1$ if ℓ is a square, $r = 2$ if ℓ is a sum of two squares and $r = 4$ otherwise (the reason of our choice of r will appear in Step 3). On $\mathcal{A}_{g,\ell n}$ the Segre embedding yields a map $S : \mathcal{A}_{g,\ell n} \to \mathcal{A}_{rg,\ell n}$, which sends the universal abelian variety $\mathcal{X}_{g,\ell n}$ to $\mathcal{X}_{g,\ell n}^r$ with its product theta structure [Mum66, Lemma 1, p. 323]. Concretely,

$$S^*(\theta_{i_1,\dots,i_r}^{\mathcal{X}_{rg,\ell n}}) = \theta_{i_1}^{\mathcal{X}_{g,\ell n}} \cdot \dots \cdot \theta_{i_r}^{\mathcal{X}_{g,\ell n}} \tag{5.3}$$

In particular, for a k-rational point of $\mathcal{A}_{g,\ell n}$ corresponding to $(A, \mathcal{L}^\ell)$ (where $\mathcal{L}$ is of level n), S sends the theta null point of level ℓn of $(A, \mathcal{L}^\ell)$ to the theta null point of $(A^r, \mathcal{L}^\ell \star \dots \star \mathcal{L}^\ell)$

Let F be an $r \times r$ matrix with integral coefficients such that ${}^tFF = \ell\,\mathrm{Id}$, the Koizumi-Kempf formula [Koi76; Kem89a] yields a map $\mathcal{A}_{rg,\ell n} \to \mathcal{A}_{rg,n}$ which corresponds to the isogeny $F : \mathcal{X}^r_{g,\ell n} \to \mathcal{X}^r_{g,\ell n}$ along with the descent of product theta structure from level ℓn to level n. By Section 4.4.3, the formula is given, for $(i_1, \dots, i_r) \in Z(\overline{n})^r$, by

$$F^*(\theta^{\mathcal{X}_{rg,n}}_{i_1,\dots,i_g}) = F^*(\theta^{\mathcal{X}_{g,n}}_{i_1} \cdot \dots \cdot \theta^{\mathcal{X}_{g,n}}_{i_r}) = \sum_{\substack{(j_1,\dots,j_r)\in Z(\overline{\ell n})^r \\ F(j_1,\dots,j_r)=(i_1,\dots,i_r)}} \theta^{\mathcal{X}_{g,\ell n}}_{j_1} \cdot \dots \cdot \theta^{\mathcal{X}_{g,\ell n}}_{j_r}. \tag{5.4}$$

Since Equation (5.4) is homogeneous, this is well defined for projective coordinates.

In particular, seeing F as an isogeny $\mathcal{F}$ sends $(A^r, \mathcal{L}^\ell \star \dots \star \mathcal{L}^\ell)$ to $(A^r, \mathcal{L} \star \dots \star \mathcal{L})$, from which $(A, \mathcal{L})$ can be recovered by projecting to one of the factor. Then π_1 is the composition of this projection with $\mathcal{F} \circ {}^tF \circ S$.

Then we define $\pi = \pi_1 \times \pi_2$ and $\Pi = \pi_1 \times \Pi_2$.

Remark 5.2.1. We may extend π, and Π to $\overline{\pi} : \overline{\mathcal{X}}_{g,\ell n} \to \overline{\mathcal{X}}_{g,n} \times \overline{\mathcal{X}}_{g,n}$ and $\overline{\Pi} : \overline{\mathcal{X}}_{g,\ell n} \to \overline{\mathcal{X}}_{g,n} \times \overline{\mathcal{X}}^{\ell g}_{g,n}$. They also have affine versions, where to a trivialisation of $\mathcal{L}^\ell$ on A we associate corresponding trivialisations of $\mathcal{M}$ on B and of $\mathcal{L}^{\star\ell}$ on A^r.

5.2.2 *Fibers of the modular correspondance*

We may exploit the modular correspondance to find isogenous abelian varieties as follow. Starting from a k-rational point $(A, \mathcal{L}, \Theta_A)$ (we will often drop the theta structure for simplicity) of $\mathcal{A}_{g,n}$, we may either look at the geometric fiber $\pi_1^{-1}(A, \mathcal{L}) \subset \mathcal{A}_{g,\ell n}$ and project it into $\mathcal{A}_{g,n}$ via π_2 or instead look at $\pi_1 \circ \pi_2^{-1}$. This exactly correspond to the two isogeny strategies considered in Chapter 4.

Let us illustrate the second method here, the first would proceed in the same manner. This is simply a restatement of the method of Section 2.10.1 from a more "modular" point of view.

Let us then start with $(B, \mathcal{M})$ a k-rational point of $\mathcal{A}_{g,n}$. We first go through the intermediate covers $\mathcal{A}_{g,\ell n} \to \mathcal{A}_{g,n,\Gamma_1(\ell)} \to \mathcal{A}_{g,n,\Gamma_0(\ell)} \to \mathcal{A}_{g,n}$, a preimage of $(B, \mathcal{M})$ in $\mathcal{A}_{g,n,\Gamma_1(\ell)}$ thus correspond to a maximal totally isotropic kernel K of the ℓ-torsion, along with an explicit indexing of the geometric points of K by $(\mathbb{Z}/\ell\mathbb{Z})^g$. The point of going through $\mathcal{J}_{g,n,\ell} = \mathcal{A}_{g,n,\Gamma_1(\ell)}$ is that we have explicit equations of $\overline{\mathcal{J}}_{g,n,\ell}$, induced by the Riemann relations on $\overline{\mathcal{A}}_{g,\ell n}$ (which define this projective scheme by Theorem 2.7.4). Indeed, up to reindexing, Π_2 simply maps the universal (projective) theta coordinates $(\theta_i)_{i\in Z(\overline{\ell n})}$ to tuples of projective coordinates $\left((\theta_{i+j})_{j\in Z(\overline{n})}\right)_{i\in Z(\bar{\ell})}$.

By construction, given such a preimage (B, K) in $\mathcal{J}_{g,n,\ell}$, geometric points of $\Pi_2^{-1}(B, K) \subset \mathcal{A}_{g,\ell n}$ corresponds to abelian varieties $A_{\overline{k}} \in \mathcal{A}_{g,\ell n}(\overline{k})$ with a level ℓn symmetric theta structure such that the universal isogeny π_2 restricted to A is the contragredient isogeny of $B_{\overline{k}} \to A_{\overline{k}} = B_{\overline{k}}/K_{\overline{k}}$. More precisely, if we start $(B, \mathcal{M}, K)/k \in \mathcal{A}_{g,n,\Gamma_0(\ell)}$, if k' is an étale extension of k such that all points of K are defined, then fixing an isomorphism $Z(\bar{\ell}) \to K$ over k' yields a k'-point of $\mathcal{J}_{g,n,\ell}$. A k"-point in $\Pi_2^{-1}(B, K)$ then correspond to a theta structure on $(A, \mathcal{L}^\ell)$ defined over k" such that the contragredient isogeny f of $\tilde{f} : B \to A$ is given by the pullback of π_2 to A. The discussions in Section 2.10.1 can then be reinterpreted as a way to use Riemann relations to give explicit equations for $\overline{\Pi}_2^{-1}(B, K)$ and $\Pi_2^{-1}(B, K)$.

Let us recall this briefly: the abelian variety $(B, \mathcal{M})/k$ is described by its theta null point $(\theta^B_i(0))_{i\in Z(\overline{n})}$, and the points of K by their projective theta coordinates $\{(\theta^B_i(P_j))_{i\in Z(\overline{n})}\}_{P_j\in Z(\bar{\ell})}\}$. Take arbitrary affine lifts of the theta coordinates of P_j. Introduce indeterminates $(\lambda_i)_{i\in Z(\bar{\ell})}$, and construct the affine point $Q = (\lambda_i\theta_j(P_i))_{i\in Z(\bar{\ell}),j\in Z(\overline{n})}$. Via the Chinese remainder theorem, Q can be seen as a point on $\mathbb{P}^{Z(\overline{\ell n})}_k$. Plugging the equations of $\overline{\mathcal{A}}_{g,\ell n}$ on Q then describes the zero dimensional scheme $\overline{\Pi}_2^{-1}(B, K)$ in term of the λ_i. In practice, as explained in Section 2.10.1, only a subset of the Riemann equations defining $\overline{\mathcal{A}}_{g,\ell n}$ are enough to describe $\overline{\Pi}_2^{-1}(B, K)$ completely: the ones needed for differential additions and three way additions.

Indeed, from differential additions, we get relations of the type $\gamma''_{ij}\lambda_{i-j}\lambda_{i+j} = \gamma'_{ij}\lambda_i^2\lambda_j^2$ for some constants $\gamma''_{ij}, \gamma'_{ij}$. Since over k there is always a Riemann relation yielding a non-zero coefficient, we can assume that γ''_{ij} is inversible and get an equation

$$\lambda_{i-j}\lambda_{i+j} = \gamma_{ij}\lambda_i^2\lambda_j^2. \tag{5.5}$$

Likewise the Riemann equations for three way additions yields relations of the form

$$\lambda_{i+j+k}\lambda_i\lambda_j\lambda_k = \gamma_{ijk}\lambda_{i+j}\lambda_{i+k}\lambda_{j+k}. \tag{5.6}$$

Thus the λ_i are completely determined from the λ_{e_i} and $\lambda_{e_i+e_j}$, where $(e_1,\dots,e_g)$ is a basis of $Z(\bar{\ell})$ for $i \neq j \in \{1,\dots,g\}$, if they are non zero. Indeed from these indeterminates, one can use Equation (5.6) repeatedly to compute all $\lambda_{e_{i_1}+e_{i_2}+\cdots+e_{i_m}}$ and then use Equation (5.5) to compute all $\lambda_{n_1e_1+n_2e_2+\cdots+n_ge_g}$ where $n_1,\dots,n_g \in \{0,\dots,n-1\}$. Finally looking at the Riemann equations yielding the opposite of a point, we get the symmetry relations

$$\lambda_i = \gamma_i \lambda_{-i}. \tag{5.7}$$

Writing $\ell = 2\ell' + 1$ (ℓ is odd), using Equations (5.5) and (5.7) we get equations of the type $\lambda_i^\ell = C_i$. But we have just seen that all λ_i can be rewritten in terms of the $\lambda_{e_i}, \lambda_{e_i+e_j}$ so in final it remains the equations:

$$\lambda_{e_i}^\ell = C_{e_i} \quad \lambda_{e_i+e_j}^\ell = C_{e_i,e_j} \tag{5.8}$$

for $i \neq j \in \{1,\dots,g\}$. We refer to [LR12; CR15] for more details.

Proposition 5.2.2. *Let $e_1,\dots,e_g$ be a basis of $Z(\bar{\ell})$. The zero dimensional scheme $\Pi_2^{-1}(B,K)$ is the open subscheme of $\overline{\Pi}_2^{-1}(B,K)$ given by $\lambda_i \neq 0$. It is isomorphic via Equations (5.5) and (5.6) to the scheme defined by Equation (5.8) in $k[\lambda_{e_i}, \lambda_{e_i+e_j}]$ where $i \neq j \in \{1,\dots,g\}$.*

Proof. By the discussion above, the scheme $\overline{S}$ defined by Equations (5.5) to (5.7) contains the scheme $\overline{\Pi}_2^{-1}(B,K)$. If $(A,\mathcal{L})$ is an abelian variety with a symmetric level ℓn theta structure coming from a geometric point of $\Pi_2^{-1}(B,K)$, then $\lambda_i \neq 0$, otherwise the image of the ℓ-torsion by π_2 would not be well defined. Furthermore, looking at the action of the subgroup $\mathcal{G}_{A,K}$ of automorphisms of the theta group on the theta null point of A which leave invariant (via Π_2) the data (B,K) shows that the orbit of A is of degree $\ell^{g(g+1)/2}$ (see [FLR11]). On the other hand the above discussion shows that the open locus S of $\overline{S}$ given by $\lambda_i \neq 0$ is isomorphic to the scheme defined by Equation (5.8), which is of degree $\ell^{g(g+1)/2}$ so we have equality. In particular the action of $\mathcal{G}_{A,K}$ on $\Pi_2^{-1}(B,K)$ is transitive, as expected since the action of the automorphisms of the theta group on the fibers of π_2 is already transitive.

Concretely, to a geometric point of S, the corresponding point Q of $\overline{\Pi}_2^{-1}(B,K)$ is constructed as follow: first use Equations (5.5) and (5.6) to compute all λ_i, $i \in Z(\bar{\ell})$, and then set $Q = (\lambda_i\theta_j(P_i))_{i\in Z(\bar{\ell}),j\in Z(\overline{n})}$. This is exactly the method of Section 2.10.1. □

Remark 5.2.3. Over a local ring (R,m), the same method can be used to describe $\Pi_2^{-1}(B,K)$. In this case we need to take primitive affine lifts of the points P_i in the kernel (meaning that their reduction modulo m is not trivial), and the condition $\lambda_i \neq 0$ becomes $\lambda_i \notin m$, so that the λ_i are inversible in R. Indeed an abelian scheme over R has good reduction over $k = R/m$, so the image of the theta null point of level ℓn on B by Π_1 has to be well defined over k.

And so we recover once again the isogeny algorithm of Section 4.4

Theorem 5.2.4. *Let n be an even integer greater or equal to 4 and ℓ be an integer prime to n. The image of $\Pi_2 \times \pi_1 : \mathcal{A}_{g,\ell n} \to \mathcal{J}_{g,n,\ell} \times \mathcal{A}_{g,n}$ induces a modular correspondence defined over $\mathbb{Z}[\frac{1}{\ell n}]$.*

Let k be a field of characteristic prime to ℓn. If (B,K) is a k-point of $\mathcal{J}_{g,n,\ell}$, then $\pi_1 \circ \Pi_2^{-1}(B,K)$ only has a single $\bar{k}$-point (with multiplicity ℓ^g and which is actually defined over k), corresponding to $A = B/K$.

Proof. The first part follows from the steps above. For the statement over a field k, by construction, each geometric point in $\Pi_2^{-1}(B,K)$ corresponds to $A = B/K$ with a level ℓn structure compatible with the level n structure on B. Descending the product level ℓn structure via F then induces the same level n structure on A.

□

Let us extend this construction to $\mathcal{X}_{g,\ell n}$, ie study the fibers of $\pi_2 : \mathcal{X}_{g,\ell n} \to \mathcal{X}_{g,n}$, once we have chosen a point A (encoded by its theta null point of level ℓn) in $\Pi_2^{-1}(B,K)$. If P is a point of B, we will also explain how to compute $f(P)$, where $f : B \to A = B/K$ is the corresponding isogeny.

The action by translation of the finite group scheme K on P yields a subscheme of $P+K$ of B^{ℓ^g} and one can then consider the points of the fiber $\Pi_2^{-1}(P+K)$. Then we descend this subscheme via π_1, ie by first using the Segre embedding, then computing first ${}^tF$ formally on this subscheme by using the addition law, so without changing level (this step was actually unnecessary for the theta null point since ${}^tF(0) = 0$), and then F by using Koizumi's relation to go from level ℓn to level n. This gives a scheme whose unique point is $f(P)$ with multiplicity $\ell^{g(g+1)/2}\ell^g$ (over each of the $\ell^{g(g+1)/2}$ geometric point of $\Pi_2^{-1}(B,K)$ in $\mathcal{A}_{g,\ell n}$, the fiber $\Pi_2^{-1}(P+K)$ in $\mathcal{X}_{g,\ell n}$ is of degree ℓ^g.

Of course we may fix A and compute the fiber $\Pi_2^{-1}(P+K)$ over A to get a scheme of multiplicity only ℓ^g.) The strategy extends to $(\mathcal{B},\mathcal{K})/\operatorname{Spec} R$ an R-point of $\mathcal{J}_{g,n,\ell}$, and P a R'-point of $\mathcal{B}$.

Concretely, if (B,K) is a k-point of $\mathcal{J}_{g,n,\ell}$ and P a k-point of B, $f: B \to A = B/K$ the isogeny, then we have that $\tilde{f}: (A,\mathcal{L}^\ell) \to (B,\mathcal{M})$ given by the ℓ-contragredient isogeny is the pullback of π_2. The abelian variety A is given by its theta null point of level ℓn, a point (possibly defined over an extension) of $\Pi_2^{-1}(B,K)$. Then the geometric points of $\Pi_2^{-1}(P+K)$ over A are the preimages $f^{-1}(P)$. We can give equations for these preimages as before, by introducing the projective point $Q = (\mu_i\theta_j(P+P_i))_{i\in Z(\bar{\ell}),j\in Zn}$, where μ_i are indeterminates and we recall that P_i are the geometric points of K. If $Q \in \mathcal{X}_{g,\ell n}$ (rather than just $\overline{\mathcal{X}}_{g,\ell n}$), then we have $\mu_i \neq 0$. Plugging the equations of A (given by Riemann relations) give equations for the μ_i. More precisely differential additions and three way additions give equations of the form $\mu_{i+j} = \epsilon_{ij}\mu_i\mu_j$. So all the μ_i are determined from μ_{e_i} where $(e_1,\dots,e_g)$ is a basis of $Z(\bar{\ell})$ and we have equations of degree ℓ: $\mu_{e_i}^\ell = C_i'$. The whole system of equations is of degree ℓ^g so we do have described all preimages of f. With the notations of Section 2.10.1, the equations on the μ_i encode the possible excellent lifts. By definition of the contragredient isogeny, to compute $\tilde{f}(P)$ we just need to compute the multiplication by $[\ell]$ on any of these preimages, which we do as a composition of ${}^tF$ and F (via the Koizumi-Kempf isogeny formula for the latter) as explained above. We obtain a non reduced scheme of degree ℓ^g with only one point: $\tilde{f}(P)$. Over a local ring (R,m), we would use the same strategy except that we ask that the μ_i are non zero in R/m.

5.2.3 *Automorphisms of the modular correspondance*

In Section 5.2.1, we have defined $\mathcal{A}_{g,n,\Gamma_0(\ell)}$ and $\mathcal{A}_{g,n,\Gamma_1(\ell)}$ as the images of $\pi : \mathcal{A}_{g,\ell n} \to \mathcal{A}_{g,n}\times\mathcal{A}_{g,n}$ and $\Pi : \mathcal{A}_{g,\ell n} \to \mathcal{A}_{g,n}\times\mathcal{A}_{g,n}^{\ell^g}$ respectively. We may also try to construct them as $\Gamma/\Gamma_0(\ell)$ and $\Gamma/\Gamma_1(\ell)$ covers of $\mathcal{A}_{g,n}$ directly.

First let us study the automorphisms of the fiber $\pi_2^{-1}(B,\mathcal{M}) \subset \mathcal{A}_{g,\ell n}$. Let $(A,\mathcal{L}^\ell)$ be a point of this fiber. The symmetric automorphisms of the Heisenberg group of level ℓn acts on A, and if γ is such an automorphism, the resulting theta null point $\gamma\cdot 0_A$ is still in the fiber if it is compatible with the theta structure of level n on B.

Let us assume ℓ prime to n for simplicity (so ℓ is odd), then γ has to preserve the symmetric level n structure on A (induced by the symmetric level ℓn structure), hence is also an automorphism preserving π_1 (ie $\pi_1(\gamma\cdot 0_A) = \pi_1(0_A)$). But from our assumptions, these automorphisms are canonically identified with $\operatorname{Sp}(A[\ell], e_{\mathcal{L}^\ell})$, by sending a symplectic automorphism $\overline{\gamma}$ of $A[\ell]$ to the unique symmetric automorphism γ of $G(\mathcal{L}^\ell)$ which respect the symmetric level n structure and whose restriction to $A[\ell]$ is $\overline{\gamma}$ (the converse mapping is simply the restriction $\gamma \mapsto \overline{\gamma} \mid A[\ell]$).

It is convenient to identify an affine lift $\tilde{0}_A = (\theta_i^{\mathcal{L}^\ell}(0_A))_{i\in Z(\overline{\ell n})}$ with the affine points $\widetilde{P}_i = \left((\theta_{i+j}^{\mathcal{L}^\ell}(0_A))_{j\in Z(\overline{n})}\right)_{i\in Z(\bar{\ell})}$ on $\widetilde{B}$.

Then using Remark 2.6.6 we can describe the automorphisms coming from $\operatorname{Sp}(A[\ell], e_{\mathcal{L}^\ell})$ as follow.

- The matrix S, which transposes $A_1[\ell]$ and $A_2[\ell]$. This acts by $S\cdot\theta_i = \sum_{j\in Z(\bar{\ell})}\langle -j,\sigma(i)\rangle\theta_{i+j}$. In particular, if we denote by $\pi_3 = \pi_2\circ S$, then $C := \pi_3(A,\mathcal{L}^\ell)$ corresponds to the quotient of A by $A_1[\ell]$.
- Matrices of the form $\left(\begin{smallmatrix} a & 0 \\ 0 & {}^t a^{-1}\end{smallmatrix}\right)$. This simply permutes the $\widetilde{P}_i$, and this also stabilizes C.
- Matrices of the form $\left(\begin{smallmatrix} 1 & 0 \\ c & 1\end{smallmatrix}\right)$. Then this sends $\widetilde{P}_i$ to $-\langle i,-\psi_0(i)\rangle\widetilde{P}_i$ where $\psi_0 : Z(\bar{\ell}) \to \hat{Z}(\bar{\ell})$ is the mapping induced by c.

  These automorphisms fixes $A_2[\ell]$, hence preserve Π_2, but change the symplectic supplement $A_1[\ell]$ hence does not preserve π_3.

The matrices of the third type generate $\Gamma_1(\ell)/\Gamma(\ell)$, while those of the second and third type generate $\Gamma_0(\ell)/\Gamma(\ell)$.

We refer to [FLR11] for more details, and to [Rob10, § 6.3] for the case when ℓ is no longer prime to n (and even to general δ-structures). Note that both these references study the fibers of π_2 (and π_3) but only consider the modular correspondance induced by $\pi_2\times\pi_3$ (hence ℓ^2-isogenies) rather than the one from $\pi_1\times\pi_2$ (hence ℓ-isogenies) as we do here. In [FLR11] (again studied in more detail and extended to ℓ not prime to n in [Rob10, Chapiter 6]) we also classify degenerate points of the fiber $\pi_2^{-1}(B,\mathcal{M})$, ie points that live in $\overline{\mathcal{X}}_{g,\ell n}$ but not in $\mathcal{X}_{g,\ell n}$. They broadly are of two types: degenerate points where the P_i no longer generate a full group of order ℓ^g in B (typically copies of 0_B), and degenerate points where $P_i = (0,\dots,0)$ is not well defined as a projective point. These degenerate points are precisely points of the fiber where the action of $\Gamma_0(\ell)/\Gamma(\ell)$ is not free.

A main problem in the modular correspondence $\mathcal{A}_{g,\ell n} \to \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$ we defined is that the modular correspondance $\mathcal{A}_{g,n,\Gamma_0(\ell)}$ we really want is only given by the image of $\mathcal{A}_{g,\ell n}$. Computing this image may be done by a Groebner basis algorithm, but in practice this is very expensive. So we use the Riemann equations of $\mathcal{A}_{g,\ell n}$ instead (or rather $\overline{\mathcal{A}_{g,\ell n}}$) but this means that the fibers encode a full level ℓ structure along with the isogenies, hence the automorphisms we have just studied.

It would be interesting to construct $\mathcal{A}_{g,n,\Gamma_0(\ell)}$ as the quotient of $\mathcal{A}_{g,\ell n}$ by these automorphisms. We would need to define elements invariant under the actions of $\Gamma_0(\ell)/\Gamma(\ell)$, typically by first taking suitable products like $\prod_j \theta_{n_j i}$ with $\sum n_j^2 = \ell$ to be invariant under $\Gamma_1(\ell)/\Gamma(\ell)$ and then taking traces under $\Gamma_0(\ell)/\Gamma_1(\ell)$. We want to find invariants such that we still can compute equations on these invariants and also descend the maps π_1 and π_2 explicitly through them. This would give an alternative construction of $\mathcal{A}_{g,n,\Gamma_0(\ell)}$, as a quotient rather than as an image.

5.3 MODULAR POLYNOMIALS

5.3.1 *Definition of the modular polynomials*

We may also look at modular polynomials. We will mainly focus on the cases $g = 1$ and especially $g = 2$, since these are the cases where non trivial modular polynomials have been computed. We recall that we give more details on Siegel and Hilbert moduli spaces in [Rob21, Chapter 5].

The Siegel case

We may look at the modular polynomials as defining a birational version of the modular correspondance $\mathcal{A}_{g,\Gamma_0(\ell)} \to \mathcal{A}_g \times \mathcal{A}_g$ (eg as the generic point of the image, since $\mathcal{A}_{g,\Gamma_0(\ell)}$ is birationally equivalent to its image through the modular correspondance).

For instance in genus 1, letting $j(\tau)$ be the usual j-invariant, the modular polynomial is given by $\Phi_\ell(X,Y)$ such that $\Phi_\ell(j(\tau),Y) = \prod_{\gamma\in\Gamma/\Gamma_0(\ell)}(Y - j(\frac{1}{\ell}\gamma\cdot\tau)$. We can also interpret $\Phi_\ell(j,Y)$ as the minimal polynomial of $j(\tau/\ell)$ over $\mathbb{C}(j) = \mathbb{C}(\mathcal{A}_1)$, as defining the extension field $\mathbb{C}(\mathcal{A}_{1,\Gamma_0(\ell)})/\mathbb{C}(\mathcal{A}_1)$, or as defining equations of the modular curve $X_0(\ell)$. It is well known that Φ_ℓ is symmetric, of degree $\ell+1$ in X and actually lives in $\mathbb{Z}[X,Y]$.

In genus $g = 2$, we fix three modular Igusa invariants j_1, j_2, j_3 and we may define modular polynomials such that the corresponding system of equations $\Phi_\ell(X_1,X_2,X_3,Y_1,Y_2,Y_3)$ is birational to $\mathcal{A}_{g,\Gamma_0(\ell)}$. (We cannot expect better since j_1, j_2, j_3 only induce a birational isomorphism of $\mathcal{A}_g$ with $\mathbb{A}^3_{\mathbb{C}}$.)

We may define $\Phi_{\ell,1}(X)$ as the minimal polynomial of $j_1(\tau/\ell)$ over $\mathbb{C}(j_1,j_2,j_3)$,: $\Phi_{\ell,1}(j_1(\tau),j_2(\tau),j_3(\tau),Y) = \prod_{\gamma\in\Gamma/\Gamma_0(\ell)}(Y - j_1(\frac{1}{\ell}\gamma\cdot\tau)$. It is of degree $\#\Gamma/\Gamma_0(\ell) = \ell^3+\ell^2+\ell+1$. Indeed, since $\Gamma_0(\ell)\subset\Gamma$ is a maximal subgroup when ℓ is prime, $\mathbb{C}(j_1,j_2,j_3)(j_1(\tau/\ell))$ is the function field $\mathbb{C}(\mathcal{A}_{g,\Gamma_0(\ell)})$. Hence, letting $j_i'(\tau) = j_i(\tau/\ell)$, we may then write polynomials such that $j_2' = \Phi_{\ell,2}(j_1,j_2,j_3)(j_1')$ and $j_3' = \Phi_{\ell,3}(j_1,j_2,j_3)(j_1')$.

One need to be careful here that while the $\Phi_{\ell,i}$ are polynomials in j_1', their coefficients in terms of the j_i are given by rational functions. Typically we take for Igusa invariants invariants such that the denominators corresponds to (powers of) the cusp form χ_{10}, the cusp form that cancels on the locus of product of elliptic curves (with their product polarisation). Then the denominators contain the equation of the Humber surface H_{ℓ^2} which corresponds to abelian surfaces ℓ-isogenous to a product of elliptic curves. This interpretation was given in [BL09]; see also [MR20b; MR19] for the denominators of Hilbert modular polynomials. We refer to [Gru10, Proposition 2.14] as to why this locus corresponds to the Humbert surface of discriminant ℓ^2. In brief: if we have a product $E_1\times E_2$ we have endomorphisms $(P,Q)\mapsto(n_1P + m_1\tilde{f}(Q), n_2 f(P) + m_2 Q)$ where $f: E_1\to E_2$ is an isogeny. If A is ℓ-isogenous to a product, pulling back these endomorphisms show that A contains an order of discriminant ℓ^2. Finally, the work of Kieffer [Kie20a] gives a complete description of the full denominator, in particular explain the origin of what we called "parasite factors" in [MR20b] (which mean extra factors than the one corresponding to the Humbert surface of discriminant ℓ^2), we explain this in Section 5.3.6.

In practice we replace the polynomials $\Phi_{\ell,2}, \Phi_{\ell,3}$ with polynomials $\Psi_{\ell,2}, \Psi_{\ell,3}$ such that $j_i'\Phi_{\ell,1}'(j_1,j_2,j_3)(j_1') = \Psi_{\ell,i}(j_1,j_2,j_3)(j_1')$ where the derivative is taken with respect to the variable j_1'. This is the so called Hecke representation, also used to represent class polynomials in genus 2 (see eg [GHK+06]) and which yields smaller polynomials. It is also sometimes convenient to work with level structure, eg to work with theta constants. For instance the four theta constants of level 2 are particularly convenient since they give a projective birational equivalence between $\mathcal{A}_{g,2,4_{\mathbb{C}}}$ and $\mathbb{P}^3_{\mathbb{C}}$.

The Hilbert case

In [MR20b] we define modular polynomials on Hilbert and Humbert surfaces (ie moduli corresponding to real endomorphism) parametrizing β-isogenies, but the definition extends easily to all dimensions.

If we fix a real quadratic maximal order $O_{\mathcal{F}}$, the Hilbert surface parametrizes abelian surface with real multiplication by O_K. Then if β is a totally positive element of $O_{\mathcal{F}}$, we can define β-modular polynomials parametrizing β-isogenies (preserving the real multiplications). In this case we have $\Gamma = \mathrm{Sl}_2(O_{\mathcal{F}} \oplus \delta_{\mathbb{F}})$ and $\Gamma/\Gamma_0(\beta)$ is of index $N(\beta)+1$. Note that if ℓ is an inert prime we only have ℓ^2+1 ℓ-isogenies preserving the real multiplication among the $\ell^3+\ell^2+\ell+1$ ℓ-isogenies, that is such that the kernel K is stable by $O_{\mathcal{F}}$, so we get modular polynomials of smaller degrees.

For instance, if g_1, g_2 are Gundlach invariants, we define for $\beta \gg 0$ totally positive in the real order $O_{\mathcal{F}}$, $\Phi_{\beta,1}(g_1(\tau), g_2(\tau), Y) = \prod_{\gamma \in \Gamma/\Gamma_0(\beta)} (Y - g_1(\frac{1}{\beta}\gamma \cdot \tau))$.

5.3.2 *Computing Siegel modular polynomials in dimension* 2

The principle behind the computation of modular polynomials is straightforward: we proceed by evaluation-interpolation over $\mathbb{C}$. The actual implementation is far from trivial through: to get a quasi-optimal algorithm in the size of the modular polynomials, we need:

- Fast evaluation of the modular invariants $j_i(\tau)$. This allows to do the evaluation part: compute the $j_i(\tau)$ and all $j_1(\frac{1}{\ell}\gamma \cdot \tau)$, so that we can write $\Phi_{\ell,1}(j_i(\tau))(X) = \sum c_i(\tau) X^i$, and likewise for the $\Psi_{\ell,i}$.
- Fast interpolation of the coefficients $c_i(\tau)$ as rational functions in the $j_i(\tau)$ (or as a polynomial over $\mathbb{Z}$ in $j(\tau)$ when $g=1$).

  There is no difficulty for the interpolation when $g=1$, but when $g=2$ not only do we have to interpolate a rational function, but to get a fast interpolation we need to be able to choose the values of the j_i.

  Indeed to interpolate a multivariate polynomial $P(x_1, \dots, x_n)$ with n variables $x_1, \dots, x_n$, the strategy requires to fix the values of $x_1, \dots, x_{n-1}$ and compute P with d_n+1 different values of x_n (where d_n is the degree of x_n in P) so that we can interpolate $P(x_1, \dots, x_n)$ as $P(x_1, \dots, x_{n-1})(X_n)$ and iterate through x_{n-1}, …, x_1. We refer to [Mil15b, § 4.1.1] for details.

So the interpolation step requires being able to fix the values $j_1(\tau), j_2(\tau), j_3(\tau)$ and from this compute the values of the $j_i(\frac{1}{\ell}\gamma \cdot \tau)$ needed for the evaluation. In practice we use an algorithm due to Dupont [Dup06] which recovers the matrix τ (in the fundamental domain).

More precisely Dupont gives an algorithm that recovers a period matrix τ from the value of the level 2 theta functions $b_i(\tau) = \theta_i(\tau/2)/\theta_0(\tau/2)$ $i = 1,2,3$ (with his indexing), and to evaluate level 2 or level 4 theta functions from a period matrix in quasi-linear time with respect to the needed precision. Since the Igusa invariants are given by explicit polynomials in terms of the theta, we also have fast evaluation and fast period matrix computation for these invariants, see Section 5.3.4.

Using Dupont's algorithm, we can implement the evaluation-interpolation algorithm for modular polynomials, in time quasi-linear in their size. This was done by Milio in his thesis [Mil15a; Mil15b]. This generalize the quasi-linear computation done in [Eng09a] when $g=1$.

Modular polynomials using Igusa's invariant are too big, so I suggested to Milio to compute modular polynomials in terms of the theta functions instead (more precisely the $b_i(\tau)$). In particular I explained how we could use the automorphisms of the theta group to show that $\Psi_{\ell,2} = \Psi_{\ell,3}$ and I suggested to look at these automorphisms to also explain the other symmetries and vanishing of coefficients he observed, which he did beautifully in [Mil15a, § 5.2 and 5.3]. Note that using [Kie20a], I can now prove Milio's Conjecture 41 that there is no "parasite" factors in the denominator: since the theta functions are of the same weight, the "rewriting" procedure of [Kie20a] does not introduce parasite factors.

In his PhD, Milio computed ℓ-modular polynomial with $\ell = 2,3$ using Streng's version of Igusa invariants (the case $\ell=2$ was already done by Dupont in [Dup06] using less effective Igusa invariants), and $\ell = 3,5,7$ for theta invariants. The polynomials for $\ell = 7$ already take several GB, we will see why in Section 5.3.7.

5.3.3 *Computing Hilbert modular polynomials in dimension* 2

The same methods extend to compute Hilbert modular polynomials in quasi-linear time in genus 2 [MR20b]. We use the forgetful map from the Hilbert surface to the Siegel threefold to do a fast evaluation of (symmetric)

Hilbert modular functions and compute their period matrix $\tau = (\tau_1, \tau_2) \in \mathfrak{H}_1^2$ form the values of the modular functions. Indeed to τ we can associate an explicit Siegel period matrix Ω_τ, see [MR20b, § 2.3]. We use Humbert's lemma for the converse: if Ω is a matrix which is $\mathrm{Sp}_{2g}(\mathbb{Z})$ equivalent to a matrix of the form Ω_τ, Humber's lemma gives an algorithm to find a matrix in $\mathrm{Sp}_{2g}(\mathbb{Z})$ inducing such an equivalence, from which it is easy to recover τ ($\mathrm{Sl}_2(O_{\mathcal{F}} \oplus \delta_{\mathcal{F}})$ embeds into $\mathrm{Sp}_{2g}(\mathbb{Z})$, but is of dimension strictly less, so a matrix Ω equivalent to Ω_τ will not be of the form $\Omega_{\tau'}$ for a τ' equivalent to τ in general).

Some quick remarks:

- If $f(\tau)$ is a function of level Γ, then $f(\tau/\beta)$ is of level $\Gamma_0(\beta)$, so this provides a convenient way to get modular functions for $\Gamma_0(\beta)$. Furthermore $\Gamma_0(\beta) \subset \Gamma(\beta)$ is a maximal subgroup [MR20b, Proposition 4.11], so it is easy to get primitive elements. However, unlike in the Siegel case, this is not true if we work with a level subgroup Γ' of Γ (typically to work directly with theta functions), if $f(\tau)$ is invariant for Γ', $f(\tau/\beta)$ may be invariant by a smaller subgroup of $\Gamma' \cap \Gamma_0(\beta)$, even if β is prime to the level. See [MR20b, § 4.3] for a discussion of this.

- Since we evaluate modular functions by going through the Siegel moduli, this naturally allows to evaluate symmetric Hilbert modular functions. We recall that an Hilbert modular function $f(\tau)$ is symmetric if $f(\tau_1, \tau_2) = f(\tau_2, \tau_1)$, this means that conjugating the embedding $O_{\mathcal{F}} \to \mathrm{End}(A)$ by the Galois action does not change the value of f. There are three approaches to deal with non symmetry, all discussed in [MR20b]:
  1. We only use symmetric modular functions, so we cannot distinguish between β and β' isogenies where β' is the real conjugate of β, and we compute symmetric Hilbert modular polynomials which parametrizes both β and β'-isogenies;
  2. We work with a level structure that allows to distinguish the symmetric morphism even on the Siegel side (this means that the matrix M_σ from [MR20b, p. 10] corresponding to the symmetry is not in the level subgroup).
  3. We use (at least) one non symmetric modular function f in the modular polynomials. Its minimal polynomial over the symmetric modular functions is of degree 2 and we can evaluate it since its coefficients are given by symmetric modular function. We can then evaluate f by using its Fourier coefficients to compute the correct root at low precision and then augmenting the precision via Newton iterations. Likewise for the period matrix.

- More generally we could define I-modular polynomials for I an ideal of $O_{\mathcal{F}}$. Then the I-isogenous variety would have a polarisation of type $[I]$, the value of I in the narrow class group $\mathrm{Cl}^+(O_{\mathcal{F}})$. The polarisation is principal if and only if I is a trivial element of the narrow class group, ie is of the form $I = (\beta)$ with $\beta \gg 0$, which explain why we have restricted to this case. In term of Hecke correspondances, the general case of I-isogenies means that the Hecke correspondance may send a connected component of the corresponding Shimura variety to another one, see [Kie20a, § 2.4]. Still, the general case would be useful to fully explore isogeny graphs, see Section 5.6.

There is also the question of which Hilbert modular functions to use. For small discriminants, the Humbert surface is rational, so since it is of dimension 2 we can use two invariants, which we call Gundlach invariants since they were defined by Gundlach in the case $K = \mathbb{Q}(\sqrt{5})$ and $\mathbb{Q}(\sqrt{2})$. In general, we can always take pullback of Igusa invariants, provided they are defined on the generic point of the Humbert surface (this is always the case with Streng's invariant since the zero locus H_1 of χ_{10} does not contain the Humbert surface H_Δ). Likewise for pullback of level 2 theta constants. But then these pullbacks are not algebraically independent over the surface, so this slightly complicate the interpolation step. We present a general strategy in Section 5.3.5.

The computation itself was done by Milio using Plafrim, the cluster at University of Bordeaux. He computed modular polynomials (in Gundlach or theta invariants) with β of norm ℓ with ℓ up to $\ell = 97$ for $\mathbb{Q}(\sqrt{2})$ and $\ell = 59$ for $\mathbb{Q}(\sqrt{5})$. Once again we refer to Section 5.3.7 as to why these are so much smaller than in the Siegel case.

5.3.4 *Evaluating modular functions and period matrices*

Let us describe briefly Dupont's algorithm to evaluate theta constants and period matrices [Dup06] when $g = 2$ and generalisations [Lab16].

Starting with the affine point $\theta_i(0, A_\tau) := \theta_i(\tau/2)$, $i = 0, 1, 2, 3$, the duplication formula $\tau \mapsto 2\tau$ relate them with the affine point $\theta_i(0, A_{2\tau}) = \theta_i(\tau)$. A subtility here is that there are several choices of roots possible in the

duplication formula, ie of choice of signs, and (essentially) all possible choices are valid in genus 2, ie correspond to a duplication formula for τ up by acting on τ by a matrix M in $\Gamma(2,4)$. Following a terminology of Labrande, I will call a good choice of sign for τ the choice of signs which give 2τ. Dupont defines what I will call here a topological choice of of sign as the choice of sign that gets the values closer together in the complex plan (Dupont calls this a "bon choix de racine" but this conflicts with our terminology.) He shows that if τ has large imaginary part, or τ is in the standard fundamental Siegel domain, then this topological choice of sign is good, ie do correspond to $\tau \mapsto 2\tau$. So starting with such a τ, and iterating these topological choices we get the values $\theta_i(2^n\tau)$ which converge to $(1,0\dots,0)$. But the duplication formula are homogeneous, so if we start with $\lambda\theta_i(\tau/2)$, in particular if we start with the three values $b_i(\tau) := \theta_i(\tau/2)/\theta_0(\tau/2)$ the limit allows to recover λ. Hence we may see the generalised AGM as a way to compute a good affine theta null point (ie coming from a τ) from the projective one. Then for a matrix $M = \begin{pmatrix} a & c \\ b & d \end{pmatrix}$, we can compute the values $b_i(M\cdot\tau) = \theta_i(M\cdot\tau/2)/\theta_0(M\cdot\tau/2)$ (eg using Remark 2.6.6). If M is a matrix such that, for the image of the fundamental domain by M, the choice of topological signs still correspond to the duplication $\tau \to 2\tau$, then using the AGM again for these new projective thetas allows to recover $\theta_0(M\cdot\tau/2)$. Using modularity of the theta functions, this gives us $\det(c\tau+d)$. With appropriate choices of M as specified in [Dup06], this allows to recover τ. This gives a quasi-linear algorithm to get τ from the $b_i(\tau)$. Inverting this algorithm via Newton iterations yields a quasi-linear algorithm to compute the $b_i(\tau)$, hence the $\theta_i(\tau)$ from τ. It is remarkable that the fast evaluation of theta functions proceed through the inversion of the computation of the period matrix rather than the converse.

Let us give some quick remarks on this algorithm:

- While Dupont specify a set of four matrices $M_1 = \mathrm{Id}, M_2, \dots, M_4$ used to recover τ, he could only prove that, in the image of the standard fundamental domain by M the good choices of sign correspond to the duplication of τ, for $M = \mathrm{Id}$. So his algorithm was heuristic, but this fact has been recently proved by Kieffer in [Kie20c], by a careful study of the behaviour of the theta functions seen as modular forms (I recommend to look at the nice pictures).

- In Dupont's original algorithm, he does not work with the theta functions of level 2 $\theta_i(\tau/2)$ but rather with the squares $\theta_i(\tau)^2$ of the first four theta functions of level 4 (or rather level $(2,2)$). The duplication formula allows to go back and forth from products $\theta_i(\tau/2)\theta_j(\tau/2)$ of two theta functions of level 2 with the squares of all 16 theta functions of level $(2,2)$. So only taking the first four squares forget some information, that Dupont has to recover afterwards. This does not change the time complexity of the algorithm, but makes it slightly harder to implement. This was done by Dupont because the resulting duplication formula on these four squares is equal to the Borchardt mean, so more closely ressembles the AGM in dimension 1.

In [LT16; Lab18; Lab16], Labrande extends these algorithms to compute the theta functions (of level 2) $\theta_i(z,\tau)$ in z and τ when $g \leq 2$, and conversely to recover z from these values. There are still some heuristics remainings, even for computing theta constants, but good progress has been made on some of them in [Kie20b; Kie20c].

The general strategy of [Lab16] is as follow: we use the duplication formula both in z and τ, this relates squares $\theta_i(z,2\tau)^2$ in term of the $\theta_i(z,\tau)$. A good choice of sign (in the terminology of Labrande) is the choice corresponding to $z \mod \Lambda_{2\tau}$ and 2τ. An algebraic choice of sign is a choice of sign corresponding to z' and $2\tau'$ for z' equivalent to z (ie $z = z' \mod \Lambda_\tau$ and τ' equivalent to τ). A good choice is algebraic, but not all sign choices are algebraic. A topological choice of sign is the choice of sign making the values closer in a quadrant.

Iterating good choices of sign, the values converge to 1. Using homogeneity (the normalisation is a bit more delicate, see [Lab16, § 6.2.4]), from the $\theta_i/\theta_0(0,\tau)$ and the $\theta_i/\theta_0(z,\tau)$, the generalised AGM (using good choices of sign) converge quadratically and gives $\theta_0(0,\tau)$ and $\theta_0(z,\tau)$. Using appropriate matrices action γ along with the full functional equation of θ_i both in z and τ, this gives explicit parameters depending on z and τ, hence allows to recover them.

A thorny question is how to choose the good choice of signs. When τ has large enough imaginary part, the topological choices are good. Otherwise, if z and τ are given in low precision, we can use the explicit Fourier series of theta functions to evaluate the $\theta_i(z,2^n\tau)$ at low precision to get the correct choice of signs. In good cases, one can identify a domain where all topological of sign are good. The best case is when this holds for the fundamental domain and its image by the action of the matrices γ used to recover z and τ. This is indeed the case when $g = 1$ for $z = 0$ [Dup06], and a general z [Lab16], and as we have seen when $g = 2$ for $z = 0$ by [Dup06] completed by [Kie20c]. In these good cases, there is no need for low precision computation.

Evaluating the theta functions $\theta_i(z,\tau)$ is done via a Newton iteration inversing the procedure above giving z,τ from the θ_i. The convergence of this Newton process is only heuristic when $g \geq 2$. Actually, the tangent spaces are not even of the same dimensions when $g > 2$ or $g = 2$ and $z \neq 0$, so the Jacobian of the system is not even

inversible. We would need to add the (tangent) equations of the moduli when $g > 3$ and of the Kummer when $g \geq 2$, but we don't have them for general g (see Section 2.13). A fun trick of [Lab16, Conjecture 7.4.4] is to simply add more relations using the matrices γ above.

Finally, this algorithm (assuming the Newton process converges) is quasi-linear in the precision, but not uniform (in z, τ). To get a uniform algorithm we assume that there is a uniform algorithm on a compact subset of the standard fundamental domain given by $\Im\tau \leq C$ for some constant C (this is more or less equivalent to assuming that the Newton iterations in Dupont's algorithm do converge on this subset).

Then for a general τ in the fundamental domain, either $\Im\tau \leq C$ so we can apply Dupont's algorithm directly, or $\Im\tau$ is large enough that a naive evaluation using the Fourier series is sufficiently fast, and in the remaining intermediate case we can apply a logarithmic number of well chosen duplication formula to get back to one of the first two cases. This strategy is done by Dupont when $g = 1$ and $z = 0$, and is extended by Labrande to a general z For $g = 2$ and $z = 0$ this is done by Kieffer in [Kie20b, § 4.1]. Furthermore when $g = 1$ the existence of the uniform convergence of the Newton process on a compact set is proven (when $z = 0$ by Dupont and for a general z by Labrande).

We summarize these results, letting N be the required precision.

- For all g, there is an improved version of the "naive" algorithm using the explicit Fourier series in $\widetilde{O}(N^{1+g/2})$ to evaluate the $\theta_i(z,\tau)$ [Lab16, Chapter 5]. When $g = 1, 2$, Labrande proves a complexity of $\widetilde{O}(N^{1+g/2}/\Im\tau_{1,1})$, and conjectures this hold for a general g, which is important for the uniform algorithm outlined above.

- When $g = 1$, there is a uniform and proved quasi-linear algorithm to compute the $\theta_i(z,\tau)$, or conversely z and τ, [Lab16, Chapter 6]. It does not require low precision approximations, because the proven domain of (z,τ) where all topological sign choices are good is large enough.

- When $g = 2$, there is a proven quasi-linear algorithm to compute τ (now that [Dup06, Conjecture 9.1] has been solved in [Kie20c]). The computation of z requires a low precision approximation to ensure the good choice, but we could probably extend [Kie20c] to show that all topological sign choices are good when z is in a nice enough domain.

  There is an heuristic (assuming convergence of the Newton process) quasi-linear algorithm to compute the $\theta_i(z,\tau)$. Again, without more knowledge of when topological choices are good for z, it requires evaluating the $\theta_i(z,\tau)$ to low precision when $z \neq 0$. Such an approximation can be computed via hyperelliptic integrals, see [MN17a; MN17b; Lab16, Chapter 8].

  Under the assumption of a uniform algorithm on a compact set, Kieffer shows uniformity in the fundamental domain [Kie20b, § 4.1] to compute theta constants. However, for a general τ, the standard reduction algorithm to the fundamental domain is only quadratic in the precision [Str10, § II.5.3] and [Kie20b, § 4.2], one would need to adapt [NSV11; NS16] to the symplectic case to get quasi-linear speed.

- For general g, Dupont proves the quadratic speed of convergence of the Borchardt mean, so we get a quasi-linear algorithm to compute τ, provided we have low precision approximation to choose the good choices of signs. The quadratic speed of convergence to get z is only heuristic [Lab16, Conjecture 7.4.3]. The fact that we can get enough relations so that the Jacobian of the equations becomes inversible is heuristic [Lab16, Conjecture 7.4.4], let alone the convergence of the Newton process to compute the $\theta_i(z,\tau)$.

  Still, Labrande conjectures the existence of a quasi-linear algorithm to compute the $\theta_i(z,\tau)$. The converse is more delicate: without more knowledge on the good vs topological choices, we need low precision approximations of z and τ. But if we are not on a Jacobian of an hyperelliptic curve, we cannot rely on hyperelliptic integrals to compute these approximations. We go back to this topic in Section 5.7.

We now explain how we could compute modular forms of higher level and period matrices from modular invariants, if we assume that we can do these computations for theta functions of level 2. The coordinate ring of modular forms is integral above the coordinate ring of theta constants (at least in characteristic zero) [Igu64; Igu66]. More precisely, Igusa proves that the modular forms of level $\Gamma(n,2n)$ is the integral closure of the ring of theta constants of level n. So if we pick up some modular invariant of a given level, and we know its minimal polynomial over the theta ring, we can evaluate it from Newton iterations, using its Fourier series at low precision for the initialisation. So we reduce to evaluating theta functions of level n. But change of level formulae or Mumford's isogeny theorem (see Section 5.2) gives us equations over theta functions of level 2, from which we can also evaluate via Newton iterations and small precision initialisation. Likewise, if we have values of the modular invariants, we plug the modular relations with the theta, so we get the theta values of level n. Then we can compute those of level 2, compute the corresponding period matrix modulo $\Gamma(2,4)$ and lift it to a period matrix Ω' modulo $\Gamma(n,2n)$.

Evaluating the theta on Ω' and comparing with our current values then give us, using the functional equation of theta functions, how to correct the Ω' to get the correct representative (this is faster than testing all the lifts).

Alternatively, we can use a fast algorithm to evaluate theta functions $\theta(z,\tau)$ rather than just the theta constants $\theta(0,\tau)$, and conversely to recover z from the values of the thetas. Then the theta constants of level $2n$ can be computed by evaluating the theta functions of level 2 at a n-torsion point.

5.3.5 *An evaluation-interpolation approach for covers and modular polynomials*

One reason that the evaluation-interpolation approach works well in dimension 1 and 2, is that the moduli space $\mathcal{A}_g$ is rational (ie birational to $\mathbb{P}^N$), so we can describe it using only primary invariants $j_1,\dots,j_N$. This works even for small level. For elliptic curve at level 1 we have the j-invariant, at level $\Gamma(2)$ we have the Legendre invariant λ, and at level $\Gamma(2,4)$ we have the theta function θ_1/θ_0. For abelian surfaces at level 1 we have the three Igusa invariants j_1,j_2,j_3, at level $\Gamma(2)$ the three Rosenhain invariants λ,μ,ν and at level $\Gamma(2,4)$ the three theta functions $b_i=\theta_i/\theta_0$, $i=1,2,3$. Also Hilbert surfaces or Humbert surfaces of small discriminant are rational, so we have two Gundlach invariants. Still we cannot expect this to hold for all moduli spaces on which we want to construct modular/Hecke correspondances (if only because $\mathcal{A}_g$ is of general type for $g\geq 7$), so we need to explain how to do interpolation when we also have secondary invariants. For Hilbert or Humbert surfaces it is also often convenient to look at pullback of Igusa invariants or theta functions, so we have a non trivial relation, ie a secondary invariant.

Here we focus on this general case. More generally if we have a finite separable G-cover $X\to Y$ over a perfect field k, where X,Y are integral of dimension d, and we have coordinates $x_1,\dots x_m$ on X we want to express birationaly in term of coordinates $y_1,\dots,y_n$ on Y, we may introduce modular polynomials as follow: since $k(X)/k(Y)$ is finite, we can write $k(X)=k(Y)(x_0)$ by the primitive element theorem. Then $k(X)$ is characterised by the minimal polynomial $\Phi(X)$ of x_0 over $k(Y)$: $\Phi(X)=\prod_{g\in G}(X-g\cdot x_0)$. Then we use the Hecke representation $x_i\Phi'(x_0)=\Psi_i(y_i,x_0)$ to represent the elements x_i: $\Psi_i(X)=\sum_{g\in G} g\cdot x_i\prod_{h\in G|h\neq g}(X-h\cdot x_0)=\sum_{g\in G} g\cdot x_i\Phi(X)/(X-g\cdot x_0)$.

In particular we can apply this to the cover given by the $\Gamma_0(\ell)$ level structure, and more generally for covers of Shimura varieties. Typically we have y_i given by $j_i(\tau)$, and the x_i given by the $j_i(\tau/\ell)$. This representation was introduced in [MR20b, § 3.4] for the case of covers of Hilbert surfaces (so $\Gamma_0(\beta)$-covers); see also [Kie20a] for an extension of this representation where rather than representing $k(X)$ by $k(Y)(x_0)$ we allow intermediate fields $k(Y)(x_0')\subset k(Y)(x_0',x_1')\subset\dots k(X)$.

We present an evaluation-interpolation approach to compute birational equations Φ,Ψ_i for this cover. We assume that we know how to compute the fibers of $X\to Y$. Then given a point y of Y, we compute all the points in the fiber X_y, and compute a representation of X_y as follow: generically x_0 will separate the coordinates, and we use the Hecke representation to represent the other coordinates. This gives polynomials in $x_0,\dots,x_m$, whose coefficients c_i are rational functions of Y evaluated at y.

It remain to explain how to recover the coefficients c_i explicitly. For simplicity, since Y is an integral variety of dimension d, we assume that we have expressed $k(Y)$ as $k(Y)=k(y_1,\dots,y_d)[y_{d+1}]$ where $y_1,\dots,y_d$ form a transcendance basis and y_{d+1} is given by the primitive element theorem, and that we have the monic minimal polynomial P of degree N of y_{d+1} in terms of the y_i. In the terminology of invariant theory, $y_1,\dots,y_d$ are primary invariant and y_{d+1} is a secondary invariant. We have a function $f\in k(Y)$, on which we know how to compute the evaluation map $t\mapsto f(t)$.

A problem for the interpolation is that the presentation of f in terms of the y_i is not unique (since it is defined modulo P), so if at each evaluation we change the representation, these won't glue together for the interpolation. But of course there is a canonical presentation by looking at the representation with the minimal degree in y_{d+1} (so doing the standard euclidean division by P). Then if we have a way in the evaluation, given $y_1(t),\dots,y_d(t)$ to write all the N-roots of $P(y_1(t),\dots,y_d(t),Y)$ as values $y_{d+1}(t_r)$ (such that $y_i(t_r)=y_i(t)$ if $i\leq d$) for $r=1,\dots,N$, then we can do interpolation on y_{d+1} to write $f(t)=\sum_{k=0}^{N-1} c_k(y_1,\dots,y_d)y_{d+1}^k$. Then it suffices to proceed through standard rational interpolation of the c_k in terms of the $y_1,\dots,y_d$.

Of course this strategy could be extended when adding more secondary invariants (for instance to get an integral model of $X\to Y$ rather than a birational model), by fixing a Grobner basis and interpolating the representation of f given by this Grobner basis. This works when using evaluation points such that the Grobner basis of the evaluated functions is the evaluation of the Grobner basis (this is the generic case).

We remark that for the computation of modular polynomials the situation is slightly different: namely given $y\in Y$ we don't know how to compute the fiber X_y directly. Instead, we go through the universal covers $\mathbf{H}_g$ (Siegel case) or $\mathfrak{H}_1^g$ (Hilbert case). Let's take a look at Hilbert modular polynomials for instance: from the value of a modular

invariant $J(\tau)$ we first need to compute $\tau \in \mathfrak{H}_1^g$ (ie go back to the universal cover) to evaluate the $J(\gamma \cdot \tau/\beta)$. So if we want fast interpolation, this requires to compute τ from $J(\tau)$ in quasi-linear time, and then evaluate the $J(\gamma \cdot \tau/\beta)$ in quasi-linear time. If we start with τ directly, then we sample points in Y randomly, so we have to resort to linear algebra to find the relations. We could also do linear algebra on the Fourier coefficients rather than on evaluation points.

Remark 5.3.1. Still even non quasi-linear algorithms to compute modular polynomials will be useful in Section 5.3.8 and Chapter 6, so let us record this fact here: as long as we have an algorithm polynomial in the precision to evaluate the needed modular functions $J(\tau)$, $J(\tau/\ell)$ on "random" matrices τ (small enough that $J(\tau)$ and $J(\tau/\ell)$ are not too large) we have an algorithm to compute the modular polynomials in time polynomial in their size, by evaluation followed by interpolation via linear algebra.

The same holds if we can compute Fourier coefficients fast enough. And if J have integral Fourier coefficients we can do linear algebra over $\mathbb{Z}$ if we use the explicit denominators of the modular polynomial, see Section 5.3.6. This has the advantage that we can solve the linear system over $\mathbb{Z}$ by a CRT approach (modulo small primes p_i) to control the intermediate growth. This bypasses stability problems from doing linear algebra using complex floating point.

If the modular invariants J are expressed as polynomials of theta functions, these assumptions hold: we have the Fourier coefficients of the theta functions, and we can evaluate them at precision m in time $\widetilde{O}(m^C)$ with $C = 1 + g/2$ or, heuristically $C = 1 + \epsilon$, by [Lab16], see Section 5.3.4. More generally for the evaluation, we could ask for the minimal polynomials of the invariants J over the theta constants of level 2, see Section 5.3.8.

Anticipating Section 5.3.7, in the Siegel case we have $\ell^{N(N+1)}$ terms with $N = g(g+1)/2$ and the coefficients are of size $\widetilde{O}(\ell^N)$. If the invariants are given as polynomials in terms of the theta constant, the evaluation at precision $\widetilde{O}(\ell^N)$ costs $\widetilde{O}(\ell^{NC+N(N+1)})$, and the linear algebra (assuming the system is sufficiently stable) costs $\widetilde{O}(\ell^{N+N(N+1)w})$ where $w \leq 3$ is the exponent for the matrix multiplication. In the Hilbert case we have ℓ^{g+1} terms and the coefficients are of size $\widetilde{O}(\ell)$, for a total cost of $\widetilde{O}(\ell^{C+g+1} + \ell^{1+(g+1)w})$.

5.3.6 *Denominators of the modular polynomials*

A modular interpretation of the denominators

We have seen that in dimension 2 the coefficients $c_i(j_1, j_2, j_3)$ of the modular polynomials are given by rational functions. Having to interpolate a rational function is annoying from the complexity point of view: already in the univariate case fast algorithms rely on fast half gcd algorithms and even if they are still quasi-linear they introduce more logarithmic factors.

This is worse in the multivariate case, because during the evaluation simplifications between the numerators and denominators can prevent getting the correct interpolation. See the examples at [Mil15b, § 4.1.2]. The solution is to fix one coefficient, but to do that in a coherent way, rather than interpolating a multivariate function $F(X_1, \dots, X_n)$ we have to interpolate $F(X_1, X_1X_2, \dots, X_1X_n)$. We can then proceed variable by variable, interpolating with X_1 last. But this means that the last interpolation step depends on the total degree of F rather than its degree in X_1. We refer to [Mil15b, § 4.1.2] for some tricks and more details in the complexity analysis.

In truth we would really like to not interpolate rational functions. So instead of computing the monic polynomial $\Phi_{\ell,1}(X)$, we would like to compute $D\Phi_{\ell,1}(X)$ where D is (a multiple of) the denominator. But to do that we need a modular interpretation of the denominator. A wonderful key insight of Kieffer in [Kie20a; Kie20b] is that we need to interpret the denominator not as a rational function in the Igusa invariants (ie as a modular function of weight 0), but as a modular form.

In other words, we need to compute the modular correspondance for modular forms rather than for modular functions. Let us first look at the case of elliptic curves: the modular polynomial is given by $\Phi_\ell(j(\tau)) = \prod_{\gamma \in \Gamma/\Gamma_0(\ell)}(X - j(\gamma \cdot \tau/\ell))$. But $j(\tau) = cg_2(\tau)^3/\Delta(\tau)$ with $c = 1728$. So if we let $D(\tau) = \prod_{\gamma \in \Gamma/\Gamma_0(\ell)} \Delta(\gamma \cdot \tau/\ell)$, we get that $D(\tau)\Phi_\ell(j(\tau)) = \prod_{\gamma \in \Gamma/\Gamma_0(\ell)}(\Delta(\gamma \cdot \tau/\ell)X - cg_2(\gamma \cdot \tau/\ell))$. The problem is that $D(\tau)$ is not modular, and even depends on the choice of representatives γ. But all this can be fixed by normalizing with $\Delta(\gamma \cdot \tau)/\Delta(\tau)$, ie by considering instead $D(\tau) = \prod_{\gamma \in \Gamma/\Gamma_0(\ell)} \det(\gamma \cdot \tau)^{-w}\Delta(\gamma \cdot \tau/\ell)$ with $w = 12$ the weight of Δ. Then $D(\tau)$ is a modular form of weight $w(\ell + 1)$. Furthermore by [Kie20b, Proposition 3.2], $D(\tau)$ is defined over $\mathbb{Z}$.

The same holds in the genus 2 case. Here we typically take for Igusa invariants j_i, invariants having some power d of the cusp form $\chi_{10}(\tau)$ as their denominators. Then if we let $D(\tau) = \prod_{\gamma \in \Gamma/\Gamma_0(\ell)} \det(\gamma \cdot \tau)^{-dw}\chi_{10}^d(\gamma \cdot \tau/\ell)$ with $w = 10$ the weight of χ_{10}, we get a modular form of weight $dw(\ell^3 + \ell^2 + \ell + 1)$. Furthermore this denominator is exactly (a power of) the modular form associated to the Humbert surface H_{ℓ^2} of discriminant ℓ^2.

The parasite factors (as dubbed in [MR20b]) are then explained as the rewriting procedure when expressing the quotient of two Siegel modular forms of the same weight in terms of the j_1, j_2, j_3, see [Kie20a, § 3.2 and § 3.3].

We can now use the following modified strategy to compute modular polynomials:

- Rather than computing the modular functions $j_i(\gamma \cdot \tau/\ell)$, we compute the modular forms $\mathfrak{g}(\gamma \cdot \tau/\ell)$ for $\mathfrak{g} = I_2, I_4, I_6, I_{10}$. Dupont's algorithm naturally give the value of the θ_i (and not only the θ_i/θ_0), so since these modular forms are given by explicit polynomials in the θ_i we have a fast evaluation algorithm.
- Then we have coefficients $c_i(\tau)$ which are modular forms (of the same known weight) that we interpolate as polynomials in I_2, I_4, I_6, I_{10}. It might seem that this require to interpolate with 4 variables rather than 3, but by homogeneity, since we know the weights of the c_i, we can scale everything by λ such that for instance $\lambda^{10} I_{10}(\tau) = 1$. Then we interpolate a polynomial in I_2, I_4, I_6 and we multiply each term by the correct power of I_{10} so that the weight is the required one.

This new strategy has not yet been implemented ([Kie20b] only deal with evaluation, since this is enough for our applications as we will see below), but I expect this would give quite a practical speed up compared to the algorithms as implemented by Milio.

Affine modular polynomials

We stress an important point: the new strategy computes first the modular polynomials $D\Phi$ expressed in terms of modular forms of a suitable weight. I will call this the "integral modular polynomials". In genus 2 this typically involves the forms of weight 20: $I_4^5, I_{12} I_4^2, I_4 I_6 I_{10}$. Then we may quotient by the leading coefficient (ie the denominator) to get a modular polynomial Φ written in terms of modular functions.

We have the following modular interpretation of the correcting factors $g(\gamma \cdot \tau)/g(\tau)$ (g the modular form appearing in the denominator of the modular invariants J): the isogeny between τ and $\gamma \cdot \tau/\ell$ is not normalised, but the one between $\gamma \cdot \tau$ and $\gamma \cdot \tau/\ell$ is.

We may use the same trick to define an "affine modular polynomial" relating $\mathfrak{g}(\tau)$ and $\mathfrak{g}(\tau/\ell)$ for $\mathfrak{g}$ a scalar modular form of weight k. We define $\Phi_{\mathfrak{g},\ell}(\tau) = \prod (X - \frac{\mathfrak{g}(\tau)}{\mathfrak{g}(\gamma\tau)} \mathfrak{g}(\gamma\tau/\ell))$. Then the coefficients of $\Phi_{\mathfrak{g},\ell}$ are given by modular forms. Algebraically, $\Phi_{\mathfrak{g},\ell}(A, w_A) = \prod (X - \mathfrak{g}(B, w_B))$ where the product is over all ℓ-isogenies $f : A \to B$ with $f^* w_B = w_A$.

So $\Phi_{\mathfrak{g}}$ encodes more information than Φ: namely it allows to recover the determinant to the power k of the action of the isogeny on differentials. This works as follow: we start with (A, w_A) and compute $\mathfrak{g}(A, w_A)$. Using the modular polynomial $\Phi_{\mathfrak{g}}$ we can recover the value $\mathfrak{g}(B, w_B)$ where $f : A \to B$ is an isogeny where $f^* w_B = w_A$. Taking any differential w'_B on B, we can evaluate $\mathfrak{g}(B, w'_B)$ and compare with $\mathfrak{g}(B, w_B)$: $\mathfrak{g}(B, w'_B) = u\mathfrak{g}(B, w_B)$. Then $f^* w'_B = M w_A$ where $\det^k M = u$. We will exploit this in Section 6.4.

In summary: the affine modular polynomials encode the normalized ℓ-isogenies, ie the isogenies $f : A \to B$ where $f^* w_B = w_A$. This interpretation is still valid algebraically, and allows us to compute them directly over a finite field for instance (by using Chapter 4 and Section 5.6.2).

Remark 5.3.2. We have seen that in dimension 2, the invariants we use have for denominator powers of χ_{10}, whose zero locus corresponds to product of elliptic curves. Thus if A is ℓ-isogenous to a B which is a product of elliptic curves, $D(J(A)) = 0$. So the evaluated modular polynomial $D\Phi_\ell(J(A), X)$ is of degree less than usual. Looking at the equation of $D\Phi_\ell$, we get that $\Phi_\ell(J(A), X) = D' \prod_B (X - J(B))$ where the product only iterate through the ℓ-isogenous B which are not product of elliptic curves.

So computing $D\Phi$ allows us to recover the isogenous B where the modular invariant J is well defined. There are two solutions to get all the isogenous B in every cases: either compute them for other system of invariants that fully cover $\mathcal{A}_g$, or better compute affine modular polynomials for a system of generators of covariants. In dimension 2 (and not too small characteristic), we could also use affine modular polynomials parametrizing the $\psi_4, \psi_6, \chi_{10}, \chi_{12}$ of the normalised isogenous variety directly, rather than just the three numerators of j_1, j_2, j_3.

Remark 5.3.3. We will see in Section 5.4 that the modular polynomials Φ_ℓ are actually sufficient to recover normalised isogenies. Indeed, if $f : A \to B$ is an isogeny, differentiating the modular equation $\Phi_\ell(J_A, J_B)$ gives a relation between dJ_A and dJ_B. When non singular, this relation allows to recover dJ_B from dJ_A and we give precise conditions in [KPR20] as to when this happens. But dJ is a vectorial form of weight Sym^2, hence it essentially allows to recover w (up to a sign). Starting with w_A, this allows to recover the normalised w_B (up to a sign), hence evaluate any scalar modular form $\mathfrak{g}$ on (B, w_B) (at least if $\mathfrak{g}$ is of even weight or the dimension is even), or even vectorial modular forms. A similar reasoning holds for Hilbert modular polynomials and Hilbert modular forms. So in practice we don't need the affine modular polynomial $\Phi_{\mathfrak{g},\ell}$ for the modular form $\mathfrak{g}$.

Denominators of Hilbert modular polynomials of abelian surfaces

Let us conclude this section by describing denominators of Hilbert modular polynomials. First the same strategy of introducing them as modular form will speed up their computation. What is more interesting is the modular interpretation of the denominator D_β of the β-modular polynomials for abelian surfaces. If we use modular invariants whose denominators are given by pullback of χ_{10}, then D_β parametrizes abelian surfaces that are β-isogenous to product of elliptic curves (with real multiplication). Pulling back the extra endomorphisms available on such a product, we expect that the abelian surface A has special endomorphisms (in addition to $\mathcal{O}_F$). This is indeed the case, in fact the Néron Severi group of A will be (generically) of rank 3.

In fact, like denominators of Siegel modular polynomials which correspond to the Humbert surface of special discriminant ℓ^2, denominators of Hilbert modular polynomials correspond to special generalised Humbert varieties. These were introduced by Kani in [Kan19a; Kan18; Kan19b] to study the decomposition of the intersection of two Humbert surfaces $H_n \cap H_m$ in irreducible components. Indeed Kani introduced a refined Humbert invariant on the Neron-Severi group $NS(A)$ of A, this is a quadratic form of rank $m-1$ where m is the rank of $NS(A)$. The generalised Humbert variety $H(q)$ then classify abelian surfaces whose generalised Humbert invariant is q, and the classical Humbert surfaces H_n correspond to $q(x) = nx^2$.

In the initial version of [MR20b], we had a section explaining in further details the link between (the components of) D_β and $H(q)$. Unfortunately we had to remove it for length reason, we should probably make a separate article about this (this is in project in [MR19]). It is still available as version v2 of the Hal url. In the rest of this section the references I give to [MR20b] are to this version.

First, if A is β-isogenous to a product of elliptic curves, it is actually m-isogenous to a product by [MR20b, Lemma 4.21]. So $D(\beta)$ lies in $H_\Delta \cap H_{m^2}$ where Δ is the discriminant of $\mathcal{O}_F$. We explain how to find possible values of m in [MR20b, Lemma 5.1]. So we can write $D(\beta)$ as an intersection of H_Δ and several H_{m^2}. Since $H(q)$ lies in any Humbert surface H_n such that q primitively represent n, we go through each form q that primitively represents Δ, and then check those who also primitively represent m^2 for the m appearing in the decomposition of $D(\beta)$ above.

A question in [MR20b, Conjecture 5.2] was whether the set of values m are the same on each irreducible component. But this can be proved using [Kan19b, Corollary 6], see [MR19, Theorem 1.2].

Example 5.3.4. For $\mathbb{Q}(\sqrt{2})$ and $\beta = 5 + 2\sqrt{2} \mid 17$, the denominator of $\Phi_{1,\beta}$ has for irreducible component the generalised Humbert variety $H(8x^2 + 4xy + 9y^2) = J_1^7 - J_1^6 J_2^3 - 6J_1^6 J_2^2 + J_1^6 J_2 + \dots$ which lies in

$$H_8 \cap H_{3^2} \cap H_{7^2} \cap H_{11^2} \cap H_{23^2} \cap H_{31^2} \dots .$$

In particular, an abelian surface β-isogenous to a product of elliptic curves is also ℓ-isogenous to such a product for $\ell = 3, 7, 11, 23, 31, \dots$.

5.3.7 Size of the modular polynomials

The degree of the modular polynomial of abelian surfaces in terms of j_1', j_2', j_3' are well understood (we recall that $j_i(\tau) := j_i(\tau/\ell)$). For instance for $\Phi_{1,\ell}$, this is of degree $\ell^3 + \ell^2 + \ell + 1$ in j_1'. What is less obvious is their height, and the degree of their coefficients $c_i(j_1, j_2, j_3)$ in term of j_1, j_2, j_3.

The discussion in Section 5.3.6 answers the first question: multiplying by $D(\tau)$, the coefficients c_i are modular forms of known weight, and it is easy to get the degree from this weight. A bound on the height has been beautifully given in [Kie20a]. Kieffer's method rely on Raynaud's height isogeny theorem [Ray85]: the Falting's height of two ℓ-isogenous abelian varieties over a number field differ by $O(\log \ell)$. This means that if $J(\tau)$ has small height $O(1)$, the height of the isogenous abelian surfaces have height bounded by $O(\log \ell)$.

If P is a polynomial of degree d in $\mathbb{Q}[X]$ such that $P(x_i)$ is of height bounded by H for $d+1$ points, then interpolation gives a bound of $O(dH)$ for the coefficients of P. But this bound can be improved if we have more points x_i such that $P(x_i)$ is small. The same holds if P is a rational function, but this is harder to prove since cancellation may occur between the denominators and numerator during evaluation, so one need to prove that cancellations cannot occur too often. Using a careful study, Kieffer proves in [Kie20a] that in a number field K, P has coefficients of height $\widetilde{O}(H)$ (neglecting some factors) if it is a polynomial and there are $2d$ evaluation points of small heights (ie the heights of the $P(x_i)$ is H), or if it is a rational function and there are $O(d^3)$ evaluation points of small heights.

Applied to modular polynomials, this proves that the height of the coefficients is in $\widetilde{O}(D)$ where D is the degree of the modular correspondance. In fact Kieffer proves a general version in [Kie20a] for a modular correspondance on a Shimura variety of PEL type, and gives refinement (in particular more explicit constants) in the case $g = 2$.

Table 5.1: Siegel modular polynomials in dimension 2

ℓ	Size (Streng's Igusa invariants)	Theta constants
2	2.1 MB	
3	890 MB	270 KB
5		305 MB
7		29000 MB

In summary: in any dimension, the total degree of the coefficients c_i is bounded by $O(D)$ and their height by $\widetilde{O}(D)$ where D is the degree of the modular correspondance. So for abelian surfaces, $D = O(\ell^3)$ for ℓ-modular polynomials in the Siegel case, and $D = O(N(\beta))$ for β-modular polynomials in the Hilbert case. In dimension $g = 1$, we recover the classical fact that $\Phi_\ell(X, Y)$ has degree $O(\ell)$ in X, and Y, and its coefficients have height $\widetilde{O}(\ell)$, so its total size is $\widetilde{O}(\ell^3)$.

More generally, in dimension g, in the Siegel case the degree $D = \#\Gamma/\Gamma_0(\ell)$ is equal to $D = O(\ell^{g(g+1)/2})$ (an explicit version can be given using q-combinatorics but we won't need this). The modular polynomial $\Phi_{\ell,1}(X)$ will then have degree D in X, and its coefficients c_i are given by polynomials of total degree D in the $g(g+1)/2$ variables given by the primary invariants of the Siegel space (we can neglect the secondary invariant since their degrees above the primary invariants do not depend on ℓ). Letting $N = g(g+1)/2$ the dimension of $\mathcal{A}_g$, this shows that we have $O(DD^N)$ terms, each with a coefficient of height $\widetilde{O}(D)$, with $D = O(\ell^N)$, for a total size of $\widetilde{O}(D^{N+2}) = \widetilde{O}(\ell^{N(N+2)})$. This still holds for Hilbert modular polynomials over a real field K of dimension g, using that the dimension of H_g is $N = g$, and with $D = O(N(\beta)) = O(Nr(\beta)^N)$, where $Nr(\beta) = N(\beta)^{1/[K:\mathbb{Q}]}$: the total size is $\widetilde{O}(D^{N+2}) = \widetilde{O}(\ell^{g+2})$ if $\ell = N(\beta)$.

For instance: if $g = 2$, in the Siegel case $N = 3$ and $D = O(\ell^3)$, so the Siegel ℓ-modular polynomials are of total size $\widetilde{O}(\ell^{15})$, while in the Hilbert case if β is of norm ℓ, $N = 2$ and $D = O(\ell)$, so they are of size $\widetilde{O}(\ell^4)$.

Remark 5.3.5. The approach developed by Kieffer in [Kie20a] also solves a question raised by Labrande in [Lab16, p. 168]. In that Chapter, Labrande computes isogenies between elliptic curves over a number field by using complex embeddings and the fast evaluation of theta functions (in z) he developed in his PhD. He asks about the precision P he needs to work with in order to be able to recognize the coefficients of the isogeny and of the isogenous curve as elements in K. We can use the isogeny height theorem to estimate this P, this bound the Faltings height of the isogenous curve E'. Then we may also estimate the heights of the polynomials defining the isogeny directly or using [Kie20a]: a n-torsion point of E is sent via the isogeny to a n-torsion point of E', and we can bound their heights (since their canonical height is 0), so we can bound the height of the polynomials defining the isogeny. For elliptic curves we have precise relations between the standard heights and Falgings height (resp. the canonical height), so this can be made fully explicit.

Example 5.3.6. Examples of the size of modular polynomials for abelian surfaces are given in Tables 5.1 and 5.2. These polynomials were computed by Milio for his PhD thesis [Mil15b; Mil15a; MR20b]. Previously Dupont had computed in [Dup06] the Siegel modular polynomials for $\ell = 2$ using Spalleck's version of Igusa invariants; this gives much bigger modular polynomials than using Streng's version: 26.8MB (compressed).

Like in Example 5.3.4, for $\mathbb{Q}(\sqrt{2})$, $\beta = 5 + 2\sqrt{2} \mid 17$, letting b_1, b_2, b_3 be the pullback of level 2 theta functions θ_i/θ_0 on the Hilbert space, the denominator of $\Phi_{1,\beta}$ is $b_3^6 b_2^{18} + (6b_3^8 - 6b_3^4 + 1)b_2^{16} + (15b_3^{10} - 24b_3^6 + 7b_3^2)b_2^{14} + (20b_3^{12} - 42b_3^8 + 9b_3^4 + 2)b_2^{12} + (15b_3^{14} - 48b_3^{10} + 37b_3^6 + 4b_3^2)b_2^{10} + (6b_3^{16} - 42b_3^{12} + 68b_3^8 - 26b_3^4 + 3)b_2^8 + (b_3^{18} - 24b_3^{14} + 37b_3^{10} + 8b_3^6 - b_3^2)b_2^6 + (-6b_3^{16} + 9b_3^{12} - 26b_3^8 - 24b_3^4 + 2)b_2^4 + (7b_3^{14} + 4b_3^{10} - b_3^6)b_2^2 + (b_3^{16} + 2b_3^{12} + 3b_3^8 + 2b_3^4 + 1)$.

To illustrate the differences of heights between Hilbert and Siegel modular polynomials, one coefficient of the denominator of the Siegel modular polynomial $\Phi_{1,5}$ is 1180591620717411303424.

5.3.8 *Evaluating modular polynomials*

We recall the Schoof-Pila method to compute the number of points of an abelian surface over $\mathbb{F}_q$ (with $q = p^n$): compute the characteristic polynomial χ_π of the Frobenius π on the ℓ-torsion and then use the CRT to recover χ_π. Using the Weil bounds this requires $O(\log q)$ primes ℓ of size $O(\log q)$. Computing the action on the ℓ-torsion is in $\widetilde{O}(\ell^6)$ operations in $\mathbb{F}_q$, for a total cost of $\widetilde{O}(\log q^7)$ operations in $\mathbb{F}_q$ using [GS12], ie of $\widetilde{O}(\log q^8)$ (see Section 5.5).

Table 5.2: Hilbert modular polynomials in dimension 2

ℓ ($\mathbb{Q}(\sqrt{2})$)	Size (Gundlach)	Theta	ℓ ($\mathbb{Q}(\sqrt{5})$)	Size (Gundlach)	Theta
2	8.5 KB		5	22 KB	45 KB
7	172 KB		11	3.5 MB	308 KB
17	5.8 MB	221KB	19	33 MB	3.6 MB
23	21 MB		29	188 MB	21 MB
31	70 MB		31	248 MB	28 MB
41	225 MB	7.2 MB	41	785 MB	115 MB
73		81 MB	59	3600 MB	470 MB
89		188 MB			
97		269 MB			

It might seem hopeless to improve point counting using a SEA like algorithm by using isogenies to restrict to a subgroup of the ℓ-torsion since the ℓ-modular polynomial is already of size $\widetilde{O}(\ell^{15})$.

But things get much better if there is a way to directly evaluate the modular polynomials on $j_1(A), j_2(A), j_3(A)$ where $A/\mathbb{F}_q$ is an abelian surface. Indeed, $\Phi_{\ell,1}$ is then a polynomial in $\mathbb{F}_q[X]$ of degree $O(\ell^3)$, so if $\ell = O(\log q)$ is of size $\log^4 q$.

A complex analytic approach for elliptic curves and abelian surfaces

We do know how to evaluate the modular polynomials over $\mathbb{C}$, this is part of the evaluation-interpolation algorithm, and can be done in time quasi-linear in the size of the evaluation. This means that we know how to evaluate the modular polynomials over a number field K. Still, an important difference is that in the evaluation-interpolation approach we may choose our evaluation points, while here it will be given.

Concretely, we may embed K into $\mathbb{C}$ via an embedding ϕ, and if we know $\phi(x)$ with $x \in K$ with enough precision (depending on the height of x and constants depending on K like its discriminant) we can use the LLL algorithm to recover $x \in K$. While the LLL algorithm can be made quasi-linear in the precision, it is still cubic in the dimension n of K, so a better alternative approach is to consider all n embedding $\phi_i : K \to \mathbb{C}$, and reconstruct x from these embeddings. This requires to compute more values but on the other hand we can also recover x using less precision on the $\phi_i(x)$. We refer to [Kie20b, § 2.3] for more details.

So this suggests the following strategy to evaluate a modular polynomial over $\mathbb{F}_q$: find a number field K and a place $\mathfrak{p}$ such that $O_K/\mathfrak{p} \simeq \mathbb{F}_q$, lift A to O_K, evaluate the modular polynomials on this lift and then reduce them modulo $\mathfrak{p}$. If $q = p$ we may of course take $K = \mathbb{Q}$. The lifted j-invariants are of height $O(\log p)$, so for abelian surfaces the evaluated modular polynomial over $\mathbb{Q}$ is of degree $O(\ell^3)$ with coefficients of height $\widetilde{O}(\ell^3 \log p)$, so if $\ell = O(\log p)$ it is of total size $\widetilde{O}(\log p^7)$. In a SEA like algorithm this would be the dominating step, so using this method we will get a complexity $\widetilde{O}(\log p^8)$, see Section 5.5. Likewise, in the Hilbert case, if ℓ is the norm of β, the evaluated polynomial over $\mathbb{Q}$ is of degree $O(\ell)$ with coefficients of height $\widetilde{O}(\ell \log p)$ for a total size of $\widetilde{O}(\log p^3)$ if $\ell = O(\log p)$.

So I suggested to Kieffer to look at this "lifting" approach, which he did brilliantly in [Kie20b]. Indeed the practical details are far from trivial:

- To count points over $\mathbb{F}_q$ we need to recognize elements in a number field, this is harder than for $\mathbb{Q}$;
- As remarked above, unlike the general evaluation/interpolation method to compute the modular polynomials, we do not choose the evaluation point. Hence Dupont's algorithm which is quasi-linear in the precision for a fixed τ is not sufficient, one need to prove some uniformity condition. Furthermore, since the evaluation point is given, the values $\gamma \cdot \tau/\ell$ may fall far from the standard fundamental domain. Even the reduction algorithm to go back to the fundamental domain has to be considered carefully [Kie20b, § 4.2]. We refer to Section 5.3.4 for more details.
- Likewise, we need to show that if the height of the modular functions is bounded, we can bound appropriately and recover uniformly the period matrix τ at some precision [Kie20b, § 5].
- Furthermore Kieffer made the interesting observation that if we take an abelian surface or an hyperelliptic genus 2 curves given by "small" coefficients over $\mathbb{F}_p$, then we may lift the modular invariants to $\mathbb{Q}$ so that

they are of small $O(1)$ height, rather than $O(\log p)$, so the coefficients of the evaluated modular polynomial are only of height $\widetilde{O}(\ell^3)$. The evaluated polynomial is then of size $\widetilde{O}(\log p^6)$.

In the Hilbert case we would have coefficients of height $\widetilde{O}(\ell)$ and the evaluated polynomial is of size $\widetilde{O}(\ell^2)$.

With all this hard work, Kieffer obtains, under some heuristics about the convergence of the Newton process to evaluate theta constants via Dupont's algorithm:

Proposition 5.3.7 (Kieffer). *Given the Igusa invariants of an abelian surface over $\mathbb{F}_p$, the Siegel modular polynomial of level ℓ can be evaluated in time $\widetilde{O}(\ell^3 \log^2 p + \ell^6 \log p)$ in general, and in time $\widetilde{O}(\ell^6)$ if the surface is given by small integral coefficients.*

In the Hilbert case, if β is of norm ℓ, the Hilbert modular polynomial of level β may be evaluated in time $\widetilde{O}(\ell \log^2 p + \ell^2 \log p)$, and in time $\widetilde{O}(\ell^2)$ if the surface is given by small integral coefficients.

Remark 5.3.8. See also [Kie20b, § 6.3] for the case of evaluating modular polynomials over $\mathbb{F}_q$. Essentially if $q = p^d$ with d constant or sufficiently small (eg $d = O(\log p)$), the same complexity holds replacing $\log p$ by $\log q$: $\widetilde{O}(D \log^2 q + D^2 \log q)$ where D is the degree of the modular correspondance, ie the number of isogenies, which is $O(\ell^3)$ in the Siegel case and $O(\ell)$ in the Hilbert case. But there are extra complexity terms if d is large with respect to $\log p$. In the general case, if $\overline{P}$ is a polynomial defining $\mathbb{F}_q$, we lift it to a (unitary) polynomial $P \in \mathbb{Z}[X]$, so that $\mathbb{F}_q = \mathbb{Z}[X]/(p, P)$ and P is irreducible modulo p, and we let $K = \mathbb{Q}[X]/P(X)$. We can then bound the complexity by $\widetilde{O}(D \log^2 q + Dd^4M^2 + D^2 \log q + D^2d^2M)$ where $M = \max(1, h(P))$ is a bound on the height of the polynomial $P(X)$, see [Kie20b, Proposition 6.4].

We detail this (and refine a bit) because it will be useful for the other strategies. We denote by σ_j the d embeddings of K into $\overline{\mathbb{Q}}$, and let α be a root of P. If $x \in K$, we represent it by the polynomial $x = \sum \lambda_i \alpha^i$, $\lambda_i \in \mathbb{Q}$. We have $h(\{\sigma_j(x)\}) \leq h(\{\lambda_i\}) + (d-1)h(\{\sigma_j(\alpha)\}) + \log d \leq h(\{\lambda_i\}) + (d-1)h(P) + (d-1)\log 2 + \log d \leq h(\{\lambda_i\}) + O(dM)$. Conversely, we can interpolate the λ_i from the values of the $\sigma_j(x)$, and using Mahler's bound (or simply Hadamard's lemma) $h(\mathrm{Disc}_P) \leq (2d-2)h(P) + 2d \log d$ to invert the Vandermonde matrix, we get that $h(\{\lambda_i\}) \leq h(\{\sigma_j(x)\}) + (2d-2)h(P) + (2d+1)\log d \leq h(\{\sigma_j(x)\}) + \widetilde{O}(dM))$. If the λ_i are in $\mathbb{Z}$ (ie $x \in \mathbb{Z}[\alpha]$), we can improve the bound to $h(\{\lambda_i\}) \leq h(\{\sigma_j(x)\}) + (d-1)h(P) + (d+1)\log d$. We also recall the obvious bound $h(\{x_i\}) \leq \sum h(x_i)$, so $h(\{\sigma_j(x)\}) \leq dh(x)$. Via fast interpolation, the λ_i can be recovered from the complex values of the $\sigma_j(x)$, working at precision $m = \widetilde{\Omega}(h(\{\sigma_j(x)\}) + dM)$ in time $\widetilde{O}(dm)$, taking into account precision losses of at most $O(h(\{\sigma_j(x)\}) + dM + d\log d)$ [Kie20b, Lemma 2.4].

If α'_i form a basis of O_K, we may also represent an element $x \in K$ as $x = \sum \lambda'_i \alpha'_i$ rather than $x = \sum \lambda_i \alpha^i$. Then we can replace in the terms above $\log \mathrm{Disc}_P = O(dM + d\log d)$ by $\log|\mathrm{Disc}_K|$ if α'_i is a sufficiently nice basis, eg an LLL or BKZ or HKZ reduced basis (plus some extra terms depending on how well our basis is reduced). For instance, by [Cou20, § 2], there exists a basis α'_i such that $h(\sigma'_j(\alpha'_i)) \leq \frac{2}{d}\log|\mathrm{Disc}_K|$, so $h(\{\sigma_j(x)\}) \leq h(\{\lambda'_i\}) + \frac{2}{d}\log|\mathrm{Disc}_K| + \log d$ and $h(\{\lambda'_i\})) \leq h(\{\sigma_j(x)\}) + 4\log|\mathrm{Disc}_K| + \frac{2d+1}{2}\log d$ (or $h(\{\lambda'_i\})) \leq h(\{\sigma_j(x)\}) + 2\log|\mathrm{Disc}_K| + \frac{d+1}{2}\log d$ if $x \in O_K$, ie $\lambda'_i \in \mathbb{Z}$). See also [Kie20b, Lemma 2.6, § 6.3] for the case of an LLL reduced basis of O_K.

If $J_A = \sum a_i \alpha^i$, we let $H_0 = h(\{a_i\})$ be a bound on the heights of the $a_i \in \mathbb{Q}$. So $H := h(\{\sigma_i(J_A)\}) = O(H_0 + dM)$ and the evaluated modular polynomial on J_A then has coefficients c_j of heights $h(\{\sigma_i(c_j)\}) = \widetilde{O}(D(H_0 + dM))$. Here we assume that we work with the affine modular polynomials, otherwise when evaluating the rational functions (assuming the denominators do not vanish), using that $h(c_N) = h(c_N^{-1})$, we get a bound of $h(\{\sigma_i(c_j/c_N)\}) = \widetilde{O}(dD(H_0 + dM))$.

We recognize the coefficients as elements $c_j = \sum c_{j,i}\alpha^i$, and the above formula show that the $c_{j,i} \in \mathbb{Q}$ have heights $\widetilde{O}(D(H_0 + dM))$. They can be recovered by interpolation in time $\widetilde{O}(dD(H_0 + dM))$ from the values $\sigma_i(c_j)$ of each complex embedding. working at precision $m = \widetilde{\Omega}(D(H_0 + dM))$ (taking into account precision losses). The total complexity of the evaluation is then $\widetilde{O}(D(dD(H_0 + dM) + (d(H_0 + dM))^2))$. When $J_A \in \mathbb{F}_q$, we take for P a lift to $\mathbb{Z}$ of a polynomial defining $\mathbb{F}_q$, so $H_0 = O(\log p)$, and $H = O(\log p + dM)$ or $H = O(dM)$ with small parameters. This gives the complexity $\widetilde{O}(dD^2H_0 + d^2DH_0^2 + d^2D^2M + d^4DM^2)$ above.

The computations are quasi-linear in the height H, except for the reduction to the fundamental domain, which is essentially quasi-quadratic (hence the term in $(dH)^2$ above). A quasi-linear reduction algorithm (modeled on [NSV11; NS16]) would give an improved (quasi-linear) complexity of $\widetilde{O}(dD^2(H_0 + dM))$ over a number field, hence $\widetilde{O}(D^2(\log q + d^2M))$ over a finite field, or $\widetilde{O}(D^2d^2M)$ with small parameters.

Over a finite field, using the trivial bound $M = O(\log p)$, we get a bound of $\widetilde{O}(Dd^2 \log^2 q + D^2 d \log q)$, with a fully quasi-linear algorithm this would give $\widetilde{O}(D^2 d \log q)$ instead. But this trivial bound on M is too pessimistic,

by [AL86; Shp96] under GRH, we can deterministically find irreducible polynomials modulo p of degree d and height $M = O(2d \log\log p)$. Actually, since a random polynomial of degree d is irreducible modulo p with probability $\geq 1/2d$, in practice we can get $M = O(1)$ by doing $O(d)$ irreducibility tests over $\mathbb{F}_p$, for a total cost of $\widetilde{O}(d^2 \log^2 p) = \widetilde{O}(\log^2 q)$ by [KU11]. We obtain a complexity of $\widetilde{O}(D \log^2 q + Dd^4 + D^2 \log q + D^2 d^2)$, resp. $\widetilde{O}(D^2(\log q + d^2))$ with a quasi-linear reduction algorithm. This may involve changing the representation of $\mathbb{F}_q$, but by [BDD+19] an isomorphism can be computed in time $\widetilde{O}(d^{(1+\omega)/2} \log p + d \log^2 p)$ (this can be seen as a precomputation) and then applying the isomorphism on each coefficient uses a modular composition, hence costs $\widetilde{O}(d^{(1+\omega)/2} \log p)$, or even $\widetilde{O}(\log q)$ using [KU11].

The same strategy holds for evaluating a modular polynomial on an invariant defined in $\mathbb{Z}/p^m\mathbb{Z}$ or $\mathbb{Z}_q/p^m\mathbb{Z}_q$, this will be useful in Section 5.4.3 and Chapter 6. This time the lift to K has height $H = O(m \log p + dM)$, so it suffices to replace in the formulae above $\log p$ (resp. $\log q$) by $m \log p$ (resp. $m \log q$). We can still use an isomorphism to switch to a polynomial of small height, by lifting the isomorphism modulo p using Newton iterations; the fast modular composition of [KU11] holds over $\mathbb{Z}_q/p^m\mathbb{Z}_q$, see [KU11, § 4.3].

Finally the complexities given above hold for an arbitrary dimension g (adjusting D of course), provided that we have fast (uniform) evaluation of theta constants and fast evaluation of (one) period matrix, and also fast reduction to the fundamental domain to get the fully quasi-linear algorithm (over K). For the evaluation/period matrix, from the discussion of Section 5.3.4, the main difficulty is the fast evaluation of the period matrix, unless we are in the special case where A is a Jacobian of an hyperelliptic curve, in which case we can compute the periods at low precision.

Remark 5.3.9. For $g = 1$ we have all these quasi-linear algorithms, using [Dup06; Lab18] for evaluation/period matrices (when $g = 1$ there is no heuristics remaining), and the reduction to the fundamental domain is done using Gauss' algorithm which is linear [VV09, Theorem 6] in the size of the period matrix. Since $D = \ell$ for $g = 1$, we get a complexity of $\widetilde{O}(\ell^2(\log q + d^2 M))$ to evaluate modular polynomials over elliptic curves above a finite field $\mathbb{F}_q$, and we recall that we can take $M = O(1)$.

In [Sut13], for the evaluation of Φ_ℓ at $j \in \mathbb{F}_q$, Sutherland gives a complexity of $\widetilde{O}(\ell^3 + \ell^2 \log q)$ but with a better space complexity of $\widetilde{O}(\ell \log q + \ell^2)$, using the CRT method (with some clever tricks) to compute the full modular polynomial modulo small primes p_i, then evaluate it at $j \mod p_i$, then reconstruct the evaluation to $\mathbb{F}_q$. So the strategy developed above for $g = 2$ because the modular polynomials are too large is actually also useful for elliptic curves.

Remark 5.3.10. We can also tweak the algorithm to not only get the evaluation of the modular polynomials, but also of their derivatives in the same complexity, this will be helpful in Section 5.4. Indeed, to evaluate the derivative at $J_A \in \mathbb{F}_q$, it suffices to evaluate them at several values $J_A + p\nu$ in $\mathbb{Z}_q/p^2\mathbb{Z}_q$. Likewise to get the derivatives in $\mathbb{Z}_q/p^m\mathbb{Z}_q$ we evaluate $J_A + p^m\nu$ in $\mathbb{Z}_q/p^{2m}\mathbb{Z}_q$, taking m large enough allows to recover the derivatives at $J_A \in K$. See Section 5.4.3 for other strategies.

A CRT and p-adic lifting approach to evaluating modular polynomials in any dimension

As mentioned in Remark 5.3.8, the main stumbling point to generalize the complex analytic strategy to arbitrary dimension is the lack of an algorithm to evaluate the period matrix from the theta constants when $g > 2$ and A is not a Jacobian (but see Section 5.7 for a potential approach). Anticipating Section 5.6.2 and Chapter 6 we outline two different approaches based on p-adic lifting and CRT reconstruction. We use the same idea as above: we lift our abelian variety/its invariants from $\mathbb{F}_q$ to a number field K. This step was easy for abelian surfaces because we just need to lift the Igusa invariants, and more generally for Jacobians we just need to lift the curve, but for higher dimension we will have equations between our invariants, so we may need to take field extensions to get a rational moduli point, and this step may not at all be trivial. Fortunately this does not depend on the modular polynomials we want to evaluate.

The question then boils down to evaluating $\Phi_\ell(J(A), X)$ where $J(A) \in K$, in quasi-linear time in its size (which of course, as we saw in Remark 5.3.8 is not the same as the size of its reduction to $\mathbb{F}_q$; getting a quasi-linear algorithm for the evaluation of the reduced modular polynomial looks much more challenging). This is a system of dimension 0, which we can compute by evaluating all of its points. In this kind of situation there are often three strategies available: use complex approximation, p-adic approximation or use a a CRT approach. We will see this again in Chapter 7.

The strategy implemented in [Kie20b] is to look at the complex embeddings: evaluate the modular polynomial for each complex embedding and then reconstruct the coefficients in K (or O_K if we take an integral lifting and use

the version of the modular polynomials with explicit denominator from Section 5.3.6). The p_0-adic version use the p_0-adic embeddings instead (typically with p_0 much smaller than p), and the CRT algorithm reconstruct the coefficients $c_i \in K$ from their values modular several prime ideals.

For simplicity we assume that our modular invariants J used as coordinates on $\mathcal{A}_g$ are defined over $\mathbb{Z}$ (ie their Fourier coefficients are integral), so they are well defined modulo p. We can also work with invariants defined over $\mathbb{Q}$, in this case we need to sieve out the primes of bad reduction in the CRT or p_0-lifting approach.

THE SIEGEL CASE. Let us focus on one place, and for simplicity assume for now that $K = \mathbb{Q}$. Here p_0 is a small prime of $\mathbb{Q}$, not the (potentially large) prime p where $A/\mathbb{F}_p$ is defined. Then we can compute $J(A/\mathbb{Q}) \bmod p_0$, ie reduce A to $\mathbb{F}_{p_0}$ (assuming p_0 is of good reduction of course), compute $A[\ell]$ using Section 5.6.2, then all the possible kernels to form $\Phi_\ell(J(A), X) \mod p_0$. This will typically require working over an extension of p_0 of degree $d = O(d_0 \ell^g)$ where $\mathbb{F}_{p_0^{d_0}}$ is the field of definition of the isogenies and $\mathbb{F}_{p^d}$ of the points in their kernels. We may also bound d by $d = O(\ell^{2g})$, the field of definition of the points of $A[\ell]$.

So we first have a precomputation step to compute a symplectic basis of $A[\ell]$ which involves point counting (more precisely determining χ_π) over $\mathbb{F}_{p_0}$ and then sampling points, which cost of $O(d^2 \log^2 p_0)$.

We then compute all the $O(\ell^N)$ isogenies, where $N = g(g+1)/2$. If the kernel is rational, its points are defined in an extension of degree at most $O(\ell^g)$ so the isogeny cost at most $O(\ell^{2g})$. If the kernel is defined over $\mathbb{F}_{p_0^{d_0}}$, we work over a bigger extension but the Galois action gives us d_0-isogenies, so we have the same average cost of $O(\ell^{2g})$. So the total cost for all isogenies is at most $O(\ell^{N+2g} \log p_0)$. The final cost, neglecting logarithmic factors, in particular algorithms polynomial in $\log p_0$ such as point counting of $A/\mathbb{F}_{p_0}$, is then $\widetilde{O}(\ell^{4g} + \ell^{N+2g})$ binary operations. This is $\widetilde{O}(\ell^{N+2g})$ if $g > 1$. If we Sieve p_0 so that all points in $(A/\mathbb{F}_{p_0})[\ell]$ are rational (or more generally such that the points of the kernels are rational over the fields of definition of the kernels), this becomes $\widetilde{O}(\ell^{N+g})$ (including $g = 1$).

If A/K has invariants of height H, we recall from Section 5.3.7 that the height of the evaluated modular polynomial is $\widetilde{O}(\ell^N + H\ell^N) = \widetilde{O}(H\ell^N)$, and its degree $O(\ell^N)$. So given $A/\mathbb{Q}$, we need to look at its reduction modulo p_0^m where $m = O(\log(H\ell^N)/\log p_0)$. We lift the basis of $A/\mathbb{F}_{p_0}[\ell]$ to $\mathbb{Z}/p_0^m$ via Newton iterations (since $A[\ell]$ is étale over $\mathbb{F}_{p_0}$ the lifting behaves well). We then compute ℓ^N isogenies over $\mathbb{Z}/p_0^m$, each isogeny costing ℓ^g operations in $\mathbb{Z}/p_0^m$, so the total cost is then $\widetilde{O}(H\ell^{2N+g})$.

A similar strategy works for the CRT algorithm, with the added bonus that using explicit CRT [BS07] we can do the CRT reconstruction modulo p directly, so this only requires a memory the size of the evaluated modular polynomial modulo p. We need $\widetilde{O}(H\ell^N)$ primes, and if we sieve for primes p_0 such that $(A/\mathbb{F}_{p_0})[\ell]$ is composed of rational points, we expect roughly one out of $O(\ell^{2g})$ prime to work. So the largest prime would be $p_0 = O(H\ell^N\ell^{2g})$, but the dependency of the algorithm on p_0 is only polynomial in $\log p_0$, so this is taken care of by the $\widetilde{O}$ notation. The advantage is that the isogeny computations are done over the base field, so we also get an algorithm in $\widetilde{O}(H\ell^{2N+g})$.

THE HILBERT CASE. A similar method works for cyclic β-modular polynomials (provided we can explicit real multiplication, see the end of Section 5.6.2). If β is of norm ℓ, we have $O(\ell)$ isogenies each costing in average at most $O(\ell)$ to compute over an extension of $\mathbb{F}_{p_0}$ of degree $O(\ell)$, ie this costs $O(\ell^3)$. Furthermore, the initialisation step computes a basis of $A[\beta]$ in $O(d^2 \log^2 p_0)$ operations over $\mathbb{F}_{p_0}$ if $d = O(\ell^2)$ is the degree of definition of the geometric points of $A[\beta]$. The total cost is $\widetilde{O}(\ell^4)$.

So like in the Siegel case, we want to Sieve for p_0 such that the points of the kernels are rational over the fields of definition of the kernels, this is the case if $A[\beta]$ has rational points. In this case each isogeny costs (on average) $O(\ell)$, for a total cost of $\widetilde{O}(\ell^2)$ operations over $\mathbb{F}_{p_0}$. The height of the evaluated modular polynomial is $\widetilde{O}(H\ell)$ and its degree is $O(\ell)$. The p_0-adic approach and the CRT approach then both have a complexity of $\widetilde{O}(H\ell^3)$.

As remarked above, we can do a more intelligent sieving than searching for all points of $A[\beta]$ to be rational. $A[\beta]$ is an $\mathbb{F}_\ell = O_A/\beta$-module of rank 2, where $O_A = \mathrm{End}^s(A)$ are the real endomorphisms, furthermore the characteristic polynomial of the Frobenius is $\chi_\pi = X^2 - tX + q$ modulo β, for $t \in \mathbb{Z}[\pi + \overline{\pi}]$. We assume that $\mathbb{Z}[\pi + \overline{\pi}]$ is locally maximal at ℓ, we let ℓ_0 be the characteristic of $\mathbb{F}_\ell$, e_0 be the embedding degree (ie the order of q in $\mathbb{F}_\ell$), e_1 the minimal extension where all kernels become rational, and e_2 the minimal extension where all points of $A[\beta]$ become rational. Then it is easy to check that either $e_2 = e_0 \wedge e_1$ or $e_2 = 2(e_0 \wedge e_1)$.

There are three possibilities: if β is split or inert in $\mathbb{Z}[\pi]$, then $\pi_{|A[\beta]} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$, with $\lambda_i \in \mathbb{F}_\ell$ in the split case, and $\lambda_i \in \mathbb{F}_{\ell^2}$ with $\lambda_2 = \lambda_1^\ell$ in the inert case. Then e_1 is the order of λ_1/λ_2, so $e_1 \mid \ell - 1$ in the split

case, and $e_1 \mid \ell+1$ in the inert case. Finally, if β is ramified, either $\pi_{|A[\beta]} = \begin{pmatrix}\lambda & 0\\ 0 & \lambda\end{pmatrix}$ in which case $e_1 = 1$, or $\pi_{|A[\beta]} = \begin{pmatrix}\lambda & 1\\ \lambda & 0\end{pmatrix}$ (this is the case at the bottom of the volcano) in which case $e_1 = \ell_0$. So there is a non negligeable probability that $e_2 = e_1$, essentially with heuristic probability $1/2$ we are in the split case, and $e_1 = e_2$ whenever λ_1/λ_2 is primitive. Since the sieving is much less agressive, this gains a polylogarithmic factor in ℓ.

COMPLEXITY. In summary, if we fix the number field K so that parameters depending on K are hidden in the $O()$ notation:

Lemma 5.3.11. *Assume that J_A is of height H in a number field K. Then when evaluating the Siegel modular polynomials $\Phi_\ell(J_A, X)$, it has $O(\ell^N)$ coefficients of heights $\widetilde{O}(H\ell^N)$ for a total size of $\widetilde{O}(H\ell^{2N})$ with $N = g(g+1)/2$ (we hide in the O notation constants depending on the number field, like its degree, discriminant, etc). The p_0-adic and the CRT method computes the evaluation in time $\widetilde{O}(H\ell^{2N+g})$. The Hilbert modular polynomial $\Phi_\beta(J_A, X)$ has $O(\ell)$ coefficients of heights $\widetilde{O}(H\ell)$ for a total size $\widetilde{O}(H\ell^2)$, the p_0-adic and CRT method costs $\widetilde{O}(H\ell^3)$.*

If we then reduce the evaluated modular polynomial to $\mathbb{F}_q$, via a fast modulo p reduction this is quasi-linear in the size of the evaluated modular polynomial in K. As a corollary, evaluating the modular polynomial over $\mathbb{F}_q$, with $q = p^d$ and d fixed, costs $\widetilde{O}(\ell^{2N+g}\log q)$ in the Siegel case, and $\widetilde{O}(\ell^3 \log q)$ in the Hilbert case. We can also evaluate the derivative of the modular polynomials as in Remark 5.3.10.

Remark 5.3.12. Note that, using the same analysis as Remark 5.3.8, we may bound the constants hidden in the $\widetilde{O}()$ using the degree of the number field and the height of the coefficients of the polynomial we use to represent it. Using the notations of this Remark, we represent K as $K = \mathbb{Q}[\alpha]/P(\alpha)$ and we note $M = \max(1, h(P))$. If $J_A \in K = \sum a_i \alpha^i$ has coefficients $a_i$of height H_0, we need to recognize coefficients c_j of height $\widetilde{O}(D(H_0 + dM))$, and the evaluated modular polynomial is of size $\widetilde{O}(dD^2(H_0 + dM))$. The cost of the evaluation using the p_0-adic or CRT method is then $\widetilde{O}(dD^2E(H_0 + dM))$, where E is the cost for computing one isogeny, ie $E = \ell^g$ in the Siegel case and $E = \ell$ in the Hilbert case. Lifting a modular invariant defined over $\mathbb{F}_q$, $H_0 = O(\log p)$, so the height is $H = \widetilde{O}(\log p + dM)$, or $\widetilde{O}(dM)$ with small parameters. Likewise, if we lift modular invariants defined over $\mathbb{Z}_q/p^m\mathbb{Z}_q$, the height is $H = \widetilde{O}(m\log p + dM)$, and the total cost is $\widetilde{O}(D^2E(m\log q + d^2M))$.

For instance, for the evaluation of the Hilbert modular polynomial over $\mathbb{F}_q$, we have $D = E = O(\ell)$, and using $M = O(1)$, so the evaluated polynomial has $O(\ell)$ coefficients of heights $\widetilde{O}(\ell(\log p + d))$, and the evaluation costs $\widetilde{O}(\ell^3(\log q + d^2))$.

We can compare this with the full modular polynomial, which has a size of $\widetilde{O}(D^{N+2})$, and evaluating it in K costs $\widetilde{O}(d(H_0 + dM)D^{N+2})$, while evaluating it directly in $\mathbb{F}_q$ costs $\widetilde{O}(\log qD^{N+1})$. In particular, the full Siegel modular polynomial has a size of $\widetilde{O}(\ell^{N(N+2)})$, and evaluating it in K costs $\widetilde{O}(d(H + dM)\ell^{N(N+2)})$, while evaluating it directly in $\mathbb{F}_q$ costs $\widetilde{O}(\ell^{N(N+1)}\log q)$. The full Hilbert modular polynomial has a size of $\widetilde{O}(\ell^{g+2})$, and evaluating it in K costs $\widetilde{O}(d(H + dM)\ell^{g+2})$, while evaluating it directly in $\mathbb{F}_q$ costs $\widetilde{O}(\ell^{g+1}\log q)$.

We need to be a bit careful when working with a number field. In the p_0-adic method, if we work with only one place $\mathfrak{P}_0$ above p_0, then we need to uses LLL to reconstruct the coefficients as elements of K, which requires to increase the precision and is in $\widetilde{O}(d^4)$ in terms of d using [NS16]. So we fix p_0 unramified in $K = \mathbb{Q}(\alpha)$, and compute all $\overline{\mathbb{Q}}_{p_0}$ Galois conjugate of the coefficients c_j, reconstructing the c_j is then a simple matter of interpolation. More precisely, with our method we get the value of c_j in $\mathbb{Z}[\alpha]/\mathfrak{P}_0^m$ for each $\mathfrak{P}_0$ above p_0, which gives the value of c_j in $\mathbb{Z}[\alpha]/p_0^m$ by a CRT reconstruction. Then we do a rational reconstruction[1] (over $\mathbb{Q}$) to recognize the $c_j \in \mathbb{Q}(\alpha)$. Since we do some Sieving on p_0, we don't want p_0 to split completely in K otherwise our probability of success decrease too much. The ideal case is when p_0 stays inert; for instance if $K/\mathbb{Q}$ to be cyclic, then by Cebotarev's density theorem this case occurs with probability $\phi(d)/d$. Otherwise we need to assume that there are sufficiently many unramified primes that do not split too much. When evaluating over $\mathbb{F}_q$, we can also change the polynomial $P(X)$ defining K rather than changing p_0, ie we may construct a nice list of number fields K as a precomputation step.

The same strategy holds for the CRT approach, either we reconstruct the c_j modulo an ideal lattice I, and this also involves an LLL computation (of dimension $d+1$ or d if the c_j are in $\mathbb{Z}[\alpha]$) to recognize the coefficients as elements of $\mathbb{Q}(\alpha)$, or more efficiently we systematically use all primes above each of the p_i we use in the CRT, to

[1] This rational reconstruction is not necessary if we evaluate at modular invariants in $\mathbb{Z}[\alpha]$, eg when lifting from a finite field, and we use the description of the denominators as modular forms from Section 5.3.6, and we use the modular version of the isogeny algorithm from Section 4.6 to compute these denominators.

get the c_j as elements of $\mathbb{Z}[\alpha]/N$, where $N = \prod p_i$. Likewise, we need to use p_i that don't split too much, and we can also combine both approaches to get a modulus $N = \prod p_i^{m_i}$.

Remark 5.3.13 (LLL reconstruction of elements in a number field)**.** As alluded to the previous Remark, when reconstructing elements in a number field, it is better to work with each embedding. But if we only have one embedding, we can still reconstruct the element x using LLL. It is not the purpose of this document to give a course on lattice, so we will only detail this very briefly, because the same kind of situation will appear in Chapter 7 where we use all Galois conjugate rather than just one to speed up the computation of the class polynomial.

There are three related problems. First, given an algebraic number α, we want to reconstruct its minimal polynomial P_α. Given each (complex or p-adic) embedding $\sigma_i(\alpha)$, we simply use a product tree to reconstruct P_α in quasi-linear time, working at precision $h(P_\alpha)$, for a total cost of $\widetilde{O}(dh(P_\alpha))$ where $d = \deg \alpha$; we recall that $dh(\alpha) - d\log 2 \leq h(P_\alpha) \leq dh(\alpha) + d\log 2$. Given only one embedding, we need to work at precision $\beta = O(d^2 + dh(P_\alpha))$ and use LLL in dimension d, see [Sch84; KLL88] for more details. Using [NS16], this costs $\widetilde{O}(d^4\beta)$.

Another problem is to reconstruct $x \in K$ given its value modulo a fractional ideal I. When $I = mO_K$ with $m = O(h(\sigma_j(x))$, the reconstruction is easy (eventually doing rational reconstruction of coefficients if $x \notin O_K$). For a general I, using [Belo4, Lemma 3.12] one can also use a LLL-reduced basis and reconstruct x provided $N(I) = \Omega(d^2 + dh(\sigma_j(x)))$ (using in the notations of [Belo4] that $\log T_2(x) = O(h(\sigma_j(x)) + \log d)$.

The last problem, is given a basis b_i of $K = \mathbb{Q}(\alpha)$, and the value of the embeddings of x along with those of the b_i, to reconstruct $x = \sum \lambda_i b_i$. When we have all embeddings this is a simple matrix inversion at precision $h(\lambda_i)$, and when $b_i = \alpha^i$ this matrix inversion can be computed in quasi-linear time by interpolation. With only one embedding, by [HPS] we need to work at precision $\beta = O(d^2 + dh(P_\alpha) + dh(\lambda_{ji}, \lambda_i))$, where $b_j = \sum \lambda_{ji}\alpha^i$.

So we don't quite get a quasi-linear algorithm over a number field K, but I conjecture one exists (we have seen this is the case if $g = 1$). With d fixed (or small enough), this would give an $\widetilde{O}(\ell^{2N}\log q)$ evaluation in the Siegel case, and $\widetilde{O}(\ell^2 \log q)$ in the Hilbert case.

Conjecture 5.3.14. *Using the notations of Remark 5.3.12, there is an evaluation algorithm over a number field K in time $\widetilde{O}(dD(DH + dM)) = \widetilde{O}(D^2(dH_0 + d^2M)$, hence an evaluation algorithm over $\mathbb{Z}_q/p^m\mathbb{Z}_q$ in time $\widetilde{O}(D^2(m\log q + d^2M))$.*

Strategies. We present three potential strategies. The first strategy uses complex evaluation, the required algorithms are detailed in Remark 5.3.8. To get a quasi-linear algorithm in the p_0-adic or CRT method, we need to evaluate our D isogenies in average $\widetilde{O}(1)$ time. This ultimately boils down to evaluate multivariate polynomials on coordinates of generators of the kernel.The second strategy would then be to see if we can use the quasi-optimal multivariate evaluation algorithm from [KU11, § 4] in the settings of isogenies.

The third strategy adapts the p_0-adic method to uses canonical lifting. Indeed, suppose that we need to evaluate the modular polynomials on $J_A \in \mathbb{Z}_q/p_0^m\mathbb{Z}_q$ (where m gives enough precision to reconstruct the evaluated modular polynomials in K). Then if J_A is the reduction modulo p_0^m of the canonical lift $\tilde{J}_A$ of $J_A \mod p_0$ (assuming this is an ordinary point), then it suffices to compute all isogenies B_i modulo p_0, and then their canonical lifts $\widetilde{B}_i \mod p_0^m$. By the theory of canonical lifts, these are exactly the varieties isogenous to $J_A \mod p_0^m$. The computation of canonical lifts uses the modular polynomial Φ_{p_0} (but remember that p_0 is small), and their computation is quasi-linear in the precision. Computing the isogenies modulo p_0 costs $\widetilde{O}(DE) = \widetilde{O}(D^2)$ operations in $\mathbb{F}_{p_0}$, so in this case we have a quasi-linear algorithm. But of course, there is no reason that $J_A \mod p_0^m = \tilde{J}_A \mod p_0^m$. But since they both reduce modulo p_0 to the same abelian variety, we can describe J_A using the Serre-Tate canonical coordinates of the local moduli [Kat81]. Furthermore the modular correspondance is easy to describe in these local coordinates (see [CN90, § 3.4]), so then we would need to convert back the local coordinates of the $B_i \mod p_0^m$ isogenous to $A \mod p_0^m$ to modular invariants $J_{B_i} \mod p_0^m$, using our computed values of $\tilde{J}_{B_i} \mod p_0^m$. This strategy requires back and forth transformation between modular invariants and local coordinates (given the modular invariants of the canonical lift), form [Kat81, Main theorem 3.7.1], this essentially requires an explicit version of the Kodaira-Spencer isomorphism and a description of the "physical" Tate module $T_{p_0}A$ with its Weil pairing at precision p_0^m. The question boils down to whether this conversion can be computed efficiently, in particular if the description of $T_{p_0}A$ at high enough precision does not require taking extensions of too large degrees. □

We obtain:

Corollary 5.3.15. *There is an algorithm in $\widetilde{O}(D^{N+2}E)$ evaluation-interpolation strategy to compute the Siegel and Hilbert modular polynomials. Under Conjecture 5.3.14, this becomes a quasi-linear $\widetilde{O}(D^{N+2})$ algorithm.*

Overview. Indeed we may work with a birational model of $\mathcal{A}_g$, using (at most) $N+1$ invariants, with N primary invariants, where N is the dimension. The defining equation of the birational model then gives a bound on the degrees and heights of the defining polynomials of our number fields (if we evaluate the primary invariants to small integers), so we get a uniform bound on the evaluation step. By Hilbert's irreducibility theorem the specialisations will generically give a number field. But in fact working with an étale algebra is not a problem, using the usual trick that when an inversion fails, we can use it to factor the polynomial.

Note that as a bonus of the way we do the evaluation step, we easily get all the Galois conjugates, ie (generically) the values of the evaluations for all the possible secondary invariants. Then we can do a fast polynomial interpolation as in Section 5.3.5 if we compute "integral modular polynomials", otherwise a rational fraction reconstruction. □

All in all these look like promising methods, and I plan to study them further: see if implementations can be made practical.

5.4 Applications of modular polynomials to isogenies between abelian varieties

An interesting algorithmic problem about isogenies is, provided that we have an oracle giving two ℓ-isogenous abelian varieties A and B, to find the corresponding isogeny $f : A \to B$.

Of course in practice this oracle may be given by modular polynomials (or more generally a suitable modular correspondance). It may seem that we lack sufficient informations to recover f (for starters there may be several isogenies between A and B). But we have seen in Section 4.7 that if we have the action of f on explicit differentials w_A and w_B of A and B, we can recover the isogeny (efficiently) by solving a differential system. Alternatively we require the tangent map M of f at 0.

From this point of view, in large characteristic (or $p = 0$) we don't even need to know ℓ, only a bound N on it (so we solve the differential system along some uniformisers, and the bound gives us the precision we need before doing the rational reconstruction). Indeed, the other possible isogenies with the same tangent map are of the form $f + g$ with g purely inseparable, so of larger degree than f for large p. More precisely, by [KPR20, Lemma 5.1], if $p > 4N$ (or $p = 0$), f is the only ℓ-isogeny with $\ell \leq N$ and tangent map M. (This is the same argument than at the beginning of Section 4.7.)

This raises the question of how we can find this matrix M.

5.4.1 *Elkies' method for elliptic curves*

Once again modular polynomials provide a solution. For elliptic curve this was Elkies' insight. Writing $\Phi_\ell(j(\tau), j(\tau/\ell)) = 0$, we get by differentiating:

$$dj(\tau)\frac{\partial \Phi_\ell(j(\tau),j(\tau/\ell))}{\partial X} + \frac{1}{\ell} dj(\tau/\ell)\frac{\partial \Phi_\ell(j(\tau),j(\tau/\ell))}{\partial Y = 0}. \tag{5.9}$$

But $dj(\tau)$ is a modular form of weight 2, so we can use it to get informations about differentials. Algebraically we proceed as follow: let E_1 and E_2 be ℓ-isogenous. Write $E_1 : y^2 = x^3 + a_1 x + b_1$, the canonical differential associated is $w_{E_1} = dx/y$ (here we assume $p > 3$, it is straightforward to adjust for $p = 2, 3$). As a modular form $/d\tau(\tau)$ (which I will write $dj(\tau)$ to have notations coherent with higher dimension) is defined over $\mathbb{Z}$ so the value $dj(E_1, w_{E_1})$ is well defined. Indeed, $dj(\tau)/j(\tau) = -E_6(\tau)/E_4(\tau)$ (where $E_4(\tau), E_6(\tau)$ are the Eisenstein series), so algebraically $dj(E_1, w_{E_1}) = cj(E_1)b_1/a_1$ for some constant c, since these Eisenstein series give the coefficients a and b of an elliptic curve (up to some constants). This constant c is easily determined ($c = 864/48 = 18$ by [Sch95, § 7]) but we won't actually need it!

Plugging $j(\tau) = j(E_1)$, $j(\tau/\ell) = j(E_2)$ and $dj(\tau) = dj(E_1, w_{E_1})$ in Equation (5.9) we recover the value of $dj(\tau/\ell) = dj(E_2, w_{E_2})$ provided that $\partial\Phi_\ell(j(E_1), j(E_2))/\partial Y \neq 0$. But $j(\tau/\ell)$ corresponds to the elliptic curve $E_2 : \mathbb{C}/(\mathbb{Z} \oplus \tau/\ell\mathbb{Z})$ such that the isogeny $f : E_1 \to E_2$ is simply $z \mapsto z$. In other words the isogeny is normalised, ie $f^* w_{E_2} = w_{E_1}$. So Equation (5.9) allows to recover $dj(E_2, w_{E_2})$ for w_{E_2} given by a normalised isogeny.

It is then easy to find an equation of $E_2 : y^2 = x^3 + a_2 x + b_2$ such that w_{E_2} is given dx/y: simply take an arbitrary equation $E_2 : y^2 = x^3 + a_2' x + b_2'$, take $w'_{E_2} = dx/y$ and compute $dj(E_2, w'_{E_2}) = cj(E_2)b_2'/a_2'$, and compare it to $dj(E_2, w_{E_2})$ (here the constant c is compensated from the once from $dj(E_2, w_{E_2})$).

Let us write $dj(E_2, w_{E_2}) = m^2 dj(E_2, w'_{E_2})$. Then since dj is of weight 2, m is exactly the value such that $f^* w'_{E_2} = m w_{E_1}$ (at this level we cannot distinguish between f and $-f$ so there is always a sign ambiguity). It then suffices to solve a differential equation. It is also easy to get the promised equation for E_2: an isomorphism of short

Weierstrass equations is given by $(x,y) \mapsto (u^2x, u^3y)$ and this acts on the canonical differential dx/y by u. So from E_2' we get that $a_2 = a_2'm^4$ and $b_2 = b_2'm^6$.

It is instructive to look at what happens if we differentiate the equation $\Phi_\ell(j(\tau), j(\ell\tau))$ instead. We find a value $dj(E_2, w''_{E_2})$ such that $dj(E_2, w_{E_2}) = \ell^2 dj(E_2, w''_{E_2})$. But analytically, $j(\ell\tau)$ correspond to the curve $\mathbb{C}/(\mathbb{Z} \oplus \ell\tau\mathbb{Z})$, and this time the isogeny is given by $z \mapsto \ell z$. So $f^*w''_{E_2} = \ell w_{E_1} = \ell f^* w_{E_2}$, so $w''_{E_2} = \ell w_{E_2}$ and this is indeed coherent with the fact that dj is of weight 2.

Summary 5.4.1. In summary, for a separable ℓ-isogeny $f : E_1 \to E_2$ (assuming E_1, E_2 have no extra automorphisms for simplicity) defined over k, the following datum are equivalent (up to a sign):

1. Differentials w_1 on E_1 and w_2 on E_2 and the relation $f^*w_2 = \lambda w_1$;
2. Equations $y_i^2 = x_i^3 + a_i x_i + b_i$ on E_i and the relation $f^*dx_2/y_2 = \lambda dx_1/y_1$;
3. The equation of the tangent space $dj_2 = \frac{\lambda^2}{\ell} dj_1$ of the ℓ-modular curve $X_0(\ell)$ at j_1, j_2, where $j_i = j(E_i)$;
4. The derivative $\partial\Phi_\ell/\partial X$ and $\partial\Phi_\ell/\partial Y$ at j_1, j_2;
5. Given the j-invariant $j_{1,\epsilon} = j(E_{1,\epsilon})$ of a (non-trivial) deformation of E_1 to $k[\epsilon]$, the value of $j_{2,\epsilon} = j(E_{2,\epsilon})$ of the unique deformation of E_2 to $k[\epsilon]$ lifting f to $E_{1,\epsilon}$.

Indeed, given $w = dx/y$ on $E : y^2 = x^3 + ax + b$, the corresponding deformation to $k[\epsilon]$ is given by $j(E_\epsilon) = j(E) + \epsilon dj(E, w) = j(E) + \epsilon cb/a$.

From the algorithmic point of view, this means that the derivatives of the modular polynomials allows to compute the normalised isogenous elliptic curve, conversely since Vélu's formula gives normalised isogenies, they can be used to compute the derivatives.

We will see in Section 5.4.2 that these equivalences hold in all dimension. See also [KPR20, § 4.5] for more details on this equivalence for Jacobians of hyperelliptic curves of genus 2.

Remark 5.4.2. Giving the j-invariant $j(E)$ of an elliptic curve only allows to recover E up to a twist. Assuming $p > 3$ for simplicity, the coefficients a and b of a short Weierstrass equation $E : y^2 = x^3 + ax + b$, which are modular forms of weight 4 and 6 respectively, of course completely determine E. However, if we are already given $j(E)$, only giving a or b is not enough to completely recover the equation of E. The solution is to give instead the value $f(E)$ of a modular function of weight 2, and the modular function $dj(E)$ is one such "canonical" example. Indeed from the above equations, it is immediate to recover a and b from E and $dj(E)$, so $dj(E)$ can be seen as a convenient way to distinguish between the twists of E. It has a pole at the curve corresponding to $j = 0$.

RECOVERING ISOGENIES. For alternative strategies to compute an explicit ℓ-isogeny $E \to E_1$ between two given isogenous elliptic curves (without knowledge of the kernel), via interpolation between points of p-torsion or r-torsion, we refer to [Cou94; Cou96; Fee10; DHP+16]. In particular the interpolation method of [DHP+16, § 5.3] has a complexity of $\widetilde{O}(\ell^2 \log^6 q)$ to compute the isogeny between two ℓ-isogenous elliptic curves over $\mathbb{F}_q$.

We remark that Elkies method to recover the isogeny via its differential equation as in Section 4.7.1 is dominated by evaluating the modular polynomial, and using the analytic method to evaluate Φ_ℓ when $g = 1$ (see Remark 5.3.9) yields a better complexity of $\widetilde{O}(\ell^2(\log q + d^2))$.

Since the isogeny can be represented by its kernel K of size $O(\ell \log q)$, we get a uniform quasi-quadratic algorithm (uniform, because contrary to [DHP+16] it is also quadratic in $\log q$). In fact the dependency in $\log q$ is actually quasi-linear whenever $d = O(\log p)$.

It is a challenging problem to get a quasi-linear dependency on ℓ. However, if we are given the $\ell + 1$ isogenous elliptic curves E_i [2] (assuming all isogenies are rational), then once we have computed $\Phi_\ell(j_E, Y)$ and $\partial\Phi_\ell/\partial_X(j_E, Y)$ in time $\widetilde{O}(\ell^2(\log q + d^2))$, recovering each isogeny $E \to E_i$ takes time $\widetilde{O}(\ell \log q)$ by Proposition 4.7.5. So we get a quasi-linear complexity (if d is small) by isogeny on average. In other words the factor ℓ^2 is because computing the evaluated modular polynomials encode all the isogenies.

An alternative strategy when q is small and E is ordinary is to compute $\mathrm{End}(E)$ and then if ℓ is an Elkies prime for $\mathrm{End}(E)$ (in particular it does not divide the conductor of $\mathrm{End}(E)$), we can use the class group to decompose the isogeny as product of smaller isogenies and get an algorithm quasi-linear in ℓ, see eg [CEL20, Theorem 19].

We argued in Section 4.7.3 that since the isogenies can be recovered in quasi-linear time from equations of E and the E_i along with their actions on the canonical differentials, or equivalently given the $j(E)$, $dj(E, w_E)$, $j(E_i)$,

[2] They can be recovered as roots of $\Phi_\ell(j_E, Y)$ in $\widetilde{O}(\ell \log q)$ operations in $\mathbb{F}_q$, ie $\widetilde{O}(\ell \log^2 q)$ binary operations

$dj(E_i, w_{E_i})$ where w_{E_i} is normalised with respect to w_E, then we can encode each of them by $O(1)$ elements in $\mathbb{F}_q$ (if ℓ is small compared to p). But to give this data for each E_i is essentially equivalent to give $\Phi_\ell(j_E, Y)$ and $\partial\Phi_\ell/\partial_X(j_E, Y)$. Indeed one direction is given by computing the product $\prod(Y - j_{E_i})$ and an interpolation, which is quasi-linear, and the other direction is given by computing the rational roots (which is in $\widetilde{O}(\ell \log^2 q)$), and multipoint evaluation (which is quasi-linear, ie in $\widetilde{O}(\ell \log q)$). From this point of view of compressed isogeny representations, we are not quasi-linear on the size of the output on average, because we only have a quasi-linear algorithm for the evaluation of modular polynomials over a number field K, not over $\mathbb{F}_q$.

In summary, we can complete Proposition 4.7.5 as follow:

Proposition 5.4.3. *Given $E/\mathbb{F}_q$, we can evaluate the modular polynomials $\Phi_\ell(j(E), Y)$ and $\partial\Phi_\ell/\partial_X(j(E), Y)$ at p-adic precision $m = O(\log_p \ell)$ in time $\widetilde{O}(\ell^2(m \log q + d^2))$. Once evaluated, the rational roots can be computed $\widetilde{O}(\ell \log^2 q)$ and then the roots $j(E_i)$ can be lifted to p-adic precision m and then the derivatives $dj(E_i)$ evaluated in quasi-linear time $\widetilde{O}(\ell m \log q)$ using multipoint evaluation. This give the compressed isogeny representation of Proposition 4.7.5.*

5.4.2 Adapting Elkies' method in higher dimension

THE SIEGEL CASE. With this reformulation of Elkie's algorithm, it is clear how to extend this to higher dimension. Letting J be a system of modular invariants on the modular space $\mathcal{A}_g$, we differentiate the modular equation $\Phi_\ell(J(\tau), J(\tau/\ell))$ to find a matrix relationship between $dJ(\tau)$ and $dJ(\tau/\ell)$, encoding the normalised isogeny $z \mapsto z$, provided that $\partial\Phi_\ell/\partial_Y$ is inversible at these points. (In other words we fix a basis of the tangent spaces such that $df(0) = \mathrm{Id}$). But since J is a modular function of weight 0, differentiating the modular equation shows that dJ is a vectorial modular function of dimension $g(g+1)/2$ and of weight $\mathrm{Sym}^2(\mathbb{C}^g)$. (In general differentiating a modular function of non trivial weight does not give a modular function, one as to take a Cohen-Rankin bracket instead. But this works in weight $\rho = 1$.)

A quick word about the Sym^2 action on $k^{g(g+1)/2}$. We illustrate this in dimension $g = 2$, the general case is similar. There are two natural Sym^2 representations, dual to each other. The first one is the standard action of a matrix $\begin{pmatrix} a & c \\ b & d \end{pmatrix}$ on the homogeneous polynomials of degree 2 in two variables x, y: $kx^2 \oplus kxy \oplus ky^2$. The other representation is to write an element (u_1, u_2, u_3) of k^3 as a matrix $u = \begin{pmatrix} u_1 & u_2 \\ u_2 & u_3 \end{pmatrix}$ and then the action of $m = \begin{pmatrix} a & c \\ b & d \end{pmatrix}$ is given by ${}^t mum$. This is this representation which comes naturally for modular functions, and to which we refer in the following. This explains the factor 2 in [KPR20, Definition 3.17].

Key Idea 6. *The differential dJ of a modular invariant J encodes (up to a sign) the full action on the differentials w_A.*

We remark that when J consists of more than $g(g+1)/2$ modular invariants, the action of the $\mathrm{Sym}^2 T_0(A)$ is recovered by working over $g(g+1)/2$ uniformisers dJ_i at $T_A\mathcal{A}_g$; this requires having the equations of the tangent space to A at $\mathcal{A}_g$ in the J_i coordinates (it suffices of course to have the equations satisfied by the J_i around A).

So to get an Elkies' like algorithm, the question boils down algebraically to being able to compute $dJ(A, w_A)$ given equations of A and an explicit basis of differentials $w_A = (w_1, \dots, w_g)$. We can then plug $dJ(A, w_A), J(A)$, $J(B)$ in the differential of the modular equation to find the value of $dJ(B, w_B)$ where $f^* w_B = w_A$ is normalised. Taking equations for B and an arbitrary basis w'_B of differentials, we can then compute $dJ(B, w'_B)$ and compare: $dJ(B, w_B) = N dJ(B, w'_B)$ where N is a matrix of dimension $g(g+1)/2$. Finding a matrix M of dimension g such that $\mathrm{Sym}^2 M = N$ then gives $f^* w'_B = M w_A$. From this equation we can either normalize w'_B or solve the differential system directly. We remark that if N is in the image of Sym^2, the only preimages are $\pm M$, with once again the inescapable sign ambiguity.

It is instructive to reformulate this approach geometrically. The values $J(A)$ can be seen as the coordinates of A in the moduli space $\mathcal{A}_g$ given by J. If A is defined over a field k, A corresponds to a k-point $\mathrm{Spec}\, k \to \mathcal{A}_g$. If $v \in T_A\mathcal{A}_g$ is a point of the tangent space to $\mathcal{A}_g$ at A, then $dJ(A)(v)$ corresponds to a morphism $\mathrm{Spec}\, k[\epsilon] \to \mathcal{A}_g$ factoring the map $\mathrm{Spec}\, k \to \mathcal{A}_g$ corresponding to A, where as usual $\epsilon^2 = 0$. But by the universal property of $\mathcal{A}_g$, this maps corresponds to an abelian variety $A_\epsilon/k[\epsilon]$. Since the composition $\mathrm{Spec}\, k \to \mathrm{Spec}\, k[\epsilon] \to \mathcal{A}_g$ corresponds to A by assumption, we have that the pullback $A_\epsilon \otimes_{k[\epsilon]} k \simeq A$. In other words, $dJ(A)(v)$ corresponds to a deformation A_ϵ of A.

Now we consider the modular correspondance $\Phi_\ell : \mathcal{A}_g(\ell) \to \mathcal{A}_g \times \mathcal{A}_g$ (this is the geometric interpretation of the modular polynomials). If we work with stacks, the two projections $\Phi_{\ell,i} : \mathcal{A}_g(\ell) \to \mathcal{A}_g$ are étale, so if $f : A \to B$ is an ℓ-isogeny such that (A, B) lies in the image of Φ_ℓ, we may define a deformation map $\mathcal{D}(f) \colon T_A(\mathcal{A}_g) \to T_B(\mathcal{A}_g)$ as

$$\mathcal{D}(f) := d\Phi_{\ell,2} \circ d\Phi_{\ell,1}{}^{-1}.$$

This is the geometric interpretation of differentiating modular polynomials.

From the analytic discussion above, we expect there is a link between the deformation map $\mathcal{D}(f)$ and the tangent map $df \colon T_0(A) \to T_0(B)$. The link is provided by the Kodaira-Spencer isomorphism: $T_A(\mathcal{A}_g)$ is isomorphic to $\mathrm{Sym}^2 T_0(A)$ (this is the algebraic interpretation of the fact that dJ is of weight Sym^2). So $\mathrm{Sym}^2 df : \mathrm{Sym}^2 T_0(A) \to T_0(B)$ can be interpreted via the Kodaira-Spencer isomorphism as a map $\mathrm{Sym}^2 df : T_A(\mathcal{A}_g) \to T_B(\mathcal{A}_g)$. We expect $\mathrm{Sym}^2 df$ and $\mathcal{D}(f) \colon T_A(\mathcal{A}_g) \to T_B(\mathcal{A}_g)$ to be related, indeed we have by [KPR20, Proposition 4.20]:

$$\mathrm{Sym}^2(df) = \ell \mathcal{D}(f). \tag{5.10}$$

So from $\mathcal{D}(f)$ we can recover df (up to a sign as always), provided we have an explicit version of the Kodaira-Spencer isomorphism (this is the geometric version of computing $dJ(A, w_A)$ given explicits A and w_A). We have the following modular interpretation of the deformation matrix $\mathcal{D}(f)$. Given the isogeny $f : A \to B$, we have seen that a tangent vector $v \in T_A \mathcal{A}_g$ corresponds to a deformation $A_\epsilon / k[\epsilon]$. The isogeny f lifts (uniquely) to A_ϵ by étaleness of $\mathcal{A}_g(\ell) \to \mathcal{A}_g$, and we let B_ϵ be the isogenous abelian scheme. This is a deformation of B, hence corresponds to a tangent vector $w \in T_B \mathcal{A}_g$. We have that $w = \mathcal{D}(f)(v)$.

We refer to [KPR20, § 4] for more details. The geometric interpretation is nice because at the level of stacks everything is smooth and the modular correspondance is étale, so the deformation map is always well defined. But in practice we work with modular polynomials, and this introduces two problems:

- This involves working with the coarse moduli spaces, and these are generically smooth but not smooth at all points (unlike the case of elliptic curves where the coarse space is $\mathbb{P}^1$).
- The modular polynomials does not even describe the modular correspondance at the level of coarse spaces but only a birational version of it.

So in [KPR20, § 4] we give precise geometric conditions on when this approach to computing isogenies will work (ie more precise than just saying it will work generically). As expected this is strongly linked to the presence of extra automorphisms on A and B and whether they respect the isogeny (see [KPR20, § 4.2.1]).

THE HILBERT CASE. We also explain how to extend this approach when considering β-isogenies on the Hilbert stack $\mathcal{H}_g$ [KPR20, § 4.2.3]. In this case the Kodaira-Spencer isomorphism in $\mathcal{H}_g$ is given by

$$T_A(\mathcal{H}_g) \simeq \mathrm{Hom}_{\mathbb{Z}_K \otimes O_S}(\mathrm{Lie}(A)^\vee, \mathrm{Lie}(A^\vee)) = \mathrm{Lie}(A^\vee) \otimes_{\mathbb{Z}_K \otimes O_S} \mathrm{Lie}(A) \otimes_{\mathbb{Z}_K} \mathbb{Z}_K^\vee.$$

Here since A/S is an S-point of $\mathcal{H}_g$, $\mathrm{Lie}(A)$ is a $\mathbb{Z}_K \otimes O_S$ locally free of rank 1.

We exploit the fact that the forgetting map $\mathcal{H}_g \to \mathcal{A}_g$ induces the following diagram:

$$\begin{array}{ccc} T_A(\mathcal{H}_g) & \longrightarrow & T_A(\mathcal{A}_g) \\ \downarrow & & \downarrow \\ \mathrm{Hom}_{\mathbb{Z}_K \otimes O_k}(\mathrm{Lie}(A)^\vee, \mathrm{Lie}(A^\vee)) & \longrightarrow & \mathrm{Hom}_{\mathrm{Sym}}(\mathrm{Lie}(A)^\vee, \mathrm{Lie}(A^\vee)). \end{array}$$

where the vertical arrows are the Kodaira–Spencer isomorphisms. This means that if we have an explicit version of the Kodaira–Spencer isomorphism in the Siegel case, we may recover an explicit version in the Hilbert case, provided we can identify the image of $T_A(\mathcal{H}_g)$ in $T_A(\mathcal{A}_g)$ in the above diagram. It is enough to have equations for the image of $\mathcal{H}_g$ in $\mathcal{A}_g$, since taking their tangents at A then cuts out the image of $T_A(\mathcal{H}_g)$. If $g = 2$, when working with coarse spaces, this amount to having the equation of the Humbert surface for $O_{\mathcal{F}}$. We refer to [KPR20] for more details.

5.4.3 *Lifting isogenies*

DERIVATIVE OF MODULAR POLYNOMIALS. It follows from the whole discussion that we do not need the full modular correspondance to extract the tangent map of f, we only need to know it locally at the isogenous points (locally meaning at $T_A(\mathcal{A}_g)$ and $T_B(\mathcal{A}_g)$). This explains why the results of Section 5.3.8 suffice to compute isogenies: we only need the evaluated modular polynomial along with their derivatives.

We briefly explain how to extend the methods of Section 5.3.8 to compute the derivative of the modular polynomials directly, rather than via an evaluation in $\mathbb{Z}_q/p^2\mathbb{Z}_q$ as in Remark 5.3.10. In this Section, we use these derivatives to relate the differentials of modular invariants $dJ_A(A, w_A)$, $dJ_B(B, w_B)$ for $f : (A, w_A) \to (B, w_B)$ a normalised isogeny. Conversely, we can compute the derivatives of the modular polynomials (notably the deformation matrix) from these differential invariants. Indeed, from the evaluation $\Phi_\ell(J_A, Y)$ we get the value of $\partial\Phi_\ell/\partial_Y(J_A, Y)$, and $\partial\Phi_\ell/\partial_X(J_A, Y)$ may be interpolated from the values $dJ_A(A, w_A)$, $dJ_B(B, w_B)$ of each of the normalised[3] isogenies $f : (A, w_A) \to (B, w_B)$.

In the complex analytic approach, the Newton's method to compute the theta constants naturally gives their derivative along the way. In the p_0-adic or CRT approach, either the isogeny formula we use naturally compute normalised isogenies (see Section 4.6), or alternatively we could fix differential basis and compute the action of the isogeny on these basis directly. This also holds for the three strategies outlined in Conjecture 5.3.14. For instance for the third strategy using canonical lifts, then in the Serre-Tate formal moduli, differentials of the lift $\widehat{A}$ are essentially given by elements $x_A \in T_p(A^\vee)(k)$ by [Kat81, § 3]. By [Kat81, Lemma 3.5.1], if $x_A \in T_p(A^\vee)(\bar{k})$ corresponds to $w_{\widehat{A}}$ and $x_B \in T_p(B^\vee)(\bar{k})$ corresponds to $w_{\widehat{B}}$ with $\hat{f}(x_B) = x_A$ ($\hat{f}$ being the dual isogeny), then $f : (\widehat{A}, w_{\widehat{A}}) \to (\widehat{B}, w_{\widehat{B}})$ is normalised. This allows to control the differentials by just computing the action of the isogeny $f : A \to B$ (ie over $\mathbb{F}_{p_0}$ rather than $\mathbb{Z}_{p_0}$) on $T_p(A)$.

LIFTING ISOGENIES. We have seen in Section 4.7 that once we have the action of the isogeny $f : A \to B$ on differentials, to solve the differential equation we need the characteristic to be large enough. If this is not the case, it suffices to lift the isogeny to $\mathbb{Z}_q/p^m\mathbb{Z}_q$, with m large enough, ie typically $m = O(\log \ell/\log p)$, so that $m \log p = O(\log \ell)$.

For that we lift A arbitrarily to $\widetilde{A}$, then we evaluate the modular polynomial on $J(\widetilde{A})$, and we solve for $J(\widetilde{B})$ via a Newton lift. Under our standard assumption that the derivatives of the modular polynomials are inversible, this is done in quasi-linear time once we have the evaluated modular polynomial. By Section 5.3.8 (and reusing the notations), the evaluation itself costs $\widetilde{O}(D^2E(m \log q + d^2M))$, with $E = 1$ under Conjecture 5.3.14 (or if $g = 1$). In [LS08] for $g = 1$, the lifting is done by computing Φ_ℓ and then evaluating it at $J(\widetilde{E})$ at p-adic precision m, for a cost of $\widetilde{O}(\ell^3 + \ell^2 m \log q)$.

Here we don't need the evaluation of the derivatives $\partial\Phi_\ell/\partial X_i$ of the modular polynomial, but they will be helpful in Chapter 6, and can be computed in the same complexity by the remarks above.

5.4.4 *Elkie's method for abelian surfaces*

This approach works for a general abelian variety. Instanciating the algorithm, we need an explicit method to solve the differential equation, see eg [KPR20, § 5] and the summary in Section 4.6 for Jacobians of genus 2 hyperelliptic curve, and an explicit version of the Kodaira-Spencer isomorphism.

Again, for Jacobians of genus 2 hyperelliptic curves this is done in [KPR20, § 3 and § 4.4]. Given an hyperelliptic Jacobian $J = \mathrm{Jac}(C)$, the curve equation gives a canonical basis $w_C = (dx/y, xdx/y)$ of differentials (the extension to characteristic $p = 2$ is straightforward). Here we use the canonical isomorphism $H^0(J, \Omega_J) = H^1(C, \Omega_C)$.

So we want to compute $dj(J, w_C)$ where j is a Igusa invariant. But if $\mathfrak{g}$ is a modular function of weight ρ, pulling back $\mathfrak{g}$ along the Torelli embedding gives $\mathfrak{g}$ as a covariant of the same weight (using a suitable normalisation as is done in [KPR20, Definition 3.7]). Furthermore by the Koecher principle, a modular function defined over $\mathcal{A}_g$ automatically extends to a toroidal compactification, so $\mathfrak{g}$ seen as a covariant is also well defined on semi-stable curves, not only on smooth curves. Since non-semi stable curves are of codimensions > 1, by normality of the moduli spaces $\mathfrak{g}$ is defined everywhere, hence is a polynomial covariant (in terms of the coefficients of the hyperelliptic curve).

We remark that since every principally polarised abelian surface is a Jacobian or a product of elliptic curves, the Torelli morphisms from compact curves to $\mathcal{A}_g$ is surjective (but it is of course no longer injective on non smooth curves). This can be exploited to go the other way around and study modular forms from covariants, see [CFv17;

[3] Of course whenever we have equations for the isogeny f, we can compute its action on differential, so we can always compute the normalised differential w_B.

CFG18]. Beware that conversely, a polynomial modular covariant $\mathfrak{g}$ need not give an holomorphic modular form, this is the case for instance for Igusa's invariant I_2 which is χ_{12}/χ_{10} when seen as a modular form. Indeed, $\mathfrak{g}$ may not be well defined at the compact curves whose Jacobian is a product of two elliptic curves (see [CFG18, § 4] for how to determine the order of vanishing of a covariant on this locus).

The identification of scalar modular forms and scalar covariants was already computed by Igusa in [Igu60]. But the case of vectorial modular form is actually easier and can be used to recover the scalar case. Indeed if $C : y^2 = F(x)$ with $F(x) = a_0 + \cdots + a_6x^6$, the form F is a canonical covariant of weight $\det^{-2}\mathrm{Sym}^6$, so $\mathrm{Disc}\cdot f$ is of weight $\det^8\mathrm{Sym}^6$. But a mass formula shows that the space of modular forms of weight $\det^8\mathrm{Sym}^6$ is of dimension 1. So if we identify such a modular form $f_{8,6}$ on the Siegel side, we know that $F = \lambda f_{8,6}/\chi_{10}$ since χ_{10} corresponds to the discriminant. Since $f_{8,6}$ is a cusp form, the quotient is well defined.

But such a function $f_{8,6}$ is well known, either as the modular function associated to some explicit lattice, or as the modular function given by the value at 0 of the derivative along z of the 6 odd level $(2,2)$ theta constants [KPR20, Example 2.8], [CFv17]. In fact this $f_{8,6}$ is defined over $\mathbb{Z}$ as can be checked from its Fourier coefficients, and while F is not well defined in characteristic $p = 2$, the covariant $\Delta\cdot F$ is well defined over $\mathbb{Z}$. So we have $\lambda = \pm 1$, and with the appropriate normalisation $\lambda = 1$, see [KPR20, § 3.3, § 4.4]. From this identification we can recover the modular functions dj_i as explicit covariants [KPR20, Theorem 3.14].

Remark 5.4.4 (Explicit Kodaira-Spencer in dimension $g = 1, 2$)**.** When $g = 1$, the j-invariant of an elliptic curve $E : y^2 = x^3 + ax + b$ only gives E up to isomorphism, and does not give the coefficients a, b. However, if we are also given $dj(E, w_E) = 18b/a$ (where $w_E = dx/y$ is the canonical differential), then we can recover a, b uniquely up to a common sign. In other words, fixing a, b (up to a sign) is the same as fixing $j(E), dj(E)$. Perhaps a better way to see this is to take an arbitrary representative $E' : y^2 = x^3 + a'x + b'$ of the isomorphism class of E, compute $dj(E') = u^2dj(E)$, and then act by the isomorphism $(x, y) \mapsto (u^2x, u^3y)$ to recover E. In both representations, we only need 2 coefficients.

The same strategy holds for abelian surfaces: namely if $A = \mathrm{Jac}(C)$, $j_i(A), dj_i(A, w_A)$ encode the equation of C (up to the action of $[-1]$) since we know conversely how to compute $dj_i(A, w_A)$ from the equation of C where $w_A = (dx/y, xdx/y)$ are the canonical differentials associated to this equation (with the ± 1 ambiguity coming from the fact that the dj_i only allows to get the $\mathrm{Sym}^2\gamma$ of the automorphism γ). Here we use the fact that in genus 2, automorphisms $\gamma : (x, y) \mapsto (\frac{ax+b}{cx+d}, \frac{y(ad-bc)}{(cx+d)^2})$ of the curve acts by $\begin{pmatrix} a & c \\ b & d \end{pmatrix}$ on the canonical differentials $w_A = (dx/y, xdx/y)$ (see [KPR20]), in other words automorphisms of C corresponds to automorphisms on the differentials. We note that we only need 6 coefficients to represent C, compared to 12 coefficients for the $j_i(A), dj_i(A)$. Another difference with elliptic curves is that given the $j_i(A)$, computing a representative C' is more difficult: Mestre's algorithm rely on finding a point on a conic, so there is an obstruction of degree 2 for the field of definition of C' to be the field of definition of the $j_i(A)$.

Finally, the same method in dimension $g = 1, 2$ allows to compute the Kodaira-Spencer isomorphism explicitly. Given a lift $E_\epsilon : y^2 = x^3 + (a + a_1\epsilon)x + (b + b_1\epsilon)$, we get $j(E_\epsilon) = j(E) + dj(E, w)\epsilon$. Comparing $dj(E, w)$ with $dj(E, w_E)$ allows to find $w = \pm uw_E$, so we get the differential corresponding to the lift E_ϵ. Conversely, given a differential $w = uw_E$, we can compute $dj(E, w) = u^2dj(E, w_E)$, and the lift E_ϵ is uniquely determined by the equation $j(E_\epsilon) = j(E) + dj(E, w)\epsilon$ above. Likewise, we can use the same method in genus 2: if $w = \gamma w_A$, and $J = (j_1, j_2, j_3)$ we can compute $dJ(A, w) = \mathrm{Sym}^2\gamma dJ(A, w_A)$, and then compute C_ϵ such that $J(C_\epsilon) = J(C) + dJ(A, w)\epsilon$, and conversely recover w (up to a sign) from C_ϵ. If $C_\epsilon = \sum(a_i + a'_i\epsilon)x^i$, the equation above give linear equations on the a'_i, so recovering them is easy. We refer to [KPR20, Remark 4.28] for more details.

We thus obtain [KPR20, Theorems 1.1, 6.2 and 6.3] as an application of Theorem 4.7.1:

Theorem 5.4.5. *Assume we are given a local version of the modular correspondance of level ℓ for abelian surfaces at A and B (implicitly assumed to be in the image of the correspondance), that $A = \mathrm{Jac}(C), B = \mathrm{Jac}(C')$ and $p > 8\ell + 7$ (or $p = 0$). Then we can compute the representation $f : C \to \mathrm{Jac}(C) \to \mathrm{Jac}(C')$ which is given by rational functions of total degree $O(\ell)$ in time $\widetilde{O}(\ell)$ operations in an extension k'/k of degree at most 8 along with $O(1)$ square roots in k'.*

The same hold in the Hilbert case for a β-modular isogeny. In this case, if $p = 0$ or $p > 4\,\mathrm{Tr}(\beta)+7$, the representation $f : C \to \mathrm{Jac}(C) \to \mathrm{Jac}(C')$ is given by rational functions of total degree $O(\mathrm{Tr}(\beta))$ and can be computed in quasi-linear time in k'/k along with $O(1)$ square roots in k'.

Proof. The modular correspondance allows to recover the tangent matrix, from which we apply the Newton algorithm to solve the differential system as explained in Section 4.7. Compared to Theorem 4.7.1, the extension k'/k can be of degree 8 because here we use Mestre's algorithm to recover a curve C from the Igusa invariants. This requires a degree 2 extension in general, and is not needed if the base field is finite. □

We note that it is easy to extend this result in the case where A or B is a product of elliptic curves, we give some details in [KPR20], and that if p is too small we can lift the isogeny (see Remark 4.7.2).

Using Section 5.3.8 and the notations there, lifting to sufficient p-adic precision $m = O(\log \ell / \log p)$, if necessary as in Section 5.4.1, the dominant step is the evaluation of the (derivative of) the modular polynomials, so we get:

Corollary 5.4.6. *Given two ℓ-isogenous abelian surfaces A and B defined over $\mathbb{F}_q$, then provided that the derivatives $\partial\Phi_\ell/\partial X$ and $\partial\Phi_\ell/\partial Y$ of the modular polynomials are invertible at J_A and J_B, the isogeny $A \to B$ can be recovered in time $\widetilde{O}(D(\log^2 q + d^4) + D^2(\log q + d^2))$ using Remark 5.3.8, $\widetilde{O}(D^2E(\log q + d^2))$ using Remark 5.3.12 [4] or $\widetilde{O}(D^2(\log q + d^2))$ under Conjecture 5.3.14. Given only A, once we have evaluated the modular polynomial (and its derivative) via the complexity above, recovering J_B take time $\widetilde{O}(D \log q)$ operations over $\mathbb{F}_q$, ie $\widetilde{O}(D \log^2 q)$.*

5.5 APPLICATIONS TO POINT COUNTING FOR ABELIAN SURFACES

Since we know how to compute isogenies on abelian surfaces from modular polynomials, it is straightforward[5] to adapt the SEA algorithm from elliptic curves to abelian surfaces, by modifying accordingly the Schoof algorithm in the Siegel or Hilbert case.

5.5.1 *Complexity of Schoof's algorithm for abelian surfaces in the Siegel case*

The goal is to compute the characteristic polynomial of the Frobenius $\chi_\pi(X) = X^4 - s_1X^3 + (s_2+2q)X^2 - qs_1X + q^2$ by finding its value modulo ℓ for several primes ℓ. Using the CRT and Weil's bounds, which give $|s_1| \leq 4\sqrt{q}, |s_2| \leq 4q$, we need $O(\log q)$ primes ℓ of size $O(\log q)$.

Schoof's algorithm find $\chi_\pi \mod \ell$ by computing π over an efficient representation of $A[\ell]$.

We follow [GS12] which is the currently fastest implementation for Jacobians of hyperelliptic curves of genus 2. For simplicity we assume asymptotically fast algorithms for the basic arithmetic operations.

1. Writing a polynomial system for $A[\ell]$: $\widetilde{O}(\ell^2)$ operations in $\mathbb{F}_q$. This system is of total degree ℓ^4 and composed of bivariate polynomials of degree $O(\ell^2)$. It can be obtained as follow: if $A = \operatorname{Jac} C$, Cantor gives an explicit algorithm[6] to compute the multiplication by ℓ on a point $P \in C$ (or rather its divisor $(P) - (\infty)$); this is given by polynomials of degree $O(\ell^2)$. The system describing $A[\ell]$ is given by writing formally $D = (P_1) + (P_2) - 2(\infty)$ and then writing $\ell(P_1 - \infty) = -(\ell P_2 - \infty)$.

2. Computing a suitable representation of $A[\ell]$ (ie an efficient parametrisation) using a resultant and subresultant (this gives a triangular representation of the system): this uses an evaluation-interpolation approach[7] to compute ℓ^4 resultants of polynomials of degree ℓ^2 in one variable for a total cost of $\widetilde{O}(\ell^6)$ operations in $\mathbb{F}_q$ [GS12, § 3.2].

3. In this efficient representation, computing the addition on formal points of ℓ-torsion costs $\widetilde{O}(\ell^4)$ while applying the Frobenius $\widetilde{O}(\ell^4 \log q)$ operations in $\mathbb{F}_q$.

   So finding the values $s_1, s_2 \mod \ell$ such that $\chi_\pi(D) = 0$ for D a formal point of $A[\ell]$ costs $O(1)$ applications of the Frobenius and $O(\ell)$ operations in $A[\ell]$ (using a baby-step giant-step approach), for a total cost of $\widetilde{O}(\ell^5 + \ell^4 \log q)$ operations in $\mathbb{F}_q$.

[4] Since we unfortunately don't quite have a quasi-linear algorithm to evaluate the modular polynomial over a number field, either with the analytic method or the p-adic or CRT method, the best strategy between the two depends on the value of ℓ relative to $\log q$.

[5] The hard part is the implementation, since this need to combine a lot of non trivial algorithms: fast evaluation of theta constants, fast evaluation of modular polynomials, fast resolution of the differential equation, fast resultants and subresultants to get an efficient representation of the kernel, CRT over a real order…This is currently being implemented by Kieffer.

[6] See also Remark 4.7.3.

[7] By [Vil18] the evaluation-interpolation approach to computing bivariate resultants is no longer the fastest method asymptotically. However for the cryptographic sizes for point counting, the interpolation approach is probably faster in practice, so we stick to the complexity of the standard evaluation/interpolation approach to the resultant. The new algorithm computes the resultant of two polynomials of degree d on X and n on Y in time $\widetilde{O}(n^{2-1/\omega}d)$, instead of $\widetilde{O}(n^2d)$ via the evaluation/interpolation method which uses $O(nd)$ evaluation points, and compute $O(nd)$ univariate resultants of degree n. When $d = n$, this improve the complexity from $\widetilde{O}(n^3)$ to $\widetilde{O}(n^{2.58})$ using $\omega = 2.376$ from the Coppersmith–Winograd algorithm (the current best bound is a variant of the Coppersmith-Winograd algorithm which gives $\omega \leq 2.373$ [AW21]). This faster resultant can also be used to compute a Grobner basis, so in good cases a triangular representation of the system [Vil18, § 7]. In the Siegel case, this improves the complexity of finding an efficient parametrisation of $A[\ell]$ to $\widetilde{O}(\ell^{5.16})$ operations in $\mathbb{F}_q$, and of the kernel of an ℓ-isogeny to $\widetilde{O}(\ell^{2.58})$. In the Hilbert case, an efficient parametrisation of $A[\beta]$ is computed in $\widetilde{O}(\ell^{2.58})$, and of the kernel of a β-isogeny in $\widetilde{O}(\ell^{1.29})$. This does not change the complexities of the algorithms below since the dominant steps are elsewhere, except in the Schoof-Siegel case which would be in $\widetilde{O}(q^{7.16})$.

4. So taking $\ell = O(\log q)$, we get than one step of the CRT costs $\widetilde{O}(\log q^6)$ operations in $\mathbb{F}_q$.

5. The total cost is then $\widetilde{O}(\log q^7)$ operations in $\mathbb{F}_q$, ie a binary cost of $\widetilde{O}(\log q^8)$.

Recall that for elliptic curves, an efficient representation of $E[\ell]$ is given by the ℓ-division polynomial, and the complexity is dominated by the evaluation of the Frobenius modulo this polynomial of degree ℓ^2; this costs $\widetilde{O}(\ell^2 \log q)$ operations over $\mathbb{F}_q$. So a step of the CRT costs $\widetilde{O}(\log q^3)$ operations in $\mathbb{F}_q$, for a total binary cost of $\widetilde{O}(\log q^5)$.

In genus 2 we could hope to find a faster algorithm to get the efficient representation of $A[\ell]$, a hypothetical quasi-linear algorithm would cost $\widetilde{O}(\ell^4)$ operations in $\mathbb{F}_q$. The dominating step would then be the BSGS step along with the evaluation of the Frobenius, which cost $\widetilde{O}(\log q^5)$ operations in $\mathbb{F}_q$, and the total binary complexity would be $\widetilde{O}(\log q^7)$.

5.5.2 *Complexity of a SEA algorithm for abelian surfaces in the Siegel case*

The SEA like algorithm follows the same approach as the Schoof algorithm except it computes an ℓ-isogeny $f : A \to B$ to recover $\chi_\pi \mod \ell$ by working over $K = \operatorname{Ker} f$ which is only of degree ℓ^2 rather than ℓ^4. Let us detail the steps, assuming for simplicity that we are over $\mathbb{F}_p$ (or that if $q = p^d$, d is small compared to $O(\log p)$, so that we have quasi-linear evaluation of the modular polynomials over a number field), and that $A = \operatorname{Jac} C$.

- Evaluating the ℓ-modular polynomial: $\widetilde{O}(\ell^3 \log^2 q + \ell^6 \log q)$ binary operations by Proposition 5.3.7, or $\widetilde{O}(\ell^6)$ with small parameters.
- Finding a root: this requires computing the Frobenius modulo a univariate polynomial of degree ℓ^3, for a cost[8] of $\widetilde{O}(\ell^3 \log q)$ operations in $\mathbb{F}_q$. Here we make the same heuristic as in the genus 1 case with respect to the density of Atkin vs Elkies prime, that there is a rational root for sufficiently many ℓ. We assume this is the case, and we denote $f : A \to B$ the isogeny, and K its kernel.
- Computing a representation of the isogeny of the form $C \to \operatorname{Jac}(C)$, given by polynomials of degree $O(\ell)$: $\widetilde{O}(\ell + \log p)$ operations in $\mathbb{F}_q$ by Section 5.4 (the $O(\log q)$ is for the square root). This allows us to represent the kernel K by writing $D = (P_1) + (P_2) - 2(\infty)$ and then plugging the equations $f(P_1 - \infty) = -f(P_2 - \infty)$.
- Computing a suitable representation of K (ie an efficient parametrisation) using a resultant and subresultant: this uses an evaluation-interpolation approach to compute ℓ^2 resultants of polynomials of degree ℓ in one variable for a total cost of $\widetilde{O}(\ell^3)$ operations in $\mathbb{F}_q$.
- In this efficient representation, computing the addition on formal points of K costs $\widetilde{O}(\ell^2)$ while applying the Frobenius $\widetilde{O}(\ell^2 \log q)$ in $\mathbb{F}_q$.

  So finding the values $s_1, s_2 \mod \ell$ such that $\chi_\pi(D) = 0$ for D by the same approach as in Schoof algorithm costs $\widetilde{O}(\ell^3 + \ell^2 \log q)$ operations in $\mathbb{F}_q$.
- So taking $\ell = O(\log q)$, we get than the dominating step is the evaluation with a cost of $\widetilde{O}(\log q^7)$ binary operations. The rest only take $\widetilde{O}(\log q^3)$ operations in $\mathbb{F}_q$. If A is given by small parameters, the evaluation step is still dominant but only costs $\widetilde{O}(\log q^6)$.
- The total cost is then $\widetilde{O}(\log q^8)$, or $\widetilde{O}(\log q^7)$ with small parameters. So compared to the Schoof algorithm of Section 5.5.1 we gain a degree in the asymptotic complexity when the parameters of the curve are small, which is usually the case in the cryptographic setting (the remark that small coefficients make the SEA algorithm asymptotically faster is due to Kieffer).

  With a faster evaluation, since the evaluated modular polynomial is of size $O(\ell^3 \log q)$, and finding a root cost $O(\ell^3 \log q^2)$ binary operations anyway, we could hope for an evaluation algorithm of complexity $\widetilde{O}(\log q^5)$. This would yield a total cost of $\widetilde{O}(\log q^6)$. The dominating step would then be shared with finding a root.

Recall that for elliptic curves, the modular polynomial is of size $\widetilde{O}(\ell^3)$, and the dominating step is the evaluation which costs $\widetilde{O}(\ell^2 + \ell \log q)$ and finding of a root which costs $\widetilde{O}(\ell \log q)$ operations over $\mathbb{F}_q$ respectively. The binary complexity is of $\widetilde{O}(\log q^3)$ by prime ℓ, and the total complexity of $\widetilde{O}(\log q^4)$.

[8] More precisely to find the rational roots of P, we compute X^q modulo P via a fast exponentiation, then take the gcd Q with P. If there are several roots we may then apply the Cantor-Zassenhaus algorithm to Q. The dominating step in our setting is essentially the Frobenius computation.

5.5.3 Complexity of Schoof's algorithm for abelian surfaces in the Hilbert case

If we have (explicit) real multiplication by $O_{\mathcal{F}}$, the Schoof algorithm finds $\chi_\pi \mod \alpha$ by computing π over an efficient representation of $A[\alpha]$, $\alpha \in O_{\mathcal{F}}$. If α is of norm ℓ, $A[\alpha]$ is of degree ℓ^2 (because $[\alpha]$ is an α^2-isogeny).

In fact rather than determining π directly, it is enough to recover $\psi = \pi + \overline{\pi}$ as an element of $O_{\mathcal{F}}$. Note that the minimal polynomial of ψ is $X^2 - s_1 X + s_2$. Letting $O_{\mathcal{F}} = \mathbb{Z}[\eta]$, we can write $\psi = m + n\eta$ and we have $m, n \in O(\sqrt{q})$ [GKS11, § 3.1]. See also [Abe20, § 2.2] and Section 5.5.5 for arbitrary g.

So the Schoof algorithm still uses $O(\log q)$ primes α of norm $O(\log q)$. Here the algorithm assumes that η is efficiently computable, and restrict to α above a split prime ℓ: $\ell = \alpha\alpha^\sigma$ where σ is the Galois involution.

1. Writing a polynomial system for $A[\alpha]$: $\widetilde{O}(\ell)$ operations in $\mathbb{F}_q$. This system is of total degree ℓ^2 and composed of bivariate polynomials of degree $O(\ell)$. It can be obtained by writing $\alpha = a + b\eta$, with $a, b \in O(\sqrt{\ell})$ and then proceeding as in the Siegel case.

2. Computing a suitable representation of $A[\alpha]$ an evaluation-interpolation approach computes ℓ^2 resultants of polynomials of degree ℓ in one variable for a total cost of $\widetilde{O}(\ell^3)$ operations in $\mathbb{F}_q$.

3. Finding the values $m, n \mod \ell$ (by working over $A[\alpha]$ and $A[\alpha^\sigma]$), costs $\widetilde{O}(\ell^2(\ell + \log q))$ field operations. A baby-step giant-step algorithm could yield a complexity of $\widetilde{O}(\ell^2(\sqrt{\ell} + \log q))$ field operations. A simplification here is that the real endomorphisms acts by a scalar over $A[\alpha]$ since $O_{\mathcal{F}}/\alpha \simeq \mathbb{Z}/\ell\mathbb{Z}$, see [GKS11, Theorem 1]. So the values of m and n are recovered modulo α and α^σ, hence modulo ℓ.

4. So taking $\ell = O(\log q)$, we get than one step of the CRT costs $\widetilde{O}(\log q^3)$ operations in $\mathbb{F}_q$.

5. The total cost is then $\widetilde{O}(\log q^4)$ operations in $\mathbb{F}_q$, ie $\widetilde{O}(\log q^5)$. Here the dominating steps are the efficient representation of $A[\alpha]$ (which is not done in quasi-linear time $O(\ell^2)$) and the BSGS step. So even improving the computation of $A[\alpha]$ would not change the complexity.

5.5.4 Complexity of a SEA algorithm for abelian surfaces in the Hilbert case

We finish by a SEA like algorithm in the Hilbert case. We reuse the notations from Section 5.5.3. Note that we do not need here η to be computable, so rather than looking for $\psi := \pi + \overline{\pi} = m + n\eta$, we will simply recover ψ as an element in $O_{\mathcal{F}}$ directly by doing a CRT over $O_{\mathcal{F}}$ rather than via a CRT of its minimal polynomial over $\mathbb{Z}$.

Indeed, to exploit the SEA algorithm we need a rational root of a α-modular polynomial, for a prime α. The situation is completely similar to the elliptic curve case (we even have volcanoes as shown in [IT14]), so we expect to have an Elkies prime α about half the time. If we wanted to do the CRT over $\mathbb{Z}$ we would also need a rational root for α^σ, the Galois conjugate, so that we can reconstruct the information modulo $\ell = \alpha\alpha^\sigma$. We expect this would decrease the probability by a factor of two. This would not change the asymptotic complexity but would lose a factor 2 in practice. So I suggested to Kieffer to look at this strategy instead, which he is currently implementing. This should help to have a nice record breaking point counting for an abelian surface with RM!

We restrict to the case $q = p^d$ with a fixed (or sufficiently small) d for simplicity.

- Evaluating the α-modular polynomial: $\widetilde{O}(\ell \log^2 q + \ell^2 \log q)$ binary operations by Proposition 5.3.7.

- Finding a root: this requires computing the Frobenius modulo a univariate polynomial of degree ℓ, for a cost of $\widetilde{O}(\ell \log q)$ operations in $\mathbb{F}_q$. We assume we are in the Elkies case and there is a rational root, corresponding to an isogeny f with kernel K.

- Computing a representation of the isogeny of the form $C \to \mathrm{Jac}(C)$, given by polynomials of degree $O(\mathrm{Tr}\,\alpha)$: $\widetilde{O}(\sqrt{\ell} + \log q)$ operations in $\mathbb{F}_q$ by Section 5.4 (the $O(\log q)$ is for the square root, and we can take a representative $\alpha = a + b\eta$ with $a, b \in O(\sqrt{\ell})$). This allows us to represent the kernel K by writing $D = (P_1) + (P_2) - 2(\infty)$ and then plugging the equations $f(P_1 - \infty) = -f(P_2 - \infty)$.

- Computing a suitable representation of K (ie an efficient parametrisation) using a resultant and subresultant: this uses an evaluation-interpolation approach to compute ℓ resultants of polynomials of degree $\sqrt{\ell}$ in one variable for a total cost of $\widetilde{O}(\ell^{3/2})$ operations in $\mathbb{F}_q$.

- In this efficient representation, we compute m such that $\psi - m = 0$ on K. Since K is an $O_{\mathcal{K}}/\alpha = \mathbb{Z}/\ell\mathbb{Z}$-module, this determines ψ modulo α. Applying π, this is the same as $\pi^2 - q - \pi m = 0$ on K. So finding m requires a DLP and $O(1)$ applications of the Frobenius, for a complexity of $\widetilde{O}(\ell(\ell + \log q))$ operations over $\mathbb{F}_q$, or $\widetilde{O}(\ell(\sqrt{\ell} + \log q))$ with a BSGS algorithm.
- The CRT in $O_{\mathcal{K}}$ requires $O(\log q)$ primes α of norm $\ell = O(\log q)$.
- The total binary cost is then $\widetilde{O}(\log q^4)$, exactly as in SEA's algorithm for elliptic curves! The dominating step here is the evaluation and finding a root of the modular polynomial, which cost $\widetilde{O}(\log q^3)$ binary operations. Having small coefficient helps the evaluation, but does not change the asymptotic (apart from logarithmic factors) since finding a root was also dominant.

Again, we refer to future work from Kieffer for more details, and hopefully a record computation.

5.5.5 *Complexity of a Schoof-Pila and SEA like algorithm in higher dimension*

When we work with a Jacobian of a curve $A = \operatorname{Jac} C$, the curve provides a compact way to represent the multiplication by ℓ or an isogeny. But we have seen that already in Schoof's algorithm in dimension 2, recovering an efficient representation of $\operatorname{Jac} C[\ell]$ by writing a divisor as a sum of two points on the curve is not quasi-linear in the degree of the system. The situation gets worse in higher dimension, since we need to express the divisor as a sum of g points. Also we want to generalize this to all abelian varieties.

The Schoof-Pila algorithm has been studied in details, and in general we have algorithms with complexity $O(\log q^c)$ with c polynomial in g [AH01] (both for curves and abelian varieties), linear in g in the hyperelliptic case [AGS19b], and even bounded [Abe20] for hyperelliptic curves with real multiplication (the bound is $9 + \epsilon$ and conjectured to be $7 + \epsilon$, but see Remark 4.7.3).

We recall that the characteristic polynomial of the Frobenius $\chi_\pi(X) = X^{2g} + \sum_{i=0}^{2g-1} c_i X^i$ satisfy $c_i = q^{2g-i} c_{2g-i}$ and $|c_i| \leq \binom{2g}{i} q^{(2g-i)/2}$ by the Weil bounds. Let $\psi = \pi + \overline{\pi}$ be the trace, this is a real endomorphism and the minimal polynomial of the Frobenius over $\mathbb{Z}[\psi]$ is given by $X^2 - \psi X + q = 0$. So the characteristic polynomial $\chi_\psi(X) = X^g + \sum_{i=0}^{g-1} s_i X^i$ determines χ_π, and we have $s_i = O(q^{(g-i)/2})$ [Abe20, Proposition 4]. If the real field is given by $K = \mathbb{Q}(\eta)$, we can write $\psi = \sum_{i=0}^{g-1} a_i \eta^i$ with $a_i = O(\sqrt{q})$ by loc. cite. The relationship between the coefficients is as follow: $\chi_\pi(X) = \sum_{i=0}^{g} \sigma_i (X^{2g-i} + q^{g-i} X^i)$ with $\sigma_i = c_{2g-i}$ if $i \neq g$ and $\sigma_g = c_g/2, \sigma_i = O(q^{i/2})$. Then $\sum_{i=0}^{g} \sigma_{g-i}(\pi^i + \hat{\pi}^i) = 0$. Write $\pi^i + \hat{\pi}^i = \psi^i + \sum_{j=0}^{i-1} \alpha_{ij} \psi^j$, $\alpha_{ij} = O(q^{(i-j)/2})$. Then $s_i = \sum_{j=i}^{g} \alpha_{ji} \sigma_{g-j}$, see [Abe20, § 2.2].

For point counting in the Hilbert case, in [Abe20] the action of η is assumed to be known, and ψ is recovered by working over totally split primes and doing a CRT over $\mathbb{Z}$. In our case, using modular polynomials, we only need to recover the action of ψ modular β, β of prime norm, and then doing a CRT over $O_{\mathcal{K}}$. We could also relax the condition that β is of prime norm, but in this case we would need to know how $O_{\mathcal{K}}$ acts on A (eg how η acts), so that we can then identify the kernel as an $O_{\mathcal{K}}/\beta$ module, to identify ψ as an element of $O_{\mathcal{K}}/\beta$.

Let us be optimistic and assume a quasi-linear algorithm could be found for all the steps in the Schoof or SEA like algorithm, and let's look at the complexity.

For a Schoof like algorithm:

- We would work with an efficient representation of $A[\ell]$ in the Siegel case, or $A[\alpha]$ in the Hilbert case (α above a totally split prime ℓ), for a complexity of $\widetilde{O}(\ell^{2g})$ or $\widetilde{O}(\ell^2)$ operations in $\mathbb{F}_q$.
- In the Siegel case, we need to recover g coefficients modulo ℓ, so a BSGS algorithm costs $\widetilde{O}(\ell^{2g}(\ell^{g/2} + \log q))$ operations in $\mathbb{F}_q$ (taking into account the Frobenius computations). In the Hilbert case this would be $\widetilde{O}(\ell^2(\ell^{1/2} + \log q))$ operations in $\mathbb{F}_q$.
- The cost for a prime is then $\widetilde{O}(\log q^{5g/2})$ (assuming $g \geq 2$) and $\widetilde{O}(\log q^3)$ operations in $\mathbb{F}_q$ respectively.
- So the ideal algorithm would cost $\widetilde{O}(\log q^{5g/2+2})$ and $\widetilde{O}(\log q^5)$ respectively. The dominating step in both cases would be the BSGS step.

For a SEA like algorithm, letting $N = g(g+1)/2$ be the dimension of $\mathcal{A}_g$:

- We have a magic evaluation algorithm which evaluate the modular polynomial in quasi-linear time. This costs $\widetilde{O}(\ell^N)$ operations in $\mathbb{F}_q$ in the Siegel case versus $\widetilde{O}(\ell)$ operations in $\mathbb{F}_q$ in the Hilbert case.

- Finding a root then costs $\widetilde{O}(\ell^N \log q)$ and $\widetilde{O}(\ell \log q)$ operations in $\mathbb{F}_q$ respectively.
- The magical algorithm to get an efficient representation of the kernel would cost $\widetilde{O}(\ell^g)$ and $\widetilde{O}(\ell)$ operations in $\mathbb{F}_q$ respectively.
- Recovering the information for a prime would cost $\widetilde{O}(\ell^g(\ell^{g/2} + \log q))$ and $\widetilde{O}(\ell(\ell^{1/2} + \log q))$ operations in $\mathbb{F}_q$ respectively.
- The cost for a prime is then $\widetilde{O}(\log q^{1+N})$ and $\widetilde{O}(\log q^2)$ operations in $\mathbb{F}_q$ respectively.
- So the ideal algorithm would cost $\widetilde{O}(\log q^{N+2})$ and $\widetilde{O}(\log q^4)$ respectively. In both cases the dominating step would be finding a root.

It is interesting to note that with ideal algorithms, in the Siegel case the Schoof like method has the same complexity as the SEA like method when $g = 4$, and becomes faster for $g > 4$. Indeed, with $g = 4$ the SEA like method allows to parametrize a system of degree ℓ^4 rather than ℓ^8, but for that we need to find a root of a polynomial of degree $O(\ell^{10})$. Of course in practice there is no quasi-linear algorithm to compute an efficient representation of $A[\ell]$ or the kernel K, so the break off point is probably a bit higher.

Let us look at the hypothesis of a magic evaluation algorithm for modular polynomials and compare it to the complexity if we make instead the more reasonable hypothesis that there exists a quasi-linear uniform algorithm to compute theta constants (and conversely period matrices from theta constants), or if we apply the p_0-adic algorithm from Section 5.3.8. We restrict to $q = p^d$, with d constant (or sufficiently small compared to $\log p$). Without small parameters, in the Siegel case we have a polynomial with $O(\ell^N)$ coefficients of heights $\widetilde{O}(\ell^N \log q)$, for a total size of $\widetilde{O}(\log q^{2N+1})$ since $\ell = O(\log q)$. With small parameters over $\mathbb{F}_p$, the coefficients are of height ℓ^N, for a total size of $\widetilde{O}(\log q^{2N})$. This becomes the dominant step if $g > 1$. For Hilbert polynomials, we have $O(\ell)$ coefficients of size $\widetilde{O}(\ell \log q)$ or $\widetilde{O}(\ell)$ with small parameter, for a total size of $\widetilde{O}(\log q^3)$ or $\widetilde{O}(\log q^2)$ with small parameters. While this is larger than the size of the reduction to $\mathbb{F}_q$, the cost is the same complexity as finding a root anyway. A caveat in the p_0-adic method is that we also need the derivative of the modular polynomials, and as explained in Section 5.4.2 this is more expansive: $\widetilde{O}(\log q^{2N+g+1})$ in the Siegel case, and $\widetilde{O}(\log q^4)$ in the Hilbert case (we gain a factor $\log q$ with small parameters).

So in the Hilbert case, fast evaluation of theta functions or p_0-adic lifting (with small parameters) provide a way to evaluate modular polynomials sufficiently fast for an "optimal algorithm". The main remaining stumbling block to get a quasi-optimal algorithm is the computation of an efficient representation of the kernel of the isogeny, we get a leeway of $\widetilde{O}(\ell^2 + \log^2 q)$ operations in $\mathbb{F}_q$ for that.

The most promising candidates are Jacobians of hyperelliptic curve. Heuristically, under the conjecture of Remark 4.7.3 the representation $f : C \to B$ of the isogenies will be of degree $O(\ell^{1/g})$. We can then adapt the complexity analysis of [Abe20, § 4.2] which gives a dominant part of $\widetilde{O}(d_x\delta^2 \log q + \delta^2 \log^2 q)$. In our case, $d_x = O(\ell^{1/g})$ and $\delta = O(\ell)$, which gives a complexity of $\widetilde{O}(\ell^{2+1/g} \log q + \ell^2 \log^2 q)$ to compute the geometric resolution, ie $\widetilde{O}(\log^4 q)$ with $\ell = O(\log q)$, which would yield an $\widetilde{O}(\log^5 q)$ point counting algorithm. But unfortunately a strong caveat is that the complexity analysis of [Abe20] works for an endomorphism. In our case, while the domain is a Jacobian of an hyperelliptic curve, the codomain won't even be a Jacobian in general. So we need to adapt the algorithm when the target variety is, for instance, represented by theta functions. This is the main obstruction to having an $\widetilde{O}(\log^5 q)$ algorithm for an RM hyperelliptic curve. (For this complexity, the evaluation of the modular polynomials and their derivatives using the p_0-adic method is fast enough even without small parameters.)

So there is hope to obtain at least an $\widetilde{O}(\log^5 q)$ "almost optimal" point counting complexity in this case, but there is a lot of work remaining. It would work as follow, when $A = \mathrm{Jac}(C)/\mathbb{F}_p$ is the Jacobian of an hyperelliptic curve of genus g with (explicit) real multiplication. For simplicity we assume that we have a rational level 2 theta structure, so we can use modular polynomials in theta invariants.

- Evaluate the Hilbert modular polynomials Φ_β.
- If there is a rational root, recover the action on differentials from the derivative of the theta constants. We refer to Section 5.7 for how we can use the heat equation to get an explicit Kodaira-Spencer isomorphism from the theta constants, and how to adapt this to other modular invariants.
- Represent the isogeny as $f : C \to B$, where B is given by theta functions.
- Solve the differential equation via Newton iterations (after an initialisation to enough precision).

- Use geometric resolution to represent the kernel efficiently (this is the most difficult step).
- Compute the action of the Frobenius.

Alternatively we could look at the contragredient isogeny $\tilde{f} : B \to \mathrm{Jac}(C)$, and determine ψ on $\mathrm{Ker}\,\tilde{f}$, this allows to work with the Jacobian on the target for easier determination of the kernel. In particular, when $g = 3$, B will (generically) be a Jacobian which simplify the computations. I am optimistic there exists an $\widetilde{O}(\log^4 q)$ algorithm.

Indeed, [AGS19a] gives an $\widetilde{O}(\log^6 p)$ (so a bit worse than the "optimal") algorithm for point counting on an hyperelliptic curve with RM using Schoof's method. In our case, the kernel of $\tilde{f}$ is modelled via the map $C' \to B = \mathrm{Jac}(C') \to \mathrm{Jac}(C)$ (where C' may not be hyperelliptic), and is given by polynomials of degree $O(\ell^{1/g})$ in the coordinates of C' (see Remark 4.7.3). Using the analysis of [AGS19a, § 3] the full kernel is computed by first taking triresultants, this gives bivariate polynomials of degree $O(\ell^{2/3})$ in time $\widetilde{O}(\ell^{5/3})$ and then by biresultants, this gives a univariate polynomial of degree $O(\ell^{4/3})$ in time $\widetilde{O}(\ell^2)$ (operations over $\mathbb{F}_q$). This is fast enough for an $\widetilde{O}(\log^4 q)$ algorithm. The main remaining block is to find a sufficiently efficient way to evaluate the modular polynomials and their derivatives. For instance the p_0-adic approach works if we have small parameters. Alternatively, we could use the analytic method; Labrande gives in [Lab16, Remark 7.4.5] a preliminary potential quasi-linear algorithm to evaluate theta functions when $g = 3$ (see Section 5.3.4). We also need to recover the period matrix sufficiently fast, but since we are over a Jacobian we could compute hyperelliptic periods at low precision to get the correct sign choices.

5.6 APPLICATIONS TO EXPLORING ISOGENY GRAPHS

Exploring isogeny graphs is a core toolbox for many applications: it allows to enumerate all abelian varieties having certain properties (which force them to be isogenous). We will see an example for algorithms to compute class polynomials in Chapter 7.

Since over a finite field the (non polarised) isogeny class is characterised by the characteristic polynomial of the Frobenius, exploring the isogeny graphs can also be used to find special abelian varieties in this class. This is typically used to try to find Jacobians of curves when looking for curves with many points.

5.6.1 *Isogeny graphs over a finite field via modular polynomials*

Of course modular polynomials are the key tools to explore isogeny graphs. When plugging the invariants of an abelian variety A, the evaluated modular polynomials parametrize the invariants of the ℓ-isogenous varieties B.

If the parametrization of the modular polynomials is done such that we have a univariate polynomial parametrizing the solutions, then over a finite field finding a rational root amount to a Frobenius computation (this is the dominating step), for a binary cost of $\widetilde{O}(D \log q^2)$, where $D = O(\ell^{g(g+1)/2})$ is the degree, ie the number $\#\Gamma/\Gamma_0(\ell)$ of isogenies). We refer to Section 5.3.8 for the evaluation of modular polynomials.

There is a slight technicality here: if we find k-rational modular invariants J_A, J_B such that $\Phi_\ell(J_A, J_B) = 0$, it only means that there is an isogeny $f : A \to B$ defined over $\overline{k}$. Indeed with modular polynomials we work over the coarse moduli space rather than the fine moduli space (which is an algebraic stack). So a k-rational point x of the coarse modular correspondance may not correspond to a rational isogeny. The obstruction is measured by an element of $H^2(\mathrm{Spec}\,k, \mathrm{Aut}(x))$ in the sens of Giraud (ie as gerbes bound by the band induced by $\mathrm{Aut}(x)$). In our case, if the automorphisms of A preserve the kernel of the isogeny f, (A, K) and A then have the same automorphisms, so the obstruction in the H^2 is the same. So if A descends to k then f too (but note that we may need to twist B). See [DR73, § VI.3.1] and [KPR20, Proposition 4.11]. In particular if A only has generic automorphisms then they all stabilize K (in the Siegel case because the generic automorphisms are $1, -1$ and in the Hilbert case because they are given by real multiplications but K is stable under the real multiplications for an Hilbert modular correspondance), so there is no problem of descent.

A related remark: if A and B are already k-isogenous, and $\mathrm{End}_{\overline{k}}(A) = \mathrm{End}_k(A)$ then every $\overline{k}$-isogeny $f : A \to B$ is actually defined over k. Indeed the free $\mathbb{Z}$-module $\mathrm{Hom}_k(A, B)$ is not empty by hypothesis, and has the same rank as $\mathrm{Hom}_{\overline{k}}(A, B)$ by the hypothesis on $\mathrm{End}_k(A)$. So a multiple g of f is rational, but if g is $\overline{k}$-divisible it is k-divisible, so f is rational.

5.6.2 Isogeny graphs over a finite field via explicit isogeny computations

So everything is nice when we have modular polynomials, but we may want to compute isogeny graphs even without them. In fact, in Section 5.3.8, we have seen that we can iterate through isogenous abelian varieties to evaluate the modular polynomials. A solution is to brute force the problem by computing the full ℓ-torsion, iterating through all maximal isotropic kernels and compute the corresponding isogenies using Chapter 4.

We can only hope to get a somewhat reasonable algorithm if the base field is a finite field $\mathbb{F}_q$. Then, we could compute the ℓ-torsion directly using the same method as the Schoof like algorithms of Section 5.5, ie using division polynomials. This is not too bad for abelian surfaces (or elliptic curves), as we have seen in Section 5.5.1 that we may compute an efficient representation of $A[\ell]$ in time $\widetilde{O}(\ell^6 \log q)$ binary operations (resp $\widetilde{O}(\ell^2 \log q)$ for elliptic curves).

In this section we consider an alternative strategy, in arbitrary dimension, when the characteristic polynomial of the Frobenius of $A/\mathbb{F}_q$ is known (we can also compute it in time polynomial in $\log q$). This is the strategy that was implemented in [BCR10] and which is described in [BCR11].

First we work with an extension of degree d such that the points of the possibles kernels live. Since we know χ_π, a bound on d is given by the order of X in $(\mathbb{Z}/\ell\mathbb{Z})[X]/\chi_\pi$, but we can get a better bound as follow: if K is totally maximal isotropic and rational, then since π stabilize K its characteristic polynomial P of its restriction to K is of degree g and divides χ_π. But since K is isotropic and π is q-symplectic, we have that $\chi_\pi(X) = P(X)X^g P(q/X)$. So we can restrict our search of rational divisors of χ_π to polynomials P satisfying this property. This gives us an extension of degree d bounded by $\ell^g - 1$.

Next we compute a basis of $A[\ell](\mathbb{F}_{q^d})$. Here we make the assumption that we have an algorithm to take an almost uniform random point in $A(\mathbb{F}_{q^d})$ (if A is a Jacobian it suffices to sum $g+1$ points on the curve). Since we have χ_π we know $\#A(\mathbb{F}_{q^d})$, so multiplying by the cofactor c, we get a (almost uniform) point in the ℓ-primary component $A[\ell^\infty](\mathbb{F}_{q^d})$. Since $\#A(\mathbb{F}_{q^d}) = O(q^{dg})$, this step takes $O(gd \log q)$ arithmetic operations in $A(\mathbb{F}_{q^d})$. We are interested in the asymptotic with respect to ℓ, so this is $\widetilde{O}(d^2 \log^2 q)$ binary operations.

Now one needs to be careful that if we have a random point P in the ℓ-primary component of order ℓ^ν, multiplying by $\ell^{\nu-1}$ to get a point of ℓ-torsion is not uniform. For instance if the ℓ-primary component is generated by P_1 of order ℓ^2 and P_2 of order ℓ, a random point P is of the form $\lambda P_1 + \mu P_2$ and is of order ℓ^2 if $\lambda \neq 0$, so ℓP is a multiple of ℓP_1 and we almost never recover multiples of P_2 this way. This is easy to correct: assume we have already sampled P_1 and we sample P as above. We do a DLP to get $\ell P = \lambda_1 \ell P_1$ where $\lambda = \lambda_1 + \ell\lambda_2$. Then we define $P' = P - \lambda_1 P_1$, this is a point of ℓ-torsion, and $(\ell P_1, P')$ gives a basis of the ℓ-torsion.

Let's formalize this as follow:

Lemma 5.6.1. *If G is a finite ℓ-primary commutative group of order ℓ^N, we say that a system of generators $P_1, \dots, P_m$ is minimal if P_i is of order ℓ^{n_i} and the application $\prod \mathbb{Z}/\ell^{n_i} \to G, \lambda_i \mapsto \sum \lambda_i P_i$ is a bijection. Since the application is surjective by hypothesis this is equivalent to $\ell^N = \prod \ell^{n_i}$, and if $P'_i = \ell^{n_i - 1}$ this is equivalent to the P'_i being a basis of the $\mathbb{Z}/\ell\mathbb{Z}$ vector space $G[\ell]$. So m is the rank of G.*

This gives the following algorithm to construct a minimal basis of G: we sample a random point P_1, this is a minimal generator of $H = \langle P_1 \rangle$. Assume we have constructed minimal generators $H = \langle P_1, \dots, P_k \rangle$. We sample a new random point P_{k+1} in G, of order $\ell^{n_{k+1}}$. We let $P'_{k+1} = \ell^{n_{k+1}-1} P_{k+1}$. Either P'_{k+1} is not in the linear span of the P'_i, $i \leq k$ and we have minimal generators of $H' = \langle P_1, \dots, P_{k+1} \rangle$ or we have a linear relation $\sum \lambda_i P'_i = 0$. We then use this relation to correct P_j where j is such that P_j is the point of minimal order such that $\lambda_j \neq 0$. We iterate again with this new P_j, updating a new point again if needed (this terminates because the new P_j is of order strictly less than the old one by construction), until we get a minimal system of generators of $H + \langle P_{k+1} \rangle$. We refer to [BCR11, § 3.1] for more details on this algorithm.

This gives a randomized algorithm which needs to sample $O(m)$ uniform points in G, perform $O(m)$ scalar multiplication for a total cost of $O(mN\ell)$ operations in G, and perform $O(m)$ search of a linear relation between points of ℓ-torsion. If G is represented by an abstract group, as is our case of $G = A(\mathbb{F}_{q^d})[\ell^\infty]$ which is of rank at most $2g$, searching for linear relations costs $\widetilde{O}(\ell^{m/2})$ using a BSGS algorithm. For our G the total cost is then $O(d \log q + \ell^g)$ operations in $A(\mathbb{F}_{q^d})$. In practice with our experiments from [BCR10] on abelian surfaces with ℓ of a few hundred, finding linear relations is the most expansive step. So it is worthwhile to speed it up.

There are two possible improvements for our particular G. The first is to use the Weil pairing e_ℓ, we can then do DLPs in μ_ℓ instead to get our linear relations, this costs $O(\sqrt{\ell})$ (of course there is no point in using a subexponential DLP algorithm for μ_ℓ here). The worst case is when our first sampled points give a group H such that $H[\ell]$ is isotropic for the Weil pairing (this will be the case if $\mu_\ell \not\subset \mathbb{F}_{q^d}$). In this case the Weil pairing cannot help us,

and we are reduced to doing BGSG in $H[\ell]$, which is of rank at most g. So we get an algorithm which costs $O(d\log q+\log\ell+\ell^{g/2})$ operations in $A(\mathbb{F}_{q^d})$.

The other possibility, if $\mu_\ell\subset\mathbb{F}_{q^d}$, is to use the Tate pairing on $A[\ell](\mathbb{F}_{q^d})\times A(\mathbb{F}_{q^d})/\ell A(\mathbb{F}_{q^d})$. We sample $O(g)$ extra points in $A(\mathbb{F}_{q^d})$ to have a good probability to have a system of generators of $A(\mathbb{F}_{q^d})/\ell A(\mathbb{F}_{q^d})$, then we use the Tate pairing and DLP in μ_ℓ to decompose our points of ℓ-torsion along our current generators. Note that if d is such that μ_ℓ is not in $\mathbb{F}_{q^d}$, we need to take a further extension $d'=O(\ell d)$ to apply this strategy: we sample extra points in $A(\mathbb{F}_{q^d}(\mu_\ell))$ to compute the Tate pairing with the points sampled in $A(\mathbb{F}_{q^d})[\ell]$. The full algorithm costs $O(d\log q+\ell^{1/2})$ operations in $A(\mathbb{F}_{q^d})$ along with $O(1)$ Tate pairings over $A(\mathbb{F}_{q^d}(\mu_\ell))$. So essentially this costs $O(d'\log q\log\ell+d^2\log^2 q+d\log q\ell^{1/2}))$ binary operations.

Then once we have a basis of $A(\mathbb{F}_{q^d})[\ell]$ we may look for rational isotropic kernels (using the methods above to construct a partial symplectic basis and computing the matrix of the Frobenius acting on our basis), and compute isogenies (costing $O(\ell^g)$ operations in $A(\mathbb{F}_{q^d})$) for each of them. This dominates the complexity above, except the sampling in $A(\mathbb{F}_{q^d})[\ell^\infty]$.

Summing up the full cost, the algorithm thus requires $\widetilde{O}(d^2\log^2 q+X\ell^g d\log q)$ operations to compute X isogenies, with d bounded by $O(\ell^g)$, along with some precomputations which do not depend on ℓ like computing χ_π (for small ℓ we could also compute a triangular representation as in Section 5.5 but this of course is polynomial in ℓ).

To compute cyclic isogenies, we have a similar strategy to compute a basis of $A[\beta]$ and then iterate through the kernels and the isogenies. In this case, we need to be able to compute the real multiplication (namely the action of ℓ/β) to sample points in the β-primary component. More precisely, there should be an algorithm in $O(\log N(\alpha))$ to evaluate multiplication by α. See Section 4.5 for more on computing real endomorphisms. We then get a complexity of $\widetilde{O}(d^2\log^2 q+X\ell d\log q)$ to evaluate X isogenies, where $d=O(\ell^2)$ if $\ell=N(\beta)$.

5.6.3 *Type of ℓ-isogenies for abelian surfaces*

We give a partial study of the number of rational isotropic kernels in $A[\ell]$ according on how $\chi_\pi \mod \ell$ split for abelian surfaces over $\mathbb{F}_q$. This could be used to get an Atkin-type point counting algorithm using the splitting behaviour of Siegel modular polynomials in dimension 2, as was done in the (much easier) case of Hilbert-modular polynomial [BGG+17]. Here we only focus on the number of rational roots.

First we have seen that if the splitting of χ_π is of type $(4),(1,3),(1,1,2)$ there is no kernel, because $\chi_\pi(X)$ cannot be written as $P(X)q^2P(q/X)$.

- In the case $(2,2)$, $\chi_\pi=Q_1Q_2$ with Q_i irreducibles, and Q_2 the q-reciprocal of Q_1. Then if $Q_1\neq Q_2$, π is cyclic over $A[\ell]$, there are two stables subspaces given by $\operatorname{Ker}Q_i$ and they are isotropic by [Rob21, Lemma 4.1.6]. $A[\ell]=\operatorname{Ker}Q_1\oplus\operatorname{Ker}Q_2$ is a symplectic decomposition.

  If $Q_1=Q_2=Q$, then if the minimal polynomial is Q^2, π is cyclic, the only stable subspace is $\operatorname{Ker}Q$ which is isotropic by [Rob21, Lemma 4.1.6] since $\operatorname{Ker}Q=\operatorname{Im}Q$. Otherwise the minimal polynomial is Q, so we may see $A[\ell]$ as an $\mathbb{F}_\ell[\pi]=\mathbb{F}_\ell[X]/Q(X)$-vector space of dimension 2. There are $((\ell^2)^2-1)/(\ell^2-1)=\ell^2+1$ stable subspaces. There can be several cases for the number of isotropic kernels.

- In the case $(1,1,1,1)$, there are four eigenvalues $\lambda_1,\lambda_2,\mu_1,\mu_2$ with $\lambda_2=q/\lambda_1,\mu_2=q/\mu_1$.
  - If they are all distincts, π is cyclic, there are 4 isotropic invariant subspaces of dimension 2 corresponding to the two symplectic decompositions $A[\ell]=\langle\lambda_1,\mu_1\rangle\oplus\langle\lambda_2,\mu_2\rangle$ and $A[\ell]=\langle\lambda_1,\mu_2\rangle\oplus\langle\lambda_2,\mu_1\rangle$.
  - If $\lambda_1=\lambda_2=\lambda,\chi_\pi=(X-\lambda)^2(X-\mu_1)(X-\mu_2)$. By [Rob21, Lemmas 4.1.5 and 4.1.6], there is a symplectic basis (e_1,e_2,f_1,f_2) where $\langle e_1,f_1\rangle=\operatorname{Ker}(X-\lambda)^2,\langle e_2\rangle=\operatorname{Ker}(X-\mu_1),f_2=\operatorname{Ker}(X-\mu_2)$.

    If the minimal polynomial is of degree 4, then π is cyclic and there are two stable isotropic kernels given by $\operatorname{Ker}(X-\lambda)(X-\mu_1)$ and $\operatorname{Ker}(X-\lambda)(X-\mu_2)$.

    Otherwise π is diagonalisable, there are $2(\ell+1)$ stable subspaces of dimension 2 (take for generators any element of $\langle e_1,f_1\rangle$ along with e_2 or f_2), which given the symplectic basis are all isotropic.
  - If $\lambda_1=\mu_1,\chi_\pi=(X-\lambda_1)^2(X-\lambda_2)^2$. The characteristic subspace of λ_i is isotropic, so we have a symplectic decomposition $A[\ell]=\operatorname{Ker}(X-\lambda_1)^2\oplus\operatorname{Ker}(X-\lambda_2)^2$.

    If the minimal polynomial is χ_π, π is cyclic, and the stable subspaces of dimension 2 are $\operatorname{Ker}(X-\lambda_1)(X-\lambda_2)$, $\operatorname{Ker}(X-\lambda_1)^2$, $\operatorname{Ker}(X-\lambda_2)^2$, which are all isotropic.

If the minimal polynomial is $(X-\lambda_1)^2(X-\lambda_2)$, then $\mathrm{Ker}(X-\lambda_1)^2$ and $\mathrm{Ker}(X-\lambda_2)$ are isotropic. Taking an adapted symplectic basis, we see there is a third isotropic kernel given by the eigenvector for λ_1 and its unique orthogonal λ_2 eigenvector.

If the minimal polynomial is $(X-\lambda_1)(X-\lambda_2)$, π is diagonalisable. We have a symplectic decomposition $A[\ell] = (\mathrm{Ker}\, X-\lambda_1) \oplus (\mathrm{Ker}\, X-\lambda_2)$, and taking an adapted symplectic basis we see that there are $\ell+1$ other isotropic kernels (since $\pi_{|K}$ has to be diagonal): take an eigenvector for λ_1 and its unique orthogonal λ_2 eigenvector.

- Finally if $\lambda_1=\lambda_2=\mu_1=\mu_2=\lambda, \chi_\pi=(X-\lambda)^4$.

  If the minimal polynomial is of degree 4, π is cyclic and there is a unique stable subspace of dimension 2, $\mathrm{Ker}(X-\lambda)^2$ which is isotropic.

  If the minimal polynomial is of degree 3, we have a vectorial decomposition $A[\ell] = C((X-\lambda)^3) \oplus C(X-\lambda)$ where $C(P)$ is the companion matrix of P. There are $\ell+1$ stable subspaces K where $\pi_{|K}$ is diagonal, and ℓ stable subspaces K where $\pi_{|K}$ is of type $C((X-\lambda)^2)$ (taking the $\mathbb{F}_q[\pi]$ span of $v+w$ where v is a generator of the $C((X-\lambda)^2)$ subspace of $C((X-\lambda)^3)$ and w an element of $C(X-\lambda)$). I don't know how many are isotropic.

  If the minimal polynomial is of degree 2, and we have a vectorial decomposition $A[\ell] = C((X-\lambda)^2) \oplus C((X-\lambda)^2)$, then $\mathrm{Ker}(X-\lambda) = \mathrm{Im}(X-\lambda)$ so is isotropic. Completing a basis e_1, e_2 of $\mathrm{Ker}(X-\lambda)$ with a symplectic basis f_1, f_2, the $k[\pi]$-space spanned by f_i is of type $C((X-\lambda)^2)$, $i=1,2$. The other stable subspaces of dimension 2 are of type $C((X-\lambda)^2$, so are cyclic and spanned by π-linear combinations of f_1, f_2 (which are not all divisible by $\pi-\lambda$). There are $\ell^4-\ell^2$ such combination, which give $(\ell^4-\ell^2)/(\ell^2-\ell) = \ell(\ell+1)$ stable subspaces. I don't know how many are isotropic.

  If $A[\ell] = C((X-\lambda)^2) \oplus C((X-\lambda)) \oplus C((X-\lambda))$ we can hold a similar reasoning.

  Finally if the minimal polynomial is of degree 1, every subspace is stable and there are $\ell^3+\ell^2+\ell+1$ isotropic kernels.

5.6.4 *The structure of the ℓ-isogeny graph of ordinary abelian surfaces*

We briefly detail the structure of the isogeny graph in the case of ordinary abelian surfaces over $\mathbb{F}_q$. This was a joint work with Ionica, Martindale and Streng [IMR+14] which was not published, because our results were partly recovered in [BJW17] looking at Tate modules instead.

The assumption that $A/\mathbb{F}_q$ is ordinary allows to lift to characteristic zero and use the theory of complex multiplication. Also the case $g=2$ helps a lot because $\mathrm{End}(A)$ is a CM quartic order. So its real suborder $\mathrm{End}^s(A)$ is an order O in a real quadratic field K. But quadratic orders are Gorenstein: its dual for the trace $O^\star$ is principal. Since for a fractional ideal I we have $II^\star = R(I)^\star$ where $R(I)$ is the order associated to I, this means that I is always invertible for its associated order. This simplifies a lot the study of O-modules such as the lattices corresponding to the lifted abelian surfaces with real multiplication by O.

We know that $\mathrm{End}(A) \supset \mathbb{Z}[\pi, \overline{\pi}]$, so $\mathrm{End}^s(A) \supset \mathbb{Z}[\pi+\overline{\pi}]$. When looking at the quadratic real orders that may appear in the isogeny graph, we may label them as O_1 (locally maximal at ℓ), O_ℓ of index ℓ in O_1, O_{ℓ^k} and so on. So we may decompose the isogeny graphs into "pancakes", one pancake for each order, and look at how isogenies move inside a pancake and across a pancake (we say that it is RM ascending if the real order increases and RM descending if the real order decreases).

Over O_1 if ℓ splits into totally positive elements as $\beta\beta^\sigma$, it is easier to understand the graphs by looking at the β and β^σ isogeny graphs. These are volcanoes [IT14], exactly as in the elliptic curve case, and a ℓ-isogeny between abelian surfaces with real multiplication by O_1 is the composition of a β-isogeny followed by a β^σ-isogeny.

More precisely, we have $\mathrm{End}(A) = O_1 + \mathfrak{f}O_L$ where $L = \mathrm{End}^0(A)$ is the full CM field, O_L is its maximal order and the O_1-ideal $\mathfrak{f}$ is the conductor of $\mathrm{End}(A)$ over K. Then

- If $\mathfrak{f}$ is prime to β, there are 2, 1, or 0 horizontal-isogenies according to whether β splits, is ramified or is inert in $\mathrm{End}(A)$, and the rest are descending to $O_1 + \mathfrak{f}\beta O_L$;
- If $\mathfrak{f}$ is not prime to β there is one ascending isogeny (to $O_1 + \mathfrak{f}/\beta O_K$) and ℓ descending ones;

- We are at the bottom when the β-valuation of $\mathfrak{f}$ is equal to the valuation of the conductor of $\mathbb{Z}[\pi, \overline{\pi}]$. Then there is one ascending isogeny.

In O_ℓ there are no β-isogenies (if we want principally polarised abelian surfaces). Ascending RM isogenies, that is ℓ-isogenies going from O_ℓ to O_1 are over $\mathbb{C}$ of the form

$$\mathbb{C}^g/(O_\ell \oplus O_\ell^\vee \tau) \to \mathbb{C}^g/(O_1 \oplus O_1^\vee \tau),$$

(we can check that this is indeed an ℓ-isogeny). So $\mathrm{Sl}_2(O_1 \oplus O_1^\vee)/\mathrm{Sl}_2(O_\ell \oplus O_\ell^\vee)$ acts on such isogenies.

We can use this action to check how many ℓ-isogenies descend from the O_1 pancake to the O_ℓ one (here there are no rationality considerations). When ℓ splits in O_1, $\mathrm{Sl}_2(O_1 \oplus O_1^\vee)/\mathrm{Sl}_2(O_\ell \oplus O_\ell^\vee) \simeq \mathrm{Sl}_2(O_1/\ell O_1)/\mathrm{Sl}_2(O_\ell/\ell O_1) \simeq \mathrm{SL}_2(\mathbb{F}_l^2)/\mathrm{Sl}_2(\mathbb{F}_l) \simeq \mathrm{Sl}_2(\mathbb{F}_l)$, so we find $\ell^3 - \ell$ ℓ-isogenies changing the real multiplication. On the other hand we have seen above that there is $(\ell+1)^2$ ℓ-isogenies preserving the real multiplication In total we find all $\ell^3 + \ell^2 + \ell + 1$ ℓ-isogenies.

If ℓ is inert, we find $\#\,\mathrm{SL}_2(\mathbb{F}_{\ell^2})/\mathrm{SL}_2(\mathbb{F}_\ell) = \ell^3 + \ell$ RM-descending isogenies, while there are ℓ^2+1 (the degree of the Hilbert ℓ-modular polynomial) RM-horizontal isogenies. If ℓ is ramified, we find $\#\,\mathrm{SL}_2(\mathbb{F}_\ell[\epsilon])/\mathrm{SL}_2(\mathbb{F}_\ell) = \ell^3$ RM-descending isogenies, while there are indeed $(\ell+1)\ell + 1 = \ell^2 + \ell + 1$ RM-horizontal isogenies.

In O_ℓ, we find $\mathrm{Sl}_2(O_\ell \oplus O_\ell^\vee)/\mathrm{Sl}_2(O_{\ell^2} \oplus O_{\ell^2}^\vee) \simeq \mathrm{Sl}_2(O_\ell/\ell O_\ell)/\mathrm{Sl}_2(O_{\ell^2}/\ell O_\ell) \simeq \mathrm{SL}_2(\mathbb{F}_l[\epsilon])/\mathrm{Sl}_2(\mathbb{F}_l)$, so there are ℓ^3 RM-descending isogenies. There is only one RM-ascending isogeny, so there are $\ell^2 + \ell$ RM-horizontal isogenies.

In summary (see [Rob15]):

- On O_1:
  - If ℓ is split there are $\ell^2 + 2\ell + 1$ RM-horizontal ℓ-isogenies and $\ell^3 - \ell$ RM-descending ℓ-isogenies;
  - If ℓ is inert there are $\ell^2 + 1$ RM-horizontal ℓ-isogenies and $\ell^3 + \ell$ RM-descending ℓ-isogenies;
  - If ℓ is ramified there are $\ell^2 + \ell + 1$ RM-horizontal ℓ-isogenies and ℓ^3 RM-descending ℓ-isogenies;
- If O is not maximal at ℓ, there are 1 RM-ascending ℓ-isogeny, $\ell^2 + \ell$ RM-horizontal ℓ-isogenies and ℓ^3 RM-descending ℓ-isogenies.

We finish by some examples of isogeny graphs of abelian surfaces along with the corresponding endomorphisms orders in Figures 5.1 and 5.2.

5.6.5 *The structure of isogeny graphs of products of elliptic curves*

When exploring a full isogeny graph (meaning allowing isogenies of various degrees), it is interesting to first have a formal description of this graph, then from it to compute the isogenies of minimal degrees that span the graph, and then apply the methods of Sections 5.6.1 and 5.6.2 to compute it. For instance when exploring a graph of abelian varieties over $\mathbb{F}_q$ with maximal complex multiplication, the structure of the isogeny graph is given by the action of the Shimura class group, see Chapter 7. For ordinary abelian varieties over $\mathbb{F}_q$, we can use Deligne's equivalence of categories [DM69], further studied in [How95].

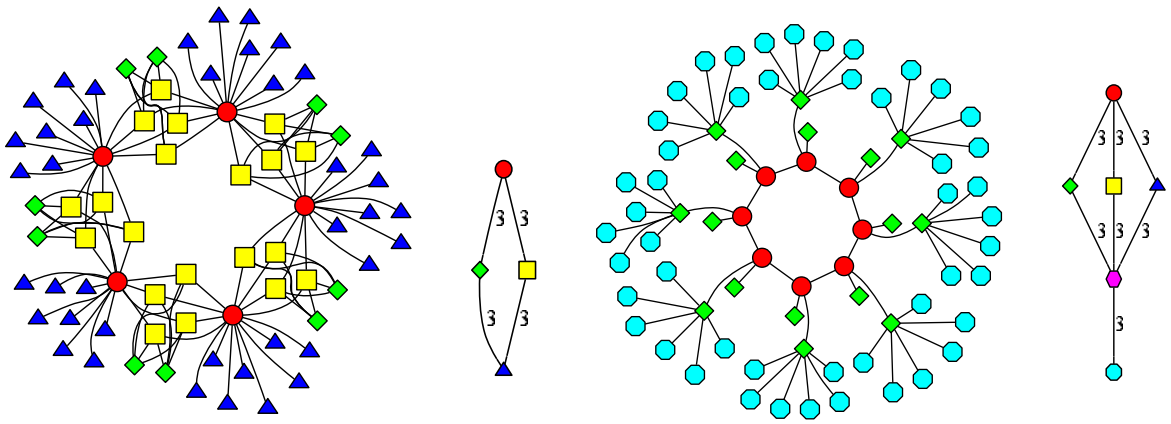
In this section we look at the description of the isogeny class of a product of g elliptic curves. A formal description of it and a way to make it effective are the core results of [KNR+20b]. This isogeny class is particularly useful when looking for curves with maximal number of points. Indeed looking at the zeta function of such a curve shows that its Jacobian J is then isogenous to E^g where E is an elliptic curve with a maximal number of points. So if we can describe the isogeny class of E^g, we can try to see if an isogenous $A/\mathbb{F}_q$ is the Jacobian of a curve defined over $\mathbb{F}_q$.

Beware that if $A/\overline{\mathbb{F}}_q$ is a Jacobian of a curve C over the algebraic closure with associated principal polarisation Θ and (A, Θ) descends to $\mathbb{F}_q$, then C descends to $\mathbb{F}_q$ but if C is not hyperelliptic A may not be the Jacobian of C over $\mathbb{F}_q$. Indeed the Jacobian $J(C)/\mathbb{F}_q$ may only be a quadratic twist of $A/\mathbb{F}_q$. We refer to [Ser01] for a beautiful exposition of this. If $A/\mathbb{F}_q$ has a maximal number of point but is a quadratic twist of $J(C)$, then $J(C)$, hence C, has a minimal number of points. It is shown in [ZLR10; Rit10] that in dimension 3 this quadratic obstruction to the descent of $A = \mathrm{Jac}(C)$ as a Jacobian over $\mathbb{F}_q$ can be measured by whether the modular form $\chi_{18}(A) \in \mathbb{F}_q$ is a square or not (it is 0 if and only if C is hyperelliptic or A is not absolutely simple).

Let us now describe the isogeny class of E^g. If $E/\mathbb{F}_q$ is an elliptic curve and $R \subset \mathrm{End}(E)$, Serre introduced two ways to define a functor between abelian varieties isogenous to a product of E and finite R-modules.

If M is a finite R-module, and we take a partial resolution $R^m \to R^n \to M \to 0$ of M, we may define $M \otimes_R E$ as the cokernel $E^m \to E^n \to M \otimes_R E \to 0$ and $\mathrm{Hom}_R(M, E)$ as the kernel $0 \to \mathrm{Hom}_R(M, E) \to E^n \to E^m$. (If R

FIGURE 5.1: ℓ-isogeny graphs of abelian surfaces



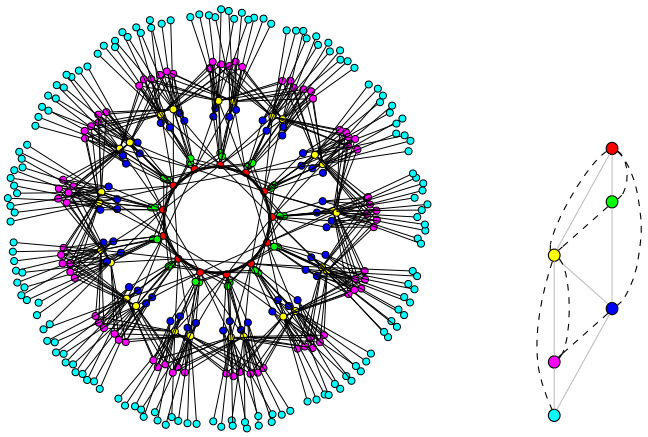
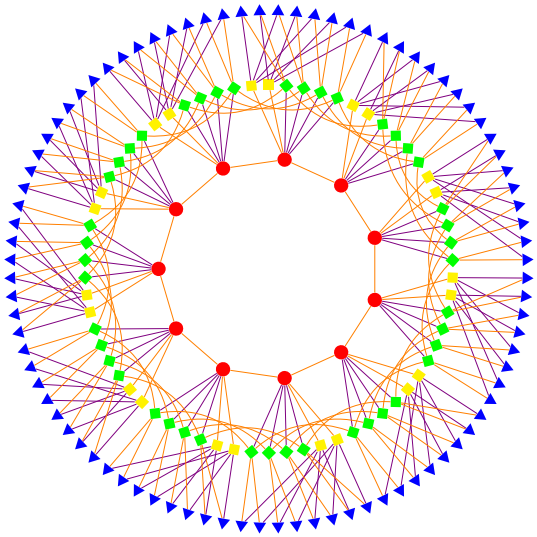
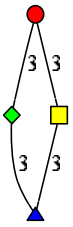


FIGURE 5.2: Cyclic isogeny graph of abelian surfaces



β_1 is inert and β_2 is



split in K.

is not commutative, for $M \otimes E$ we take M a right R-module and for $\mathrm{Hom}_R(M,E)$ a left R-module.) In the other direction, to such an abelian variety A we may associate the R-modules $\mathrm{Hom}(E,A)$ and $\mathrm{Hom}(A,E)$ respectively.

We may also introduce $\mathrm{Tor}_R^1(M,E)$ and $\mathrm{Ext}_R^1(M,E)$ functors the usual way. For instance, the resolution above gives rise to $0 \to \mathrm{Tor}_R^1(M,E) \to E^m \to E^n \to M \otimes_R E \to 0$. As an example, if $R = \mathbb{Z}$ and $M = \mathbb{Z}/n\mathbb{Z}$, then $M \otimes_R E = 0$, and $\mathrm{Tor}_R^1(M,E) = \mathrm{Hom}_R(M,E) = E[n]$. See also [Wat69, Proposition A.3] for a characterisation of $\mathrm{Ext}_R^1(R/I,A)$ for A an abelian variety with $R = \mathrm{End}(A)$. If $R = \mathrm{End}(E) = \mathbb{Z}[\pi]$ is a commutative maximal order, Serre proves that $M \mapsto M \otimes_R E$ is an equivalence of category between abelian varieties isogenous to E^g and torsion-free R-modules (ie projective modules since R is Dedekind).

Although it is contravariant, the function $\mathrm{Hom}_R(\cdot,E)$ is often more convenient. Indeed if $R = \mathrm{End}(E)$, this functor restricted to the finitely presented torsion-free left R-module is fully faithful and exact and has for quasi-inverse $A \mapsto \mathrm{Hom}(A,E)$ on its image by [JKB+18, Theorem 4.4 and Theorem 4.8]. It is an equivalence of category (to the abelian varieties isogenous to E^g) if E is ordinary and $R = \mathrm{End}(E) = \mathbb{Z}[\pi]$ [JKB+18, Theorem 1].

Under this last condition, in [KNR+20b, § 3.2] we exploit that $\mathrm{Hom}_R(\cdot,E)$ is compatible with duality to give an equivalence of categories between polarised abelian varieties isogenous to E^g and Hermitian R-lattices of rank g. In [KNR+20b, § 3.3] we explain how to recover the kernel of the isogeny $E^g \to A$ corresponding to such an Hermitian R-lattice. Then in [KNR+20b, § 4] we explain how to compute this isogeny via theta functions. We describe in [KNR+20b, § 2] an algorithm to enumerate the equivalence class of all Hermitian R-lattices (we do not assume R maximal which complicates this enumeration), so we have a full description of the isogeny class.

Since we have an explicit formula for χ_{18} in terms of the theta constant, this allows to find maximal curves in genus 3. There is a fun little trick: since the Theta isogeny algorithm of Chapter 4 requires a symmetric theta structure, we need to take an extension $\mathbb{F}_{q^d}$ of $\mathbb{F}_q$ to compute the isogeny. It may seem that we first need to descend the isogenous A back to $\mathbb{F}_q$ to be able to compute the true value of $\chi_{18}(A)$. But $\chi_{18}(A)$ really depends on the choice of a differential basis w_A of A (see also Section 2.9.3), so even if we only have A defined over $\mathbb{F}_{q^d}$, we only need to be sure we evaluate χ_{18} at a differential basis w_A which descends to $\mathbb{F}_q$. Concretely, we use the affine version of the isogeny theorem (ie the modular version), and we have seen in Section 4.6 how to keep track of the differentials. In particular for an isogeny $A \to B$, if we start with an affine theta lift of the theta constants induced by a rational differential w_A, then on B we have affine lifts of r-fold product of theta constants induced by w_B such that $f^* w_B = w_A$, so w_B is rational when w_A is so. So we only need to be sure that we use an affine (ie modular) version of Thomae's formula which gives the affine lift of the theta null point of E corresponding to the rational differential dx/y and this is explained in [KNR+20b, § 4.4]. Since w_B is rational, this allow us to compute rational modular forms $g(B, w_B)$ (if g is defined over the base field). Likewise in [KNR+20b, § 5.3] we explain how to compute Schottky's modular form (which identify Jacobians in dimension 4) algebraically.

There are two inconveniences of our algorithm: first we have seen in Section 4.6 that we only have for now a modular interpretation of the isogeny formula which uses Koizumi's descent (ie the matrix version). So we cannot yet use the faster descent which gives faster isogeny computations (in particular when $r = 4$). Secondly, we recall that we have $r = 2$ if ℓ is a sum of two squares, and $r = 4$ otherwise. But in the latter case, since we only compute r-fold products of theta constant, we cannot express a modular form of odd weight this way (since the thetas are of weight $1/2$). More generally: we can only evaluate modular forms that are expressed in terms of r-fold product of thetas. Of course, by a suitable normalisation afterwards, rather than computing all r-fold products in the isogeny formula, it is enough to just compute the products of the form $\theta_i(0)\theta_0(0)^{r-1}$ in the isogeny algorithm, see [KNR+20b, p. 26]. Hopefully the new descent algorithm will also allow to compute modular forms of odd weight when $r = 4$.

5.7 CONCLUSION AND PERSPECTIVES

Fast evaluation of theta functions

As we have seen, a key tool in computing modular polynomials in dimension $g = 1, 2$ is fast evaluation of the theta constants $\theta_i(0,\tau)$ and period matrices, using the algorithms from [Dupo6]. As explained in Section 5.3.4, these algorithms have been extended in [Lab16] to fast algorithms to compute the theta functions $\theta_i(z,\tau)$ in z and τ. There are still some heuristics remainings, even for computing theta constants, but good progress has been made on some of them in [Kie20b; Kie20c].

A big challenge is to extend these computations to genus $g > 2$. One key question is the choice of signs in the duplication formula. First, one has to make an algebraic choice, ie one which do correspond to a symmetric theta structure, see Section 2.13. But for the convergence, we need more: namely the "good" choice corresponding

exactly to $\tau \mapsto 2\tau$ (and not one of the other $2\gamma\tau$ where $\gamma \in \Gamma/\Gamma(2,4)$).The problem gets worse for computing the period matrix: here we cannot rely on small precision computations to get the correct choice of sign, except when $A = \mathrm{Jac}(C)$ is the Jacobian of an (hyperelliptic) curve where we can do small precision period integral evaluation [MN17a; MN17b], see [Lab16, Chapter 8].

When $g = 2$, the topological choice of sign, at least when τ is in the standard fundamental domain, corresponds to a "good choice of roots" [Dup06, Chapitre 7]. This is proven for the fundamental domain in [Dup06], but the algorithm actually needs this for a larger domain (to recover the explicit action of some specific matrices on τ in order to then recover τ, as explained in Section 5.3.2), and this has been proved in [Kie20c].

Another approach is as follow: for application to isogenies and point counting in Sections 5.4 and 5.5 we have seen that we need the derivative $(\theta_i(\tau)/\theta_0(\tau))'$ of the theta constants (see Key Idea 6). Since we have them, we might as well push them through the duplication formula (the duplication formula relates normalised isogenies, so we can control how the basis of differential are transformed). In other words, rather than simply looking at how the modular forms θ_i behave when τ goes to infinity, we can also look at how the derivatives $(\theta_i/\theta_0)'$ behave. This has the advantage that we can directly recover $\mathrm{Sym}^2(c\tau + d)$ rather than just $\det(c\tau + d)$, hence this requires only two calls to the AGM (keeping track of the derivative) rather than severals. This should also help in the choice of sign, and to extend the algorithm in higher dimensions.

Like for the computation of theta constants, I hope that looking at the derivatives of the thetas and pushing these through the duplication formula to get more informations from only one AGM iteration will also allow to generalize the computation of the $\theta_i(z,\tau)$ and its converse to higher dimensions g, without computing at small precision to get the correct signs.

Modular correspondances and evaluating modular polynomials

I also plan to look in more details the p-adic and CRT based algorithms of Section 5.3.8 for computing evaluated modular polynomial. It would be nice to have a quasi-linear algorithm in all dimension, following one of the strategies of Conjecture 5.3.14.

For the modular correspondances from Section 5.2, using the theta model, we have already seen that the main question is to find equations for the image. An exception to this is for modular correspondances of the form $\mathcal{A}_{g,\ell n} \to \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$ when $\ell \mid n$, in this case it is easier to find equations, see for instance **??** 6.3.1.(ii).

Still the modular correspondance should provide enough information to study Hecke correspondances. This requires finding points in $\mathcal{A}_{g,\ell n}$, but we can do so either from starting with a point of $\mathcal{A}_{g,n}$ and computing the ℓ-torsion (we will come back to this in Example 6.3.1), or simply by looking at product of smaller dimensional abelian varieties, for instance taking $A = E^g$, E an elliptic curve (it is easier to build the level $n\ell$ theta null point for an elliptic curve than for a higher dimensional abelian variety, and then it suffices to take the Segre embedding). Of course this latter approach only give special points, but using deformation afterwards should give us enough informations about the modular correspondance. We plan to apply this to study p-adic Hecke-correspondances with Caruso and Lubicz.

Evaluation of modular forms and computing modular correspondances of higher levels

A related problem is to construct explicit birational models of Siegel and Hilbert moduli with other level structures than the $\Gamma_0(\ell)$-ones, eg $\Gamma_1(n)$ and $\Gamma(n)$, and the maps between them. In other words: construct the minimal polynomial (or a tower of field extension) over $k(\mathcal{A}_g)$ of modular forms of the corresponding levels. This is strongly related to constructing moduli for polarisations of type $\delta = (\delta_1, \dots, \delta_g)$. The analytic evaluation/interpolation approach requires computing period matrices from the modular values, and conversely, we refer to the end of Section 5.3.4 for an approach.

The CRT approach and p-adic approach should be straightforward to generalise: in the first case we simply build up all level structures modulo small primes, and in the second case we do so over one prime p_0, and then lift the structures to sufficient p_0-adic precision (this should work well as long as the level structure is étale over the base field).

Applications of modular polynomials to isogenies, point counting, and isogeny graphs

One major goal will also be to extend the computation of isogenies from modular polynomial done in Section 5.4. The algorithmic outline in [KPR20] is sufficiently general that to apply it to higher dimensional abelian varieties we only need an explicit version of the Kodaira-Spencer isomorphism. Of course this depends on the models of abelian

varieties we have, the modular invariants we use, and how we encode our basis of differential. In [KPR20] we use Jacobians of hyperelliptic curves $C : y^2 = f(x)$ of genus 2, the basis of differential $dx/y, xdx/y$ being implicitly encoded by the choice of the equation of C. The main work would be to extend this to higher dimensional Jacobians. For instance, for genus 3 we could try use the theory of concomitants of ternary quadrics [CFG20]. A somewhat orthogonal but interesting question would also be to express via the Kodaira-Spencer isomorphism the tangent space at $\mathrm{Jac}(C)$ of the moduli of curves (inside $T_{\mathrm{Jac}(C)}(\mathcal{A}_g)$), respectively of hyperelliptic curves if C is hyperelliptic.

The advantage of Jacobians is that we can express the differential equation expressing the isogeny at the level of the curve C, as in Section 4.7, hence work with a power series ring of one variable rather than g. Finally, for abelian varieties, we can of course always use the theta model. We then have explicit projective equations of the abelian variety by Theorem 2.7.4, so we can use differentials given by the $d\theta_i$. The Kodaira-Spencer isomorphism can then be recovered, for theta functions, by the heat equation (see [Cvoo, p. 9; Per16, Eq 18] for an algebraic, respectively an analytic, approach to the heat equation), along with some linear algebra to express this in terms of the $d(\theta_i/\theta_0)$:

$$2\pi i(1+\delta_{jk})\frac{\partial\theta_i}{\partial\tau_{jk}} = \frac{\partial^2\theta_i}{\partial z_j \partial z_k}.$$

This allows to express the derivative (in τ) of the modular thetas in terms of a basis of $\mathrm{Sym}^2\,\Omega_0(A)$ induced by the derivative (in z) of the theta coordinates. For more general modular invariants, if we have modular relations relating them to theta constants, we can then derive these relations to recover the derivative of the modular forms in terms of the derivative of the theta constants. Likewise, on a Jacobian $J = \mathrm{Jac}(C)$, we can relate the differentials on the curve with the derivatives of the theta coordinates using the generalised Thomae's algorithm from Chapter 2. (As remarked in Section 4.6, it would also be nice to have a way to evaluate algebraically for more general modular forms $\mathfrak{g}$ than dJ. For instance given the projective theta constants and a basis of differential given by some uniformisers in the Riemann relations, can we compute the affine lift of these constants induced by this differential?)

One of the main application is then point counting, as explained in Section 5.5. Of course, as explained in this Section, when $g \geq 3$, we only expect to gain compared to Schoof algorithm when we have real multiplication.

Finally, to explore isogeny graphs, by Section 5.6.4 it is clear that we need cyclic isogenies to fully explore them. But the (non polarised) Hilbert moduli decomposes under components with polarisations of type indexed by the narrow class group of the RM field K (see [Rob21, Sections 5.5 and 5.8] for more details). So if I is a fractional ideal of the real multiplication, then I-isogenies send principally polarised abelian varieties to abelian varieties with a polarisation of type $[I] \in \mathrm{Cl}^+(\mathcal{F})$, which is not principal unless $I = (\beta)$ with $\beta \gg 0$. So to fully explore isogeny graphs, we should construct modular polynomials between those components, even if we are only interested in principally polarised abelian varieties (this helps to decompose an isogeny into compositions of P_i-isogenies, P_i prime ideals of $\mathcal{O}_{\mathcal{F}}$). A question is then how to find good modular invariants on these components (eg induced by theta constants of mixed level δ).

6 CANONICAL LIFTS

CONTENTS

6.1 INTRODUCTION

In Section 6.2 we review Satoh's method (and following improvements) of using canonical lifts of elliptic curves for point counting. We explain how to adapt this to compute canonical lifts for abelian varieties in Section 6.3. We then give three different methods to do point counting (more precisely recovering χ_π): in Section 6.4 we lift modular forms, in Section 6.5 we lift modular functions along with the kernel of the Verschiebung and compute the normalised isogeny using Chapter 4, and finally in Section 6.6 we apply Section 5.4 instead, ie we lift derivative of modular invariants, which are vectorial modular forms of weight Sym^2. We give some perspectives in Section 6.7.

6.2 CANONICAL LIFTS AND POINT COUNTING

6.2.1 *Canonical lifts*

Let K be a p-adic field, $k = O_K/m$ its residue field. By Serre-Tate theory, deforming an abelian variety A/k is the same as deforming its p-divisible group $A(p)$. By Grothendieck-Messing this is the same as deforming the corresponding crystal $\mathbb{D}(A(p)/W(k))$.[1] This then becomes in principle a problem of linear algebra: deforming a crystal over a nilpotent divided power thickening (we assume for simplicity here that $p > 2$ so that $p^n/n!$ is nilpotent) is the same as deforming its Hodge filtration to an admissible filtration [Mes72, Theorem V.1.6]. We refer to [Rob21, Section 3.4] for more details.

We will work over $K = \mathbb{Q}_q$, the unramified extension of $\mathbb{Q}_p$ of residue field $\mathbb{F}_q$: $\mathbb{Q}_q = \mathrm{Frac}(\mathbb{Z}_q)$ with $\mathbb{Z}_q = W(\mathbb{F}_q)$ the ring of Witt vectors, and we will call σ the Galois action given by the lift of the (small) Frobenius π of $\mathbb{F}_q$ to $\mathbb{Q}_q$.

Given $A/\mathbb{F}_q$, we can thus control its deformations to $\mathbb{Z}_q/p^n\mathbb{Z}_q$. In particular, there is no obstruction to deforming A since there is none to deforming a p-divisible group by [Mes72][2], and we may glue deformations to $\mathbb{Z}_q/p^n\mathbb{Z}_q$

[1] Since we are over a perfect field, this crystal is essentially the corresponding Dieudonné module $\mathbf{D}(A(p))$, more precisely $\mathbf{D}(A(p)) = \mathbb{D}(A(p)/W(k))_{W(k)}$.

[2] Of course this can be proved directly for an abelian scheme, this is due to Grothendieck and Mumford, see [Gro62; DA70; MFK94, Proposition 6.7; Oor71, § 2.2, § 2.3, § 2.4].

together to form a formal abelian scheme $\widehat{A}$ over $\mathbb{Z}_q$. By Grothendieck's algebraicity theorem [GD64, § III.5.4.5], if $A/\mathbb{F}_q$ is separably polarised then we have effectivity [Oor71, § 2.4], ie $\widehat{A}$ is actually an abelian scheme $\widetilde{A}$ over $\mathbb{Z}_q$. By the theory of Néron models, $\widetilde{A}$ is then completely determined by its generic fiber $\widetilde{A}_{\mathbb{Q}_q}/\mathbb{Q}_q$ (since $\widetilde{A}$ is an abelian scheme it is the Néron model of its generic fiber).

We are mainly interested in the ordinary case. We recall that since k is perfect we have a (split) connected-étale exact sequence on $A(p)$. The étale part lifts canonically[3], and in the ordinary case the connected part is of multiplicative type, hence lifts canonically too (eg by taking the Cartier dual of the canonical lift of its étale Cartier dual). Letting $\widetilde{A}(p)^{\text{ét}}$ and $\widetilde{A}(p)^\circ$ be the lifts, lifting A then amount to finding an extension $0 \to \widetilde{A}(p)^\circ \to G \to \widetilde{A}(p)^{\text{ét}} \to 0$, ie corresponds to $\mathrm{Ext}^1(\widetilde{A}(p)^\circ, \widetilde{A}(p)^{\text{ét}})$ (G is then the p-divisible group of the lift $\widetilde{A}$). This gives rise to canonical coordinates on the moduli space [Mes72, Appendix; Kat81]. The 0 element, which corresponds to the unique extension which is still split, then gives the canonical lift.

By Serre-Tate theory and formal GAGA, the canonical lift is thus a fully faithfull functor $A/\mathbb{F}_q \mapsto \widetilde{A}_{\mathbb{Q}_q}$ for ordinary abelian varieties. In particular $\mathrm{End}(A) \simeq \mathrm{End}(\widetilde{A})$, and this characterizes $\widetilde{A}$ [Mes72, Appendix, Corollary 1.3].

Canonical lifts are compatibles with the Galois action. Hence if $A/\mathbb{F}_q$ is an ordinary abelian variety, the (small) Frobenius $\pi_p : A \to \pi_p(A)$ lifts to an isogeny $\Sigma_p : \widetilde{A} \to \sigma(\widetilde{A})$. The existence of this lifting is also enough to characterize $\widetilde{A}$ [Mes72, Appendix, Corollary 1.2] and provides a way to compute it (in good cases), see Section 6.3.

Furthermore, by construction of the canonical lift, $\widetilde{A}(p) \simeq \widetilde{A}(p)^\circ \oplus \widetilde{A}(p)^{\text{ét}}$, and $T_p(\widetilde{A}(p)^{\text{ét}})$ is isomorphic as a $\mathbb{Z}_p[\pi]$-module to $T_p(A(p)^{\text{ét}})$, while $T_p(\widetilde{A}(p)^\circ)$ is its dual. Over the finite extension $\mathbb{F}_{q^e}$ where the étale p^m-torsion points become defined, $A[p] \simeq \mu_{p^m}^g \times (\mathbb{Z}/p^m\mathbb{Z})^g$, and so the same holds for the lift $\widetilde{A}$ over $\mathbb{Q}_{q^e}$, in particular $\widetilde{A}[p^m](\overline{\mathbb{Q}_p}) \subset \mathbb{Q}_{q^e}[\zeta_{p^m}]$. So the étale part of $A[p^m]$ lifts into the unramified extension $\mathbb{Q}_{q^e}$, while the local part lift into a (tamely if $m = 1$) ramified extension.

The idea to use canonical lifts algorithmically for point counting is due to Satoh [Sato0]. We have seen in Chapter 5 that we can also use lifts to compute modular polynomials, and in Chapter 7 we will use lifts to compute (Shimura) class polynomials.

6.2.2 *Using lifts for point counting*

Let $q = p^d$ and $A/\mathbb{F}_q$ be an ordinary abelian variety. We denote by π_q the Frobenius over $\mathbb{F}_q$ and by π or π_p the small Frobenius. We will let $\hat{\pi}_q$ be the big Verschiebung and $\hat{\pi}_p$ the small Verschiebung. We will use the abusive but convenient notation $\pi(A)$ to denote the image of the relative Frobenius: $\pi(A) = A \times_\pi \mathbb{F}_q$.

Then since the Verschiebung $\hat{\pi}_q$ is separable, the Frobenius χ_π has g inversible eigenvalues λ_i modulo p whose product modulo p is the determinant of $d\hat{\pi}_q$ on the tangent space of $\hat{\pi}_q$ at 0. The remaining eigenvalues are given by q/λ_i, so the inversible eigenvalues are enough to recover χ_π. In the following we will often switch from tangent spaces to differentials (ie the cotangent space). By duality taking a basis of one induce a basis of the other.

Since $A[p^d] \simeq A_{\text{ét}}[p^d] \oplus A_{\text{mult}}[p^d]$, we may compute the étale part of $A[p^d]$ to get the kernel of $\hat{\pi}_q$ and apply an isogeny algorithm to compute its action on a differential basis w_A. For instance if A is an elliptic curve E, the p^d-division polynomial $\Psi_{E,p^d}(x)$ is of the form $\left(\psi_{E,p^d}\right)^{p^d}$ with ψ_{E,p^d} of degree $\frac{p^d-1}{2}$ encoding the x-coordinates of the points of $E_{\text{ét}}[p^d]$. (Here we assume $p > 2$ for simplicity in the description of the division polynomials. The division polynomial is not of degree $(p^{2d}-1)/2$ as expected because its leading coefficients are zero modulo p.).

If E is given by a Weierstrass equation $y^2 = x^3 + ax + b$, we have the canonical differential $w_E = dx/y$. Applying Vélu's formula to P, we compute another elliptic curve $E' : y^2 = x^3 + a'x + b'$, such that the isogeny $f : E \to E'$ is normalised, ie $f^* w_{E'} = w_E$. But E' is isomorphic to E and computing the isomorphism gives us exactly the invertible eigenvalues (modulo p). Indeed, isomorphisms of short Weierstrass equations are given by $(x, y) \mapsto (u^2x, u^3y)$, if $E_u : y^2 = x^3 + au^4x + bu^6$ is the image of E by this isomorphism, $w_{E_u} = \frac{1}{u} w_E$. So if $a'/a = u^4$ and $b'/b = u^6$, we get that the action of π_q on w_E is given by the multiplication by u. (This method only recovers u^2; we need the equation of the isogeny rather than just the coefficients of the normalized curve to get u.)

Of course computing an isogeny of degree q is not possible in the cryptographic setting of a large q, but if $q = p^d$ with a small p and a large d, we may instead decompose the Verschiebung $\hat{\pi}_q$ as d small Verschiebungs $\hat{\pi}_p$. In fact we only need to compute one small Verschiebung: given the action M on differentials of $\hat{\pi}_p : (\pi(E), \pi(w_E)) \to$

[3] It is also fully determined by $T_p(A) = T_p(A(p)_{\text{ét}})$ which is both the Galoisian free $\mathbb{Z}_p$-module associated to $A(p)_{\text{ét}}/k$ by Grothendieck's étale Galois theory and also the dual of $H^1(A, \mathbb{Z}_p) \simeq \mathrm{Hom}(\pi^1_{\text{ét}} A, \mathbb{Z}_p)$, encoding the étale covers of A of degrees a power of p.

(E, w_E) (ie $\hat{\pi}_p^* w_E = M\pi(w_E)$), then the action of $\hat{\pi}_p : (\pi^2(E), \pi^2(w_E)) \to (\pi(E), \pi(w_E))$ is simply $\pi(M)$ and so on. So the full action of $\hat{\pi}_q : (E, w_E) \to (E, w_E)$ is given by the $\mathbb{F}_q/\mathbb{F}_p$ norm of M. Of course the same strategy holds for an abelian variety, except this time M is a $g \times g$ matrix rather than a scalar. Unfortunately if p is small, knowing the eigenvalues modulo p does not give enough informations.

Satoh's insight in [Satoo] was to first lift $E/\mathbb{F}_q$ to its canonical lift $\widetilde{E}/\mathbb{Q}_q$ where $\mathbb{Q}_q$. Then $\sigma(\widetilde{E})$ is a canonical lift of $\pi(E)$. Furthermore, since $\sigma_q := \sigma^q$ reduces to π_q modulo p, its action gives the lift of the endomorphism of E induced by the Frobenius $\pi_q \in \mathrm{End}(E)$, ie it corresponds to $\pi_q \in \mathrm{End}(\widetilde{E}) = \mathrm{End}(E)$. Alternatively, by étaleness of $T_\ell(E)$ for $\ell \neq p$, $T_\ell(E)$ is isomorphic to $T_\ell(\widetilde{E})$ as a $\mathbb{Z}_\ell[\pi]$-module, so the eigenvalues are the same.

The action σ can be computed efficiently if we represent $\mathbb{Z}_q$ by a Teichmuller lift, ie by $\mathbb{Z}_p[X]/T(X)$ where T is a factor of $X^{q-1} - 1$, ie $x = X \mod T$ is a $q-1$ root of unity. In this case $\sigma(x) = x^p$, so $\sigma(\sum a_i x^i) = \sum a_i (x^p)^i$ and the computation of σ can be done in time $O(md)$ if p is fixed and m is the p-adic precision. The Teichmuller lift T can be efficiently computed via Newton iterations from a defining polynomial for $\mathbb{F}_q$.

Since canonical lifting is functorial, the Verschiebung $\hat{\pi}_p$ and Frobenius π_p lift to $\widetilde{E}$, we will call the lifts $\hat{\Sigma}_p$ and Σ_p. Since we are in characteristic zero, computing the action of $\hat{\Sigma}_q : \widetilde{E} \to \widetilde{E}$ on tangent space is enough to recover χ_π, and as before it suffices to compute the action M of $\hat{\Sigma}_p : \sigma(E) \to E$ and take its norm over $\mathbb{Q}_q/\mathbb{Q}_p$.

Note that we could also compute the action of Σ_q instead (via Σ_p) but we would get the non invertible eigenvalues modulo p, so this would cause a loss of precision.

This yields the following algorithm for an abelian variety $A/\mathbb{F}_q$.

Algorithm 6.2.1. (i) Compute the canonical lift $\widetilde{A}$ at p-adic precision m;
(ii) Lift the kernel of the Verschiebung $\hat{\pi}_p$ to $\widetilde{A}$ to get the kernel of $\hat{\Sigma}_p$;
(iii) Compute the action M of $\hat{\Sigma}_p : \widetilde{A} \to \sigma^{-1}(\widetilde{A})$ on tangent spaces or differentials given by $w_{\widetilde{A}}, \sigma^{-1} w_{\widetilde{A}}$, via an isogeny algorithm.
(iv) Compute the characteristic polynomial P of the norm of M in $\mathbb{Q}_p$.
(v) Recover $\chi_\pi(X) = P(X) q^g P(q/X)$ as an element of $\mathbb{Z}[X]$.
From Weil's bound, the last step shows that we need to take the precision m to be $gd/2 + O(1)$.

Considering p and g fixed, we will see that there exists a quasi-linear algorithm for each of these steps, so since we work at precision $O(d)$ over $\mathbb{Z}_q$ we have a complexity of $\widetilde{O}(d^2)$ binary operations, ie a quasi-quadratic point counting algorithm *with respect to* d. The complexity in terms of p and g depends on the algorithms used, but will typically be polynomial in p^g.

Let us detail the different steps, in the case of an elliptic curve. For more details, we recommend the excellent survey [Gauo4].

6.2.3 *Computing a canonical lift of an elliptic curve*

For reasons we will explain below, we assume from now on that $j_E \notin \mathbb{F}_{p^2}$ (and we will do similar assumptions when lifting abelian varieties).

For **??** 6.2.1.(i), Satoh's original algorithm was to lift all curves E_i together, where $E_i = \pi^i(E)$ via modular equations $\Phi_p(j_{\widetilde{E}_i}, j_{\widetilde{E}_{i+1}}) = 0$, letting $j_{\widetilde{E}_n} = j_{\widetilde{E}_0}$. Using multivariate Newton iterations, this gives a $O(d^3)$ lifting algorithm.

The paper [VPVo1] remarked that rather than lifting the whole isogeny cycle directly, we could instead proceed isogeny by isogeny along the cycle, augmenting the precision by one each time. This does not change the $O(d^3)$ time complexity but the memory is $O(d^2)$. In fact this is just a variant of the fixed point algorithm when we have a contracting function: $j_{\widetilde{E}}$ is a fixed point of $\Phi_{p^d}(j_{\widetilde{E}}, j_{\widetilde{E}}) = 0$, but when we use Φ_p instead we have a fixed cycle of length d. So we start with the fixed cycle modulo p, $\Phi_p(j_{E_i}, j_{E_{i+1}})$ and iterate the function Φ_p. Using the properties of $\Phi_p \mod p$ given by Kroenecker's equality (see below), it is not hard to show that we obtain a linear convergence to the cycle $j_{\widetilde{E}_i}$.

A quasi-linear lifting algorithm (ie in $O(d^2)$) was proposed by Harley in [Haro2], using the equation $\Phi_p(j_{\widetilde{E}}, \sigma(j_{\widetilde{E}}))$ instead. For a Newton iteration, given a solution j at precision k, we write $j_{\widetilde{E}} = j + p^k e$ and solve $\Phi_p(j + p^k e, \sigma(j) + p^k \sigma(e) = 0 \mod p^{2k})$. This gives an equation of the form

$$v + e\partial\Phi_p/\partial X(j, j^\sigma) + e^\sigma \partial\Phi_p/\partial Y(j, j^\sigma) = 0 \mod p^k.$$

Kronecker's equality states that $\Phi_p(X,Y) = (X^p - Y)(Y^p - X) \mod p$. So we get that $\partial\Phi_p/\partial X(j,j^\sigma) = 0 \mod p$ while $\Phi_p/\partial Y(j,j^\sigma) \neq 0 \mod p$. The last inequality is only valid if $j \notin \mathbb{F}_{p^2}$. Indeed if E is defined over $\mathbb{F}_{p^2}$, $\pi = \hat{\pi}$ so we have multiplicity problems when lifting.

So the equation is of the form $e^\sigma + Ae + B$ with $A = 0 \mod p$. This is often called an Artin-Schreier equation in the literature (wrongly IMHO since its reduction modulo p is $e^p + B = 0$ which is not separable hence not an Artin-Schreier polynomial). Solving this equation is done via a Newton iteration (so we do a Newton iteration inside a Newton iteration, which should please fans of the movie Inception), and we get the solution in quasi-linear time. Indeed, writing $e = e_1 + p^k e_2$ with e_1 a solution at precision k we have to solve an equation $e_2^\sigma + Ae_2 + B' = 0 \mod p^k$. Since $A = 0 \mod p$, the initialisation modulo p is simply taking a p-th root.

We note that while for the isogeny step it is more convenient to lift the Verschiebung to compute $\hat{\Sigma}_p$, for the canonical lift step it is better to look at the Frobenius, ie solve $\Phi_p(j,j^\sigma) = 0$ rather than for $\Phi_p(j,j^{\sigma^{-1}})$. Indeed, lets take a basis (P,Q) of $\widetilde{E}[p](\overline{\mathbb{Q}_q})$, where P reduces to an étale point modulo p and Q reduces to 0_E. In particular P lives in an étale extension of $\mathbb{Q}_q$ while Q in a (tamely) ramified extension. Then $\langle Q\rangle$ is the unique lift of the Frobenius, while there are ℓ lifts of the Verschiebung given by $\langle P + \lambda Q\rangle$. So from the point of view of Newton lifts, it is better to lift using the Frobenius.

6.2.4 *Lifting the kernel of the Verschiebung*

We have seen that there are several ways to lift the Verschiebung, but since P is in an unramified extension of $\mathbb{Q}_q$ and Q is in a ramified extension, there is a unique unramified lift given by $\langle P\rangle$. We will call this the canonical lift of the Verschiebung. We discuss several ways of computing this lift. The first idea is to compute the point P by lifting its reduction $P \mod p$.

So we have a system of equations in (x_P, y_P) given by the equation of $\widetilde{E} : y_P^2 = f_{\widetilde{E}}(x_P)$ and $[p](x_P, y_P) = 0$. Since the multiplication by $[n]$ acts by n on the tangent space at 0 of an elliptic curve, and since $\widetilde{E}/\mathbb{Z}_q$ is an elliptic curve so in particular is smooth, we see that the Jacobian of this system has a Smith normal form with diagonal $(1,p)$, hence is not invertible.

Of course for elliptic curves it suffices to solve for x_P, ie solve $\Psi_{\widetilde{E},p}(x_P) = 0$ where $\Psi_{\widetilde{E},p}$ is the p-division polynomial ($p > 2$). By the same reasoning as above, Ψ_p' is of valuation 1, as can also be seen from the fact that $\Psi_{\widetilde{E},p}(X) \mod p = \Psi_{E,p}(X) = \psi_{E,p}(X)^p$, $\psi_{E,p}(X)$ being the polynomial of degree $(p-1)/2$ giving the roots of the étale points of $E[p]$, hence is separable.

Aside 6.2.2. Lets look in more details on how to do Newton iterations to solve an equation $f(x) = 0$ where we have a solution $X_n \mod p^n$ such that $f'(X_n)$ is of valuation e. Writing $Z = X_n + p^n Y_n$, we get (if $p \neq 2$) $f(Z) = f(X_n) + p^n f'(X_n)Y_n + p^{2n}\frac{f''(X_n)}{2}Y_n^2 + O(p^{3n})$.

1. If $e < n$, $f'(X_n)$ is well defined modulo p^n, $f(X_n)$ is well defined modulo p^{n+e} (ie does not depend on the choice of representative of $X_n \mod p^n$) and we need $f(X_n) = 0 \mod p^{n+e}$. Writing $n = e + m$, we look for $Y_n \mod p^m$, and we solve for $f(Z) = 0 \mod p^{2n}$, so we have to solve an equation $f(X_n)/p^{n+e} + f'(X_n)/p^e Y_n = 0 \mod p^m$. This gives us $Y_n \mod p^m$, hence $Z \mod p^{e+2m}$ such that $f(Z) = 0 \mod p^{2e+2m}$. Furthermore $f'(Z) = f'(X_n) + f''(X_n)p^{e+m}Y_n + O(p^{e+2m})$ is still of valuation e. Letting $X_{e+2m} = Z$, we find $X_{e+2m} \mod p^{e+2m}$ such that $f'(X_{e+2m})$ is of valuation e and $f(X_{e+2m}) = 0 \mod p^{2e+2m}$, doubling the m precision. To boostrap we need $m = 1$, ie $X_{e+1} \mod p^{e+1}$ such that $f(X_{e+1}) = 0 \mod p^{2e+1}$. We recover the standard Newton lifting when $e = 0$.

   The exact same reasoning holds in the multivariate case, if we let e be the smallest integer such that $p^e f'(X_n)^{-1}$ has integral coefficients.

2. The more interesting case is $n \leq e$. A complication is that the valuation of $f'(X_n)$ may depend on the choice of representative of $X_n \mod p^n$, so here we fix one representative such that $f'(X_n)$ is of valuation e. However, all representatives X_n satisfy $f'(X_n) = 0 \mod p^n$, so $f(X_n)$ is well defined modulo p^{2n}, and if furthermore $f''(X_n) \neq 0 \mod p$ we may take a representative such that $e = n$.

   In any case, we need to assume $f(X_n) = 0 \mod p^{2n}$. We write $X_{n+1} = X_n + p^n x_n$ and we try to determine the value of $x_n \mod p$. We need to solve a quadratic equation $ax_n^2 + bp^{e-n}x_n + c = 0 \mod p$ where $a = f''(X_n)/2$, $b = f'(X_n)/p^e$, $c = f(X_n)/p^{2n}$. If a solution x_n exists, then $f(X_{n+1}) = 0 \mod p^{2n+1}$, $f'(X_{n+1}) = f'(X_n) + pf''(X_n)x_n = bp^e + 2ap^n x_n$.

a) If $f(X_n)$ is of valuation exactly $2n$, then $c \neq 0 \mod p$, so if a solution exists, $x_n \neq 0 \mod p$. If $a = 0 \mod p$, a solution is possible only if $e = n$, and then $f'(X_{n+1})$ is still of valuation e. Since $f(X_{n+1}) = 0 \mod p^{2n+1}$, we can now bootstrap using Item 1. Note that in this case we may compute x_{n+1} as $-f(X_n)/f'(X_n)$, ie exactly as a standard Newton iteration (but which only increase the precision by 1 rather than doubling it) which is well defined modulo p because both terms are exactly divisible by p^e.

   If $a \neq 0 \mod p$, there are two potential solutions. If $e > n$, the two solutions are distincts. If x_n is one of them, then $f'(X_{n+1})$ is of valuation $n, f(X_{n+1}) = 0 \mod p^{2n+1}$ so we can bootstrap using Item 1. If $e = n$, there are still two, not necessarily distinct, potential solutions. If they are distinct, $f'(X_{n+1})$ is still of valuation e, and $f(X_{n+1}) = 0 \mod p^{2n+1}$ so we can bootstrap using Item 1. If they are not distinct, then $b + 2ax_n = 0 \mod p$, hence the valuation of $f'(X_{n+1}) > e$.

b) Otherwise $c = 0 \mod p$, so $x_n = 0$ is a solution. There is a second solution only if $e = n$ and $a \neq 0 \mod p$. In this case $f'(X_{n+1})$ is still of valuation e, so we may bootstrap using Item 1.

Going back to elliptic curves, we may check that our solution $\Psi_{\widetilde{E},p}(x_P) = 0$ satisfy $\Psi'_{\widetilde{E},p}(x_P \mod p)$ is of valuation exactly $e = 1$, $\Psi''_{\widetilde{E},p}(x_P \mod p) = 0 \mod p$ and $\Psi_{\widetilde{E},p}(x_P \mod p) = 0 \mod p^2$ (these value does not depend on the choice of representatives). (Here we assume $p > 3$, see [FGHoo] for the cases $p = 2, 3$.) So at the initialisation, we are exactly in the case of Item 2a with $n = e = 1$, and $a = 0 \mod p$, so we may compute a solution without worries using Newton iterations (even though the required precision at the start is only 2 rather than 3 as we would have naively expected to apply Newton iterations). This explains eg why [FGHoo, § 7.2] converge, even through we are not in the conditions of their Lemma 2.1.

Remark 6.2.3. When doing a Newton iteration, when we double the precision m to $2m$, then $(p^m)^2 = 0$ in $\mathbb{Z}_q/p^{2m}\mathbb{Z}_q$. Thus we don't need the polynomial equation $f(x) = 0$, we just need a way to evaluate f, then evaluating at x and $x + p^m$ gives $f'(x)$. A similar method holds in the multivariate case. We can apply this when lifting a p-torsion point: the evaluation $[p]P$ is given by the double and add algorithm.

Of course this method has the inconvenience of requiring to work in an extension of $\mathbb{F}_q$ (hence an unramified extension of $\mathbb{Q}_q$) where the étale points of p-torsion are defined. It is better to lift the polynomial $\psi_p(X) \mod p$ as a factor of $\Psi_p(X)$ directly. It is standard to use Newton lifting to lift a polynomial decomposition $P(X) = P_1(X)P_2(X)$ provided $P_1 \mod p$ and $P_2 \mod p$ are prime to each other, but in our case $\psi_p(X)$ is a factor of multiplicity p. Satoh carefully handles this case in [Satoo, Lemma 2.1].

As Xavier Caruso explained to me, a high level overview of Satoh's algorithm may be seen as follow: we know from the general theory of canonical lift that there is a unique unramified lift of $\psi_p(X)$. In particular this lift is a factor of the horizontal slope part of the slope decomposition of $\Psi_p(X)$. We recall that the slopes of the Newton polygon of a polynomial P each correspond to a factor P_i (not necessarily irreducible) of P, whose roots have valuations encoded by the slope: P_i is of degree the x-length of the line segment, and the roots have valuation $-\mu_i$ where μ_i is the slope of the line segment. In particular our unramified factor corresponds to the horizontal segment of the Newton polygon of Ψ_p.

Slope factorization may be computed by Newton iterations [CRV16, § 4]. Hence we may see Satoh's algorithm [Satoo, Lemma 2.1] as a fancy way to combine slope factorization and Newton lifting of factorizations into relatively prime components in one step.

6.2.5 *Computing the isogeny*

Once we have lifted the kernel of the Verschiebung, either by lifting its defining polynomial directly or by lifting a generator, we may apply Vélu's formula. This gives us the equation of a curve $\widetilde{E}_1 : y^2 = x^3 + a_1x + b_1$. But if $\widetilde{E} : y^2 = x^3 + ax + b$, we know that $\widetilde{E}_1$ is isomorphic to $\sigma^{-1}(\widetilde{E}) : y^2 = x^3 + \sigma^{-1}(a)x + \sigma^{-1}(b)$. As we have seen, the equation of an elliptic curve E implicitly encodes the choice of a differential $w_E = dx/y$ (up to a sign), and $\sigma^{-1}(\widetilde{E})$ corresponds to the choice $\sigma^{-1}(w_{\widetilde{E}})$, while the curve computed by Vélu's formula correspond to the choice $f^*w_{\widetilde{E}_1} = w_{\widetilde{E}}$ where f is the isogeny. So the isomorphism encodes the action on differentials. Using the notations of Section 6.2.2, if $\sigma^{-1}(\widetilde{E}) = \widetilde{E}_{1,u}$ where $\widetilde{E}_{1,u}$ is the image of $\widetilde{E}_1$ by the isomorphism $(x, y) \mapsto (xu^2, yu^3)$ which acts by $M := 1/u$ on the canonical differentials dx/y, then $\hat{\Sigma}_1^*\sigma^{-1}w_{\widetilde{E}} = Mw_{\widetilde{E}}$, and M is then the inversible eigenvalue. Of course we could also compute the lift of the Verschiebung on $\sigma(\widetilde{E}) \to \widetilde{E}$ instead.

We have $a_1/\sigma^{-1}(a) = M^4, b_1/\sigma^{-1}(b) = M^6$. The only complication is that the isomorphism between $\widetilde{E}_1$ and $\sigma^{-1}(\widetilde{E})$ only gives the value of M^2, to get M we also need the equation of the isogeny $\widetilde{E} \to \widetilde{E}_1$ so we can compute the pullback of $\sigma^{-1}(w_{\widetilde{E}})$ through the map $\widetilde{E} \to \widetilde{E}_1 \simeq \sigma^{-1}(\widetilde{E})$. In practice, taking the norm v of M^2 allows easily

to recover the trace t^2 as $t^2 = v + q^2/v + 2v$, so we get $\pm t$, and we can recover t by standard methods (by testing on a point, or computing the trace modulo p using Hasse's formula, see [Satoo, Theorem 4.4]).

Remark 6.2.4. A similar remark holds for Jacobians of hyperelliptic curves of genus $g = 2$: if $J = \mathrm{Jac}(C)$, a choice of curve equation of C encodes a basis of differential $w_J = (x^i dx/y)$. If we compute a canonical lift $\widetilde{J}$ of J via a lift $\widetilde{C}$ of C, and we have an isogeny formula which allow to compute the normalised isogenous curve $\widetilde{C}_1$ for the isogeny f corresponding to the (canonical) lift of the Verschibung, then computing an isomorphism between $\sigma^{-1}\widetilde{C}$ and $\widetilde{C}_1$ allows to recover the tangent matrix (see eg [Meso1, § 2]). For $g > 2$, the action of automorphisms on hyperelliptic curves is still given by Gl_2, so not every choice of basis of differential correspond to a choice of curve equation. In particular we may not always encode the normalised isogeny by only giving the curve equations.

6.2.6 *Taking the norm*

For elliptic curves, the action on differentials is given by a scalar $M \in \mathbb{Q}_q$. If $\mathbb{Q}_q$ is represented by a Teichmuller lift $T(X)$, the norm of M is simply given by the resultant of M (seen as a polynomial in $\mathbb{Q}_p[X]$ modulo T) with T, so can be computed asymptotically in quasi-linear time in the precision m.

In practice most implementations use the formula $N_{\mathbb{Q}_q/\mathbb{Q}_p}(x) = \exp(\mathrm{Tr}_{\mathbb{Q}_q/\mathbb{Q}_p}(\log x))$ which can be computed in time $O(m^{3/2})$ [SSTo3].

Let us now look at how to generalise these steps for abelian varieties.

6.3 CANONICAL LIFTS FOR ABELIAN VARIETIES

The obvious generalisation of Section 6.2.3 is to use a modular correspondance Φ_p defined on some modular invariant J on $\mathcal{A}_g$ to lift an ordinary abelian variety $A/\mathbb{F}_q$.

Letting $A_i = \pi^i(A), \widetilde{A}_i$ the canonical lift of A_i, which is also $\sigma^i(\widetilde{A})$, we may either use a fixed point like algorithm to lift the isogeny cycle $\Phi_p(J_{\widetilde{A}_i}, J_{\widetilde{A}_{i+1}}) = 0$ by iterating the modular correspondance, or solve directly the equation $\Phi_p(J_{\widetilde{A}}, \sigma(J_{\widetilde{A}})) = 0$.

Like in Section 6.2.3, a Newton approach write $J = J + p^k e$ and then solves an equation of the form

$$v + e\partial\Phi_p/\partial X(J, J^\sigma) + e^\sigma \partial\Phi_p/\partial Y(J, J^\sigma) = 0 \mod p^k.$$

Now if $J \notin \mathbb{F}_{p^2}$, we expect $\partial\Phi_p/\partial Y(J, J^\sigma)$ to be inversible modulo p (because the Frobenius lifts uniquely), and we expect $\partial\Phi_p/\partial X(J, J^\sigma) = 0$ (essentially because otherwise we could have multiple possible lifts). In dimension 1 this was a corollary of the Kronecker's equality, so we call this assumption the Kronecker condition. In dimension 2 we prove the Kronecker condition (generically) in [MR20a] in a somewhat ad-hoc way, by directly differentiating the equation defining the polynomials and looking at the multiplicity of the roots. This should extend in any dimension, but a cleaner proof follows formally from standard arguments using the canonical coordinates of the Serre-Tate local moduli [Kat81]. In particular [CN90, § 3.4] gives a description of the modular correspondance $\widetilde{\Phi}$ in the local moduli, from which we get immediately that in terms of the local coordinates x, x^σ corresponding to J, J^σ, $\partial\widetilde{\Phi}/\partial X(x, x^\sigma) = 0 \mod p$ while $\partial\widetilde{\Phi}/\partial Y(x, x^\sigma)$ is inversible modulo p. Note that this only means that the Kronecker conditions are satisfied generically for the modular polynomial Φ_p, because it only encodes a *birational* equation of the *coarse* moduli space. That's why, although the *fine* modular space of level $\Gamma^0(p)$ is smooth at ordinary points modulo p, for elliptic curves both derivative of the modular polynomial vanish whenever $j_E \in \mathbb{F}_{p^2}$, ie the modular polynomial is not smooth at these points. In particular, Φ_p does not define $X_0(p)$ around the points (j_E, j_E^σ) when $j_E \in \mathbb{F}_{p^2}$, so to lift around these points one need to compute the normalisation so that we get the correct equations (at least when there are no extra automorphisms so that the coarse space is equal to the fine space).

Note that if the Kronecker condition were not satisfied, we could still solve the Artin-Schreier like equation using a Newton approach, the only things that change is the initialisation step: modulo p we have to solve an equation of the type $Ae^p + Be + C = 0$ (e a vector of length $g(g+1)/2$). If $B \neq 0$, we could try to find a solution by viewing the equation as an $\mathbb{F}_p$-linear equation of dimension $dg(g+1)/2$, but then we are not quasi-optimal anymore. This situation happens for instance when there is an ℓ-isogeny relating A and a Galois conjugate, so we can use ϕ_ℓ instead of ϕ_p to lift A, see [Koho8].

Note also that if the Kronecker condition is satisfied, we can also lift the abelian variety by using a fixed point approach like in Section 6.2.3, iterating the modular correspondance to lift the isogeny cycle. The increase of

precision is only linear rather than quadratic as in Newton's approach, hence this does not give a quasi-linear lifting algorithm either.

Example 6.3.1. (i) Mestre introduced in [Meso1] the idea to use the AGM to compute the canonical lift of an elliptic curve in characteristic 2. Since the AGM encodes the duplication formula for theta functions, it can be seen as a modular correspondance of level $p = 2$. Hence Mestre's algorithm to iterate the AGM may be seen as the fixed point variant of the canonical lift algorithm (so the lifting algorithm is not quasi-linear). Of course this can be generalised to abelian varieties in characteristic two by using the duplication formula for theta functions of level 2, as Mestre does in [Meso2]. See also [Rito3], and [Caro3, § 2] for an algebraic proof of the convergence of the fixed point approach. We also refer to [LLo3; LLo6] for the Newton approach to lifts via the duplication formulae, and to [Koho8, § 4.1] for a simplification of the modular equations induced by the duplication formula in the case $g = 2$ using a different parametrization.
This requires to work in a field extension where the level 2 symmetric theta structure is defined. In dimension $g = 2$, since a principally polarised abelian surface is generically a Jacobian of an hyperelliptic curve, one can also use the level 2 modular correspondance induced by Richelot's isogeny [Ric36; Ric37], using curve equations directly as in [Meso1, § 2] (via a fixed point approach) or using the Rosenhain invariants λ, μ, ν as in [GHK+06]. (which only requires to take a field extension given by a $A[2]_{\text{ét}}$ and allows a Newton approach).

(ii) Using theta functions there is a level p^2 "multiplication formula", see Remark 2.8.1. Indeed, with the notation of this Remark, taking $n = m = 1$ and $s = t = p$, the isogeny $F : A \times A \to A \times A$, $(P, Q) \mapsto (P + pQ, P - pQ)$ satisfy $F^*(\mathcal{L} \star \mathcal{L}) = \mathcal{L}^2 \star \mathcal{L}^{2p^2}$.
The p^2-multiplication induces a modular correspondance Φ_{p^2} of degree p^2 (ie encoding p^2-isogenies) on $\mathcal{A}_{g,np}$. Here $\mathcal{A}_{g,n}$ refers to the moduli of symmetric theta structure of level n ie to $\mathcal{A}_{g,n,2n}$, see Remark 2.7.5. Typically we use $n = 2$ or $n = 4$. This is used in [CLo8a] for canonical lifting and point counting. For lifting the equation to solve is then $\Phi_{p^2}(\theta_i(\widetilde{A}), \sigma^2(\theta_i(\widetilde{A}))$. We can reformulate [CLo8a, Theorem 2.1] as follow: the p^2-multiplication map relates theta null points on $\mathcal{A}_{g,np}$ with theta null points of $\mathcal{A}_{g,np^3}$. Projecting back to $\mathcal{A}_{g,np}$ via Mumford's isogeny formula, this defines a modular correspondance $\mathcal{A}_{g,np^3} \to \mathcal{A}_{g,np} \times \mathcal{A}_{g,np}$. Compared to the general modular correspondance of Section 5.2, the fact that we stay of level p on the target allows to define equations for the image directly on $\mathcal{A}_{g,np}$.
A 3-multiplication formula was used in [CKLo8] to compute canonical lifts of abelian surfaces in characteristic $p = 3$ and the general p^2-multiplication formula is used in [CLo8a]. We remark that Kempf's multiplication Theorem 2.12.2 formula (or simply Lemma 2.6.9) gives a direct (and much simpler) proof of [CKLo8, Proposition 3.7, Lemma 3.8].
We can also use Section 2.6 to revisit (and give simpler proofs of) [Caro3, Theorem 4.1.1 and Corollary 4.1.2; Caro7, Theorem 2.1] for the cases we are interested in. Indeed, if we suppose that $\mathcal{L}$ is symmetric, then using the notation of Carls, $\mathcal{L}^{(q)}$ is symmetric, so we may construct it by taking a symmetric lift of the kernel. But if $p > 2$ this symmetric lift is unique.
Furthermore, if $A/\mathbb{Z}_q$ is an abelian scheme with ordinary reduction and which is the canonical lift of its reduction, there is a canonical symplectic decomposition $A[p] = A^{\text{ét}}[p] \oplus A^0[p]$ induced by lifting the corresponding decomposition modulo p (see Section 6.2.1). (In fact by [Caro3, § 2.1] there is also such a canonical decomposition on $A^{(p)}$ without assuming A canonical, this is a consequence of the fact that iterating the AGM converges to the canonical lift $\widetilde{A}$, so we have symplectic decompositions on $A^{(p^d)}[p^d]$ which converge to the symplectic decomposition of $\widetilde{A}(p)$.)
If $\mathcal{L}$ is symmetric and principal, this gives a canonical symmetric theta structure on $\mathcal{L}^p$ ($p > 2$). Similarly, if $\mathcal{L}$ is a totally symmetric line bundle of level m with m prime to p and we have a symmetric theta structure on $\mathcal{L}$ induced by a symplectic decomposition of $A[2m]$ (see Section 2.6.3), we get a canonical symplectic decomposition of $A[2mp]$, hence a canonical symmetric theta structure on $\mathcal{L}^p$, compatible with the Galois action. In particular since σ and Σ reduces to the Frobenius modulo p, we have that $\Sigma : (A, \mathcal{L}^p) \to \sigma(A, \mathcal{L}))$ is compatible with the symmetric theta structures (compare with the proof in [Caro3, § 4.5.3]). Using Section 2.6.3 we can also treat the case $p = 2$.

(iii) In [FLR11], we use a modified version of the modular correspondance of Section 5.2, namely we use $\pi_3 \times \pi_2 : \mathcal{A}_{g,np} \to \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$ which maps A to $(A/A_1[p], A/A_2[p])$ where $A[p] = A_1[p] \oplus A_2[p]$ is the symplectic decomposition induced by the theta structure of level np (because we did not have the version of Section 5.2 at the time).
This is also of degree p^2, and can be seen as a simplification of the equations of [CLo8a]. Indeed, although only implicitly stated in the article, we can use this modular correspondance to compute the canonical lift, by

solving for $\theta_i(\widetilde{A}) \in \mathcal{A}_{g,np}$ such that $\pi_2(\theta_i(\widetilde{A})) = \sigma^2(\pi_3(\theta_i(\widetilde{A}))$. This give simpler equations than using the p^2-multiplication formula in $\mathcal{A}_{g,np}$.
We could instead use the modular correspondance $\pi_1 \times \pi_2$ defined in Section 5.2.1, given by $A \mapsto (A, A/A_2[p])$ which is of degree p (ie encoding p-isogenies). This does not change the dependency on d but improve slightly the dependency on p: using a p^2-modular correspondance does not really change the Newton step but it requires to compute σ^2 rather than σ, so is a bit slower.
A drawback of the modular correspondance $\mathcal{A}_{g,np} \to \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$ is that the real modular correspondance is only defined by the image of $\mathcal{A}_{g,np}$, but we don't have equations for this image (see the discussion at the end of Section 5.2.3). So in practice starting with $A/\mathbb{F}_q$ with a theta structure of level n on $\mathcal{L}_0^n$, we compute a level np theta structure (ie a point in $\mathcal{A}_{g,np}$ given by $\theta^{\mathcal{L}_0^{np}}(0)$) and lift this theta null point of level np. So while the equations are simpler, similarly to the modular correspondance of **??** 6.3.1.(ii) on $\mathcal{A}_{g,np}$ induced by the p^2-multiplication formula, we lift a theta null point of level np. So in the initialisation step, we need to go from the theta null point of level n over $\mathbb{F}_q$ to the theta null point of level np.
In [CL08a] lifting A of level n to level np was done by a Groebner basis algorithm. Our main motivation in [FLR11] to introduce the modular correspondance above was that this helped the lifting of A from level n to level np. This still used a Groebner basis algorithm, but one optimised for the symmetries coming from the automorphisms Section 5.2.3 of the system, see [FLR11, § 6].
With the results of Section 2.10 we can now easily do this lift of level computation provided we have the points of $A[p]_{\text{ét}}$. We can use both the lift via an isogeny version of Section 2.10.1 or by staying on the same variety as in Section 2.10.2. Indeed since A is ordinary it lifts, so a theta structure of level np on A makes sense even if the polarisation is inseparable. We remark that since $A[p] = A[p]_{\text{mult}} \oplus A[p]_{\text{ét}}$ is a symplectic decomposition (here we work with group schemes), the points of $A[p]_{\text{ét}}$ implicitly determine the formal points of $A[p]_{\text{mult}}$.
This reduces the problem of lifting to level np to finding the étale points of p-torsion on A, which has been well studied, if only for the Schoof-Pila point counting algorithm, see Section 5.5. We can then bound the dependency on p of the algorithm, for instance [AH01] gives an algorithm polynomial in p using multivariate resultants to describe $A[p]$. We could also use the geometric resolution algorithm of [AGS19b, Proposition 3].

(iv) In [MR20a], we compute lifts of abelian surfaces using the Siegel modular polynomials defined in Section 5.3. Using modular polynomials in Igusa invariants has the advantage that it can always be done over the base field, while with the theta modular correspondance above we have seen that we need to take an extension where the level np theta structure is defined.
We also computed canonical lifts using theta modular polynomials (ie using a symmetric theta structure of level 2). We then lift the kernel of the Verschiebung (see Section 6.5), and compare the action of σ^{-1} and of the normalised isogeny. In this model we can use the modular isogeny algorithm from Section 4.6 to evaluate the normalised Igusa invariants I_2, I_4, I_6, I_{10} on the isogenous curve (as in Section 5.6.5). This gives us the product $(\lambda_1\lambda_2)^2$ of the inversible eigenvalues.
Streng's invariants used to compute modular polynomials have bad reduction modulo 2. So in [MR21, Theorem 2.4] we use Igusa's covariants $J_2, J_4, J_6, J_8, J_{10}$ which are defined over $\mathbb{Z}$ to define arithmetic Igusa invariants j_1, j_2, j_3 over each affine of the cover $\mathcal{A}_2[J_2^{-1}J_{10}^{-1}]$, $\mathcal{A}_2[J_4^{-1}J_{10}^{-1}]$, ... $\mathcal{A}_2[J_8^{-1}J_{10}^{-1}]$ (minus the singular locus). In particular, the reduction of $\mathcal{A}_2[J_2^{-1}J_{10}^{-1}]$ modulo 2 corresponds to hyperelliptic curves of type $(1,1,1)$, which are exactly the curves whose Jacobians are ordinary. Using the arithmetic invariants j_1, j_2, j_3 for this locus, we can define modular polynomials with good reduction modulo 2, this allows us to lift abelian surfaces in characteristic two.

(v) We also explain in [MR20a], when A has rational real multiplication for which we have already computed Hilbert modular polynomials, how to use an Hilbert modular correspondance instead (in any dimension g). Indeed the Frobenius preserves real multiplication by assumption, so it suffices to lift it via the Hilbert modular polynomial Φ_p rather than the Siegel one, since it is of smaller degree.
We can improve this if p splits in the real order $\mathcal{O}$. Assume $g = 2$ for simplicity, then if p splits into two totally positive primes p_1p_2 in $\mathcal{O}$, $A[p]_{\text{ét}}$ decomposes into $A[p_1]_{\text{ét}} \oplus A[p_2]_{\text{ét}}$. If we let $A' = A/A[p_1]_{\text{ét}}$, rather than solving for $\Phi_p(J(\widetilde{A}), \sigma(J(\widetilde{A})) = 0$, we solve for $\Phi_{p_1}(J(\widetilde{A}), (J(\widetilde{A}')) = 0, \Phi_{p_2}((J(\widetilde{A}'), \sigma(J(\widetilde{A}))) = 0$. This allows to work with modular polynomials of degrees $p+1$ rather than $O(p^2)$. For the initialisation, if $J(A')$ is the reduction of $J(\widetilde{A}')$ modulo p we find it via a gcd on $\Phi_{p_1}(J(A), X)$ and $\Phi_{p_2}(\pi(J(A)), X)$. For the Newton iteration, via linear algebra to eliminate the variables coming from $\widetilde{A}'$, we also reduce to doing an Artin-Schreier lifting.

We will see in Sections 6.4 to 6.6 how to get either the determinant or the full matrix of the action of the $\hat{\Sigma}_q$ on

the tangent space (via a norm of the action of the small Vershiebung $\hat{\Sigma}_p$). From the full matrix, we can recover χ_ψ (using the notations of Section 5.5.5) at some p-adic precision m, and from the Weil bounds we need to work with $m = gd/2 + O(1)$. When we only have the determinant, we recover χ_ψ via an LLL algorithm, but this requires to increase the precision, and does not always allow to recover χ_ψ (see the discussion at the end of Section 6.4).

6.4 COMPUTING THE ACTION ON TANGENT SPACE WITHOUT LIFTING ISOGENIES (REVENGE OF THE SITH)

At the time the canonical lift algorithms were extended to abelian varieties, the extension of Vélu formula to abelian varieties of Chapter 4 were not known.

So the strategy was different than lifting the kernel and computing the isogeny. Indeed even Mestre's AGM algorithm for elliptic curve did not follow this approach. As we have seen in Section 6.3, the theta duplication formula can be used to lift the modular invariants $\theta_i(A)/\theta_0(A)$. But the duplication formula is also well defined on the modular forms $\theta_i(A)$. Analytically the version of the duplication formula which express $\theta_i(0,\tau/2)$ in term of the $\theta_i(0,\tau)$ encodes the normalised isogeny while the one which express $\theta_i(0,2\tau)$ encode the isogeny acting by $z \mapsto 2z$ on the tangent space.

This means that once we have lifted A to $\widetilde{A}$, doing a cycle of d 2-isogenies using the affine duplication formula encode the action of $\hat{\Sigma}_q$ on the tangent space. Here we only recover the determinant of the action, since the theta functions are scalar modular forms.

This is beautifully explained in [Mes02]: starting with a arbitrary lift $\theta_i^{(0)}(0)$ to $\mathbb{Z}_q$ of the theta null point of A, and iterating the duplication formula to get points $\theta_i^{(j)}(0)$, then not only do the (level $\Gamma(2,4)$) modular invariants $\theta_i^{(mj+k)}(0)/\theta_0^{(mj+k)}$ converges (when $j \to \infty$) to the modular invariants of $\Sigma^k\widetilde{A}$, but the theta modular forms allows to recover the determinant $u := \det M$ via $\sigma^k(u) = \theta_i^{(mj+k)}(0)/\theta_i^{(m(j+1)+k)}(0)$. (The quotient is done the other way around from Satoh's algorithm, because Mestre uses a form of the duplication formula that normalises the action on differential to the multiplication by 2 rather than 1.)

In practice, lifting is faster when done via a Newton process, and rather than computing the full (affine) isogeny cycle, we just do one step of the affine duplication formula, and take a norm. A slight difficulty is that since the θ_i are of level $\Gamma(2,4)$, one may need (depending on the choice of normalisations) to first compute an isomorphism of the level structure too. In other words, if $\theta_i^{(1)}(\widetilde{A})$ denotes one step of the duplication formula for the normalised Verschiebung from $\theta_i(\widetilde{A})$, we may first need to find an automorphism such that $\theta_i^{(1)}(\gamma\widetilde{A})/\theta_0^{(1)}(\gamma\widetilde{A}) = \sigma^{-1}\left(\theta_i(\widetilde{A})/\theta_0(\widetilde{A})\right)$ (for all i) before recovering the common scalar $u = \theta_i^{(1)}(\widetilde{A})/\sigma^{-1}(\theta_i(\gamma\widetilde{A}))$ (for any i).

This is not hard but an easier solution is to simply evaluate a modular form $\mathfrak{g}$ of level 1 and weight m, like the trace of all θ_i^{2m}. Since $\mathfrak{g}$ is of weight $\det^m$, we then recover $(\lambda_1\cdots\lambda_g)^m$ as $N_{\mathbb{Q}_q/\mathbb{Q}_p}\left(\mathfrak{g}(\widetilde{A}_1)/\mathfrak{g}(\sigma^{-1}\widetilde{A}_0)\right)$.

We can use this approach in any situation where we have a modular correspondance not only on modular invariants but also on modular forms. We will call this an affine modular correspondance, and this parametrizes the values $(G(A,w_A),G(B,w_B))$ where $G=(g_1,\dots,g_j)$ is given by some modular forms of weight m, $f: A \to B$ is an ℓ-isogeny (we will use $\ell = p$), and w_A, w_B are normalized in a suitable way, eg by $f^*w_B = w_A$. All theta modular correspondances used in Section 6.3 are naturally affine (see also Sections 4.6 and 5.2). For instance, the same techniques as above hold for the modular correspondance $\pi_1 \times \pi_2 : \mathcal{A}_{g,np} \to \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$: we compare a modular form $\mathfrak{g}$ on $\pi_2(\widetilde{A})$ with $\mathfrak{g}$ on $\sigma^{-1}(\pi_1(\widetilde{A}))$.[4]

We have also seen in Section 5.3.6 that we can define affine modular polynomials using modular forms rather than modular invariants. Using these affine modular polynomials in [MR20a] could have dispensed us from lifting the kernel of the Verschiebung and then computing the isogeny, but at the time I had not yet realised that we could use and compute affine modular polynomials in the Igusa invariants I_2, I_4, I_6, I_{10} directly. But see Section 6.6 for an even better strategy.

The method of Section 6.2.5 can also be seen as an avatar of this principle: using the notations of this Section, when we only compute the coefficients a_1, b_1 of the normalised isogenous elliptic curve (without computing the isogeny itself), and we compare them to $\sigma^{-1}(a)$, $\sigma^{-1}(b)$, we are comparing modular forms of weight 4 and 6 respectively, hence we only recover the action of M^2.

[4] Since our theta null points are of the same level, the use of $\mathfrak{g}$ is not required. But in [CL08a], since they did not have a version of π_1 or π_3, they were comparing a theta null point of level n with one of level np, so using modular forms of level 1 was necessary to get a meaningful computation. A nice trick is that since they have a modular correspondance of degree p^2, using a modular form of weight $1/2$ allows to recover the product of eigenvalues; a form of weight 1 would only have given their squares. We also remark that since we are taking a norm anyway afterwards, we could also take the norm of $\mathfrak{g}(\pi_2(\widetilde{A}))/\mathfrak{g}(\pi_1(\widetilde{A}))$.

A drawback of this approach is that this only recovers the element $\lambda = \lambda_1 \cdots \lambda_g + q/\lambda_1 \cdots \lambda_g$. This element is a root of what is called the symmetric characteristic polynomial of the Frobenius χ_π^{sym} in [Rito3, § 4.2.2]. So we may try to recover χ_π^{sym} from λ via an LLL algorithm. Then [Rito3, § 4.2.2] explain how we may recover (in good cases, like when χ_π is irreducible, ie $A/\mathbb{F}_q$ is absolutely simple) χ_π from χ_π^{sym}. A difficulty is that we may not always recover χ_π^{sym} from λ, because this polynomial is not always irreducible (even if χ_π is). This does not happen when $g \leq 3$, but Mestre gives an exemple in [Meso2] with $g = 4$.

Thus the approach of this Section does not always allow to recover enough information. Even in the cases it does, the LLL step to recover χ_π^{sym} requires to work with precision $m = Cgn/2 + O(1)$ for some constant C depending on g while we would like to work in precision $m = gn/2 + O(1)$. For a bound on C, see [LLo6, § 5.4; CLo8a, p. 19]. For instance, when $p = 2$, we have $C = 3/2$ with $g = 2$ and $C = 6$ with $g = 3$.

6.5 Computing the action on tangent space via lifting the isogeny (a new hope)

To get a better precision bound and recover χ_π in all cases, going back at the root of Satoh's algorithm for elliptic curves, the straightforward solution is to compute the isogeny corresponding to $\hat{\Sigma}$.

We need to recover the matrix M of $\hat{\Sigma}^*$ acting on the basis $\sigma^{-1}(w_A)$ and w_A (or alternatively on w_A and $\sigma(w_A)$). Then taking the characteristic polynomial P of the norm of M, $\chi_\pi = P(X)X^g P(q/X)$. In practice it is often easier to compute the matrix M as follow: we first compute the normalised isogeny $\hat{\Sigma} : (A, w_A) \to (A', w_{A'})$. By abuse of notation, we will denote $w_{A'} = \hat{\Sigma}_* w_A$ when $\hat{\Sigma}^* w_{A'} = w_A$ (ie $w_{A'} = \Sigma^* w_A/p$). We let $F : A' \to \sigma^{-1}A$ be an isomorphism, then M is given by the action of F^* on the basis $\sigma^{-1}(w_A), w_{A'}$.

6.5.1 *Isogeny induced by the modular correspondance*

When we said in Section 6.4 that the alternative method of using modular functions to recover the action on the tangent space was used because Vélu's like formulas were not know in higher dimension, this is not quite true. Indeed the theta duplication formula also gives $\theta_i(2z, 2\tau)$ in function of the $\theta_i(z, \tau)$, so gives equations for the 2-isogeny. More generally the modular correspondance $\pi : \mathcal{A}_{g,np} \to \mathcal{A}_{g,n} \times \mathcal{A}_{g,n}$ also encodes the p-isogeny.

The fact that the theta modular correspondance of level p explicitly gives the p-isogeny is of course well known, but strangely (as far as I know), it seems this was not used to get the full action on the tangent space until [LR20b].

Letting $A \in \mathcal{A}_{g,np}$ and fixing a basis of differentials w_A, since π gives the correspondance $\mathcal{X}_{g,np} \to \mathcal{X}_{g,n} \times \mathcal{X}_{g,n}$ explicitly we can keep track of the differentials and compute $(\pi_1(A), \pi_{1,*}(w_A))$ and $(\pi_2(A), \pi_{2,*}(w_A))$. So to recover the action on differentials at level m, it suffices to compute an isomorphism F between $\pi_2(A^\sigma)$ and $\pi_1(A)$ and compute the matrix M such that $F^*\pi_{1,*}(w_A)) = M\pi_{2,*}(\sigma(w_A))$. The action of the Frobenius is then recovered from the norm of M as usual. (Working with a modular correspondance of degree p^2 we would act by σ^2 instead, and we only recover the characteristic polynomial of π_q^2 at the end if d is even.)

6.5.2 *Recovering the matrix on tangent space over the Kummer varieties*

Apart from this trivial remark, the main interest of [LR20b] is that we explain how to use this strategy with theta functions of level 2 rather than level 4. As shown in Section 2.12 this only encodes the Kummer variety rather than the abelian variety. We want to recover the action on tangent spaces by working only on the Kummer varieties.

Since this has applications others than point counting, we treat the general case of $f : A \to B$ an isogeny. We want to recover $df : T_0A \to T_0B$ while working over the Kummer varieties K_A and K_B.

If we had a rational point P on A not of 2-torsion we could compute the action on tangent space of K_A at P since it is isomorphic to the tangent space of A at P, but we do not want to assume that. (In the cryptographic setting we are given a point anyway, so it would suffice to lift it to $\mathbb{Q}_q$, and compute the tangent space at this lift, but this is less fun.)

We always have the neutral point 0_A, and since K_A is not smooth at 0_A we work with the tangent cone $T_0^c K_A$ at 0_A instead. By general invariant theory, $T_0^c K_A \simeq T_0A/\pm 1$, and more generally since $\hat{O}_{A,0} \simeq k[[x_1, \dots, x_g]]$, we have $\hat{O}_{K_A,0} \simeq k[[x_1, \dots, x_g]]/\pm 1 \simeq \text{Sym}^2 k[[x_1, \dots, x_g]]$, so $T_0^c K_A \simeq \text{Spec Sym}^2 k[x_1, \dots, x_g]$. A concrete model of $\text{Spec Sym}^2 k[x_1, \dots, x_g]$ is given by $U = \text{Spec}\, k[u_{ij}]/(u_{ij}u_{kl} - u_{ik}ujl)$ where u_{ij} represents $x_i x_j$. Given the equations of K_A, we can compute $T_0^c K_A$ and then an isomorphism of $T_0^c K_A$ with U (or a quadratic twist of U if needed). This isomorphism allows us to get a basis of T_0K_A which is the Sym^2 of a basis of A. We can then recover $\text{Sym}^2 M$ where M is the matrix acting on tangent spaces of the isogeny $f : A \to B$, hence recover $\pm M$. We

refer to [LR20b] for more details, and mention that the isomorphism between $T_0^c K_A$ and U is computed using the arithmetic of Section 2.12.1.

6.5.3 *Lifting the kernel*

It remains to explain how to compute the isogeny when we compute canonical lifts via modular polynomials or theta modular polynomials, ie when we stay in level n and do not go to level np. In this case we simply follow Satoh's original algorithm: we lift the kernel of $\hat{\pi}$ and use Chapter 4 to compute the isogeny. Like in the dimension 1 case, the Verchiebung has many lifts, but only one which is unramified. For instance for abelian surfaces, letting $\langle e_1, e_2, f_1, f_2\rangle$ be a symplectic basis of $\widetilde{A}[p]$ where e_1, e_2 reduces to étale points (so live in an étale extension) and f_1, f_2 to multiplicative points (so live in a tamely ramified extension), $\langle f_1, f_2\rangle$ is the only kernel reducing to the Frobenius, there are $\ell(\ell+1)$ kernels similar to $\langle e_1, f_2\rangle$ which reduce to a kernel with p-rank 1, and ℓ^3 kernels which reduce to the Verschiebung, but $\langle e_1, e_2\rangle$ is the only unramified one.

We lift the kernel by lifting its geometric points (it suffices to lift generators). Ideally we would like to lift the whole kernel at once as is done in Satoh's algorithm. This would requires to compute a triangular representation of $\widetilde{A}[p]$ and $A[p]$ (as in Section 5.5.1), to lift univariate polynomials. For our computations with small p this was not worth the hassle, and we will see in Section 6.6 how to bypass this step anyway.

So we simply compute points in $A[p]_{\text{ét}}$, write the equation $pP = 0$ formally in $\widetilde{A}$ (or better $(p'+1)P = -p'P$ when $p = 2p'+1$), and lift via Newton iterations, as in Section 6.2.4. Note that we only need a triangular representation of $A[p]$ to find P, any system of equations of $\widetilde{A}[p]$ is enough for a Newton lifting.

If $\widetilde{A}$ is embedded into $\mathbb{P}^N$, we get a polynomial system of equations in N variables. Since $\widetilde{A}/\mathbb{Z}_q$ is smooth the equations are smooth over $\mathbb{Z}_q$, and since the action of $P \mapsto [p]P$ is given by p on the tangent space $T_0 A$, this means that the Jacobian of this polynomial system has for Smith normal form the diagonal $(1, \dots, 1, p, \dots, p)$ with the factor p repeated g times (they correspond to the embedding of $T_0 A$ into $\mathbb{P}^N$). In particular, if J is the Jacobian of the polynomial system, J^{-1} is of valuation -1, so Newton's algorithm converge as soon as we have a solution at precision 3. Indeed the analysis of Section 6.2.4 is exactly the same in the multivariate case. However reaching precision 3 is harder than in the univariate case because the theory of multivariate Newton polygons is less helpful, so we bootstrap the Newton lifting by solving the quadratic system directly (via a Grobner basis). In practice this initialisation step is sufficiently fast. Due to the nature of the Jacobian of the system, not all variables and equations are at the same precision, so if we do the linear change of variable induced by the Smith normal form we can keep track of the variables that naturally have better precision. We could also use the tools to track the p-adic precision developed in [CRV18]. Geometrically this is explained as follow: lifting P means finding $\widetilde{P} \in \mathbb{P}^N$ such that $\widetilde{P} \in \widetilde{A}$ and $\widetilde{P} \in \widetilde{A}[p]$, and only the second step involves a loss of precision.

We refer to [MR20a] for more details on how to lift the points in the kernel of the Verschiebung for abelian surfaces given by their theta model of level $m = 2$.

Remark 6.5.1. By the same method we can lift points of ℓ-torsion, ℓ prime to p. In fact this case is easier since $A[\ell]$ is étale over $\mathbb{F}_q$, the tangent map is inversible and Newton iterations converge immediately.

The case of lifting abelian surfaces in characteristic two is special. We may assume that A is a Jacobian, $A = \mathrm{Jac}(C)$ (otherwise A is a product of elliptic curves which we may lift directly). Since A is ordinary, C is a genus 2 curve of type $(1,1,1)$ following the terminology of [Igu60]. We lift C using its universal normal form (a form introduced by Igusa valid in any characteristic), see [MR21, § 4.3]. The Weierstrass points of the curve then encode the 2-torsion. It remains to identify the Verschiebung. In this case, it is easier to identify the Frobenius instead by looking at the divisors which reduce to a principal divisor modulo 2 [MR21, Proposition 4.7], and we recover the Verschiebung as the dual isogeny. We can then uses the Richelot isogeny $\mathrm{Jac}(C) \to \mathrm{Jac}(C')$ (it is not hard to check that it is normalised), so the isomorphism between C' and $\sigma^{-1}(C)$ recovers the matrix of the Verschiebung on tangent space by Remark 6.2.4.

6.6 COMPUTING THE ACTION ON TANGENT SPACE WITHOUT LIFTING ISOGENY (THE EMPIRE STRIKES BACK)

Still it is annoying having to lift the kernel, if only because it requires to compute a nice representation of $A[p]$. We would like to revisit the modular method of Section 6.4 but somehow recover more informations than just the determinant of the action on the tangent space.

Table 6.1: Improvement of the new version of Satoh's algorithm in dimension 1

q	Time (old)	Memory (old)	Time (new)	Memory (new)
11^{1008}	48.5s	512MB	4.5s	128MB
101^{102}	91s	1024MB	9s	128MB
101^{256}	633s	4096MB	26s	128MB
101^{310}	924s	8192MB	35s	256MB
101^{418}	1813s	16384MB	55s	256MB

The solution is obvious: rather than looking for a modular correspondance between scalar modular forms, it suffice to look for a modular correspondance for vectorial modular forms, using Key Idea 6. Then we can use the exact same strategy as in Section 6.4 using the vectorial modular form.

But if j is a modular function, dj is a vectorial modular form of weight Sym^2 by the Kodaira-Spencer isomorphism, see Section 5.4.2. And differentiating the modular polynomial $\Phi_p(J_A, J_B) = 0$ exactly gives us the modular relation between dJ_A and dJ_B we are looking for. And we need the modular polynomial Φ_p for lifting A anyway.

This gives the following strategy: we lift $\widetilde{A}$, compute $dJ(\widetilde{A}, w_{\widetilde{A}})$ for any rational differential basis $w_{\widetilde{A}}$ of $\widetilde{A}$, plug it into the differentiation of the modular polynomial (ie look at the tangent space of the modular equation) to get $dJ(\sigma^{-1}\widetilde{A}, w_{\sigma^{-1}\widetilde{A}})$ where $w_{\sigma^{-1}\widetilde{A}}$ is the normalised differential. We compare with the value of $\sigma^{-1}(dJ(\widetilde{A}, w_{\widetilde{A}})) = dJ(\sigma^{-1}\widetilde{A}, \sigma^{-1}(w_{\widetilde{A}}))$ to get $\mathrm{Sym}^2 M$ where M is the matrix of the action of $\hat{\Sigma}$ on the differentials $w_{\widetilde{A}}, \sigma^{-1}w_{\widetilde{A}}$. We recover M (up to a sign), and compute the norm as usual to get the matrix of $\hat{\Sigma}_q$, hence the characteristic polynomial χ_π.

In fact, looking at the equation shows that we can dispense with an explicit version of the Kodaira-Spencer isomorphism. To get the normalised value $dJ(\sigma^{-1}\widetilde{A}, w_{\sigma^{-1}\widetilde{A}})$, we solve the system

$$\partial\Phi_p/\partial X(J, \sigma^{-1}(J))dJ(\widetilde{A}, w_{\widetilde{A}}) + \frac{1}{p}\partial\Phi_p/\partial Y(J, \sigma^{-1}(J))dJ(\sigma^{-1}\widetilde{A}, w_{\sigma^{-1}\widetilde{A}}) = 0.$$

We remark that while $\partial\Phi_p/\partial Y(J, \sigma^{-1}(J))$ is not inversible modulo p, $\frac{1}{p}\partial\Phi_p/\partial Y(J, \sigma^{-1}(J))$ is, so there is at most one bit loss of precision. This gives us

$$\mathrm{Sym}^2 M = \sigma^{-1}(dJ^{-1}(\widetilde{A}, w_{\widetilde{A}}))dJ(\sigma^{-1}\widetilde{A}, w_{\sigma^{-1}\widetilde{A}}) \tag{6.1}$$

$$= \sigma^{-1}(dJ^{-1}(\widetilde{A}, w_{\widetilde{A}}))\left(\frac{1}{p}\partial\Phi_p/\partial Y(J(\widetilde{A}), \sigma^{-1}(J(\widetilde{A})))\right)^{-1}\left(\partial\Psi_p/\partial X(J(\widetilde{A}), \sigma^{-1}(J(\widetilde{A})))\right)dJ(\widetilde{A}, w_{\widetilde{A}}). \tag{6.2}$$

Taking the norm of $\left(\frac{1}{p}\partial\Phi_p/\partial Y(J(\widetilde{A}), \sigma^{-1}(J(\widetilde{A})))\right)^{-1}\left(\partial\Psi_p/\partial X(J(\widetilde{A}), \sigma^{-1}(J(\widetilde{A})))\right)$ gives the same norm as the norm of $\mathrm{Sym}^2 M$ up to conjugation by $\sigma^{-1}\left(dJ(\widetilde{A}, w_{\widetilde{A}})\right)$.

The whole strategy fully dispenses with lifting kernels and only requires the modular polynomial. We only need that there is no non generic automorphisms (so the partial derivative of the modular polynomials are inversible). As far as I am aware, this strategy was never implemented even for elliptic curves. For elliptic curves, since we work with the modular invariant j, unlike Satoh's original algorithm, there is no special cases when $p = 2, 3$ and the elliptic curve equations are different. However, we also only recover the square t^2 of the trace t (since we compute the action of the Sym^2), in other words on the level of moduli we cannot distinguish E from its quadratic twist. The implementation is so simple that it only takes a 50 lines GP script to illustrate this in dimension one:

Example 6.6.1. Let $E : y^2 = x^3 + ax + b$ be the elliptic curve defined over $\mathbb{F}_{5^5}$ which is defined using the irreducible polynomial $x^5 - x - 1$, with $a = x^2 - x - 1$ and $b = x^3 - 1$.

We lift E to $\widetilde{E}$ at precision $(n+5)/2$. We compute $u = \left(\partial\Psi_p/\partial X(\sigma(j(\widetilde{E})), (j(\widetilde{E})))\right)\left(\frac{1}{p}\partial\Phi_p/\partial Y(\sigma(j(\widetilde{E})), j(\widetilde{E}))\right)^{-1}$. The norm v of u is equal to $4 + 3\cdot 5 + 4\cdot 5^2 + 5^3 + 2\cdot 5^4 + O(5^6)$, and we have $v = (\lambda_1\lambda_2)^2$ where λ_1, λ_2 are the inversible eigenvalues. So if t is the trace, $t^2 = v + q^2/v + 2q = 4 + 3\cdot 5 + 4\cdot 5^2 + 5^3 + 2\cdot 5^4 + 2\cdot 5^5 + O(5^6)$, so $t = \pm(3 + 2\cdot 5 + 3\cdot 5^2) = \pm 88$, and we check that the correct value is $t = 88$.

Allombert converted the script to the Pari library, it gains a factor 10 (for $p = 11$) to 25 (for $p = 101$) compared to the older implementation (and it gains a large factor in the memory). We give example of improvements to the

point counting of an elliptic curve over $\mathbb{F}_q$ for several different values of q in Table 6.1. The gains should be even higher for higher dimensions.

As a corollary, we get a quasi-quadratic (in d) algorithm for point counting. This was already announced in [CL08a] using a variant of the modular correspondance from Section 5.2 (see **??** 6.3.1.(ii)), but our algorithm works over the base field directly (no need for a theta structure), does not rely on LLL (and can always reconstruct χ_π, whereas the reconstruction step of [CL08a] needs to assume that P_{sym} is irreducible, which may not be the case even if χ_π is irreducible), and has a much better understood dependency on p (and the dependency on p is better than the one from **??** 6.3.1.(iii)).

Dependency on p of the algorithm.

The main dependency on p is the cost of computing Φ_p, using the results of Chapter 5 notably Remark 5.3.1 we know that it is in $O(p^C)$ for some explicit constant C depending on g.

In fact, we'd rather evaluate it directly (along with its derivative), using Section 5.3.8. We recall that these evaluations cost $\widetilde{O}(D^2E(m\log q + d^2M))$, so the dependency on p is $\widetilde{O}(p^C)$ with $C = g(g+1)+g$ in the Siegel case and $C = 3$ in the Hilbert case. Under Conjecture 5.3.14 (or if $g = 1$), we have $E = 1$ in which cas $C = g(g+1)$ and $C = 2$ respectively.

However we need to be careful that if we work over the representation of $\mathbb{Z}_q/p^m\mathbb{Z}_q$ defined by the Teichmuller lift, then $M = O(m\log p)$. For point counting, we work at precision $m = \Theta(d)$, so this gives a complexity cubic in d. The solution is either to work with a polynomial of small height for the evaluation, and then switch to the Teichmuller representation, or use fast modular composition for the evaluation of σ without relying on the Teichmuller lift. By [KU11], both approaches can be done in quasi-linear time.

This is only interesting when $g > 1$ and the full modular polynomials are too big (at least in the Siegel case) to even be computed. When $g = 1$ we gain the $O(p^3)$ precomputation for Φ_p, and the evaluation goes from $O(p^2md\log p)$ when we have Φ_p to $\widetilde{O}(p^2(dm\log p + d^2M) = \widetilde{O}(p^2d^2)$ with a small M and taking $m = \Theta(d)$. So the soft $\widetilde{O}$ factors are the same, but in practice the evaluations are slower since they hide some extra logarithmic factors. Since the lift is in $\widetilde{O}(p^2d^2)$ anyway, if p is sufficiently large compared to d that the gain of $O(p^3)$ becomes worthwhile, we are better of with Kedlaya's algorithm in this case.

Remark 6.6.2. It is thus interesting to compare this algorithm to Kedlaya's algorithm. Kedlaya has a much better dependency on p and g but a worse dependency on the degree d: $\widetilde{O}(pd^3)$ [Ked01; GG03; Ked16], or $\widetilde{O}(p^{1/2}d^{3.5})$ [Har07].[5] It was originally given for elliptic curves, but has been extended to all curves [Tui16], and even to smooth projective hypersurfaces [CHK19] (but has only been really used for surfaces). By contrast, our improved version of Satoh's algorithm can only handle abelian varieties, but works even on abelian varieties which are not Jacobians.

If $g = 1$, the full complexity to lift to precision m (for point counting, $m = \Theta(d)$) is $\widetilde{O}(p^3 + p^2dm)$. The p^3 is for the computation of Φ_p, but it may be thought of as a precomputation since it does not depend on E. We have seen that by evaluating Φ_p directly, it actually can be reduced to $\widetilde{O}(p^2dm)$.

It is plausible that, when $g = 1$, there exists an algorithm that is in $\widetilde{O}(p^2d + pd^2 + p^{3/2}dm)$ (or even $\widetilde{O}(p^2d + pd^2 + \sqrt{p}dm)$, provided that the étale part of $E[p]$ lives in a small extension). Namely, we lift E to a candidate $\widetilde{E}_0$ along with an étale point of p-torsion $\widetilde{P}_0$. Modulo p, this point P_0 live in an extension e of degree at most $p-1$ and is computed in time $\widetilde{O}(p^2d + pd^2)$: $\widetilde{O}(p^2d)$ to compute the division polynomial via the recurrence formula, then we factorize its separable part which is of degree p in time $\widetilde{O}(p^{1.5}d + pd^2)$ by [KU11]. We then lift the point to $\widetilde{P}_0 \in \widetilde{E}_0$ in time $\widetilde{O}(mde)$ using Remark 6.2.3, since we know how to evaluate $[p]$ efficiently. We can then compute the isogeny via the fast version of Vélu's formula from [BDL+20] in time $\widetilde{O}(\sqrt{p}edm)$, and check if we get $\sigma^{-1}(\widetilde{E})$, and use a Newton iteration to converge, using Remark 6.2.3 again. Once P_0 is computed, the cost to compute $\widetilde{E}$ at precision m is then $\widetilde{O}(\sqrt{p}edm)$.

6.7 CONCLUSION AND PERSPECTIVES

We have seen that using modular correspondances or modular polynomials, we can compute a canonical lift in time quasi-linear in the precision. As Section 6.6 shows, the modular polynomials also allows to get relations on the derivative of the modular invariants, hence get the Sym^2 action of the lift of the Frobenius on tangent space.

[5] Here we assume g fixed for simplicity. The dependency on g of Kedlaya's algorithm is polynomial in g, whereas for Satoh's algorithm it is polynomial in p^g or even p^{g^2}, at least if we don't assume that we know the real multiplication so that we can use Hilbert modular polynomials rather than Siegel modular polynomials.

Using the modular polynomial Φ_p is not the only way to compute canonical lifts. Indeed we can also lift other isogenies between Galois images (see [Koh08, § 3.1]). This can for instance be used to lift abelian varieties when their modular invariant is in $\mathbb{F}_{p^2}$. (We could also take a non rational isogeny, lift the isogenous variety, and then redescend, or simply boostrap the Newton step to enough precision.)

Lifting (not necessarily canonical) is an important tool, which we already used in Section 5.3.8 to evaluate modular polynomials and which we will use again in Section 7.4 to compute class polynomials. We have seen in Section 5.3.8 that it would also be nice to know how to compute non-canonical lifts of ordinary abelian varieties (given their local moduli coordinates).

I am also interested in computing other lifts than (canonical) lifts of ordinary abelian varieties (they exist by smoothness of the moduli space), in view of applications to the security of SIDH. This is the reason I mentioned the full Serre-Tate correspondance in Section 6.2.1. Related to this, we know that the p-divisible group $A(p)$ (hence its Dieudonné module $\mathbf{D}_p(A)$) encodes all the required information on A (the formal group law of A is the formal group associated to its connected part, the endomorphisms of A can be read of it by Tate's isogeny theorem [Tat66]…). It would be interesting to study this in more details, especially for endomorphism rings computations.

7 CLASS POLYNOMIALS

CONTENTS

7.1 INTRODUCTION

We give a quick overview of class polynomial computations in Section 7.2, the interested reader will find more details in [Rob21, Chapter 9]. The class polynomials parametrizes the moduli $\mathcal{A}_{g,\Phi}$ of all abelian varieties with CM by $(O_{\mathcal{E}}, \Phi)$. They also define the Shimura class field $\mathfrak{H}_E$.

There are three methods to compute class polynomials: analytic (ie working over $\mathbb{C}$), p-adic using canonical lifts, and a CRT approach. The CRT approach is briefly described in Section 7.3 (more details are given in [LR13; ER13]). A new result of this Chapter is that we give a quasi-linear algorithm to compute class polynomials via the p-adic approach in Section 7.4, under the (cheating) assumption that $\mathfrak{H}_E$ has a sufficiently small prime $\mathfrak{P}$ (of ordinary reduction), or that we are already given an abelian variety with CM by $O_{\mathcal{E}}$ over $\mathbb{F}_{\mathfrak{P}}$.Some perspectives are in Section 7.5.

7.2 AN OVERVIEW OF CLASS POLYNOMIAL COMPUTATIONS

7.2.1 *The main theorem of complex multiplication*

For the theory of complex multiplication, we refer to [Rob21, Chapter 9] and its references. If $\mathcal{E}$ is a CM field of degree $2g$, the class polynomials encode the modular invariants of all abelian varieties with complex multiplication by $O_{\mathcal{E}}$.

In the case of elliptic curves (ie $g = 1$), it is well known that these elliptic curves form a torsor under the action of $\mathrm{Cl}(O_{\mathcal{E}})$ by isogenies (where the action of $[I]$ on E is given by $E/E[I]$). And if E has complex multiplication, the main theorem of complex multiplication relates the action of $I \in \mathrm{Cl}(O_{\mathcal{E}})$ on E with the Galois action of I given by class field theory (ie by the Artin Symbol) on $j(E)$.

In particular the class polynomial gives an equation for the Hilbert class field of E. This can be generalised to construct ring class fields (by looking at the j-invariant of elliptic curve with CM by an order O of $\mathcal{E}$), and ray class field (by evaluating modular functions of level n, typically the x-coordinate of an n-torsion point). See [Sil94] and [ER13, § 1.2.1] for a very brief overview.

Shimura extended these results to (principally polarised) abelian varieties with complex multiplication. Here, due to the polarisation, a slightly different class group acts on such abelian varieties A, which I termed the Shimura class group in [LR13]. The whole theory of complex multiplication extends to non maximal orders, and even to CM algebras (ie product of CM orders, but this does not occur for simple abelian varieties with complex multiplication anyway). In this Chapter, I focus on the maximal case for simplicity.

A brief summary of the main theorem of complex multiplication is given in [Rob21, Chapter 9]. Let $\mathcal{E}$ be a CM field, Φ a CM type, $\mathcal{F}$ its real subfield, $O_{\mathcal{E}}$ the maximal order, $\mathcal{E}^r$ the reflex field, $\mathcal{F}^r$ its real subfield. Let $\mathcal{A}_{g,\Phi}/\mathbb{Q}$ be the moduli (of dimension 0) of all abelian varieties with CM by $(O_{\mathcal{E}}, \Phi)$. For the purpose of this Chapter, we just need to know that the points of $\mathcal{A}_{g,\Phi}$ are defined over a class field $\mathfrak{H}_E$ of $\mathcal{E}^r$ (defined in [Rob21, Theorem 9.4.1]), and that the splitting of $\mathcal{A}_{g,\Phi}$ into $\mathcal{E}^r$-irreducible components under the Galois action corresponds to the action of

the image of Cl_{K^r} by the type norm in the Shimura class group [Rob21, Corollary 9.4.2]; and these components are defined over $\mathcal{F}^r$. Furthermore, the Taniyama-Shimura formula [Rob21, Theorem 9.5.1] describes the characteristic polynomial χ_π of the reductions modulo $\mathfrak{P}$ of the abelian varieties with CM by $O_\mathcal{E}$ (they have potential good reduction everywhere). Furthermore, if the reduction is ordinary (eg if p splits completely in $\mathcal{E}$), then there is a bijection between the reduction of the abelian varieties in $\mathcal{A}_{g,\Phi}$ and the abelian varieties in $\mathbb{F}_q$ with CM by $O_\mathcal{E}$, see [Rob21, Section 9.5]. These results thus gives an explicit description of the abelian varieties with CM over number fields and finite fields.

7.2.2 *Strategies to compute the Shimura class polynomial*

Since $\mathcal{A}_{g,\Phi}$ is (geometrically reduced) of dimension 0, it suffices to enumerate all its geometric points and construct the corresponding polynomial representation.

We typically use a Hecke representation, like for modular polynomials. Let $J = (j_1, \dots, j_n)$ be modular invariants. If j_1 separate the points, then $H_{\Phi,1}(X) = \prod_{A\in\mathcal{A}_{g,\Phi}}(X - j_1(A))$, and $j_i(A)H'_{\Phi,1}(j_1(A)) = H_{\Phi,i}(j_1(A))$ with

$$H_{\Phi,i}(X) = \sigma_{A\in\mathcal{A}_{g,\Phi}} j_i(A) \prod_{B\in\mathcal{A}_{g,\Phi}, B\neq A} (X - j_1(B)).$$

Here we assume that $j_1(A)$ separates the points for simplicity, see eg [Str10, § III.5] for a general triangular definition when this is not the case.

To get an irreducible component of $\mathcal{A}_{g,\Phi}$ over $\mathcal{F}^r$, it suffices to find one point of this component and then to look at the action of the image of the type norm in the Shimura class group to get the other ones. So enumeration is easy once we have found a starting point.

There are three main strategies:

1. Via complex approximation. This strategy constructs the period matrices of the abelian varieties in $\mathcal{A}_{g,\Phi}(\mathbb{C})$, and then evaluate modular forms on these matrices. In dimension 1 and 2, we can use fast evaluation of theta constants to get fast evaluation of the modular invariants. This allows to get an approximation of the class polynomials in quasi-linear time in their size (for a given precision), and if the precision is large enough a rational reconstruction (over $\mathbb{Q}$ or $\mathcal{F}^r$ depending on what type of irreducible component we want to compute) is done.

2. Via p-adic approximation. This is the same strategy as via complex approximation, except that the class polynomials are computed via p-adic approximations. We detail this in Section 7.4.

3. Via a CRT approach. This is similar to the p-adic method, except that rather than working with one prime p at precision N we work with several primes p_i at precision 1 (ie in $\mathbb{Z}/p_i\mathbb{Z}$ or $O_{K^r}/\mathfrak{p}_i$), and use the CRT to recover the class polynomials. See Section 7.3

In the complex case, a starting point can be taken to be the lattice $\Phi(O_\mathcal{E})$. However we will see that both for the p-adic and CRT approach getting a maximal variety $A/\mathbb{F}_q$ with CM by $(\mathcal{E}, \Phi)$ is harder.

We briefly mention the complexity of the analytic approach, which essentially boils down to being able to evaluate the modular invariants associated to the CM points in quasi-linear time. When $g = 1$ the analytic approach is studied in [Eng09b] and for $g = 2$ in [Str10] using the naive but rigourous evaluation of theta constants and [ET14] using Dupont's fast but heuristic algorithm to get a quasi-linear algorithm. Letting N be the degree of the class polynomials and H their height, using the very heuristic putative fast algorithm to evaluate theta functions of [Lab16, § 7.4] (see Section 5.3.4) along with [NSV11; NS16] for fast rational reconstruction, the analytic method gives an $\widetilde{O}(NH)$ algorithm. Using the proved "improved naive" algorithm for the evaluation, we get an $\widetilde{O}(NH^{1+g/2})$ algorithm instead.

7.3 Enumerating abelian varieties with CM over a finite field

Fix a primitive CM type $(\mathcal{E}, \Phi)$, and $A \in \mathcal{A}_{g,\Phi}$. Let $\mathfrak{p}$ be a prime of $O_{\mathcal{F}^r}$, $p = \mathfrak{p} \cap \mathbb{Z}$, $\mathfrak{P}$ a prime of $O_{\mathfrak{H}_E}$ above $\mathfrak{p}$, and $\mathbb{F}_q := \mathbb{F}_\mathfrak{P}$ with $q := N(\mathfrak{P}) = p^{f_\mathfrak{P}}$. Assume that p is unramified in $O_\mathcal{E}$ and $\mathfrak{p}$ in $\mathfrak{H}_E$.

Let $\mathfrak{a} = N_\Phi(\mathfrak{p})$. Then by the Taniyama-Shimura formula (see [Rob21, Theorem 9.5.1]), $f_\mathfrak{P}$ is the order of $\mathfrak{a}$ in $\mathrm{Cl}(O_\mathcal{E})$, and the Artin symbol $\left(\frac{\mathfrak{P}}{\mathfrak{p}}\right)$ corresponds to the Galois action of the small Frobenius $\pi_\mathfrak{p}$ on $A_\mathfrak{P}/\mathbb{F}_q$ which is an $\mathfrak{a}$-isogeny. In particular, the Frobenius $\pi_q = \pi_\mathfrak{P}$ seen as an element $\pi \in O_\mathcal{E}$ satisfy $\pi_\mathfrak{P} O_\mathcal{E} = N_{\mathfrak{H}_E,\Phi}(\mathfrak{P})$.

We say that $\mathfrak{p}$ is an ordinary prime if $A_{\mathfrak{P}}$ is ordinary (and simple) (since the elements of $\mathcal{A}_{g,\Phi}$ are isogenous, this does not depends on the representative). By [Rob21, Section 9.5] the reduction of the points of $\mathcal{A}_{g,\Phi}$ modulo $\mathfrak{p}$ are exactly given by the abelian varieties with CM by $(\mathcal{E}, \Phi)$ and defined over $\mathbb{F}_q$, with q given by the Taniyama-Shimura formula above. By [Sug14, Theorem 1.2] $\mathfrak{p}$ is an ordinary prime if p splits completely in $\mathcal{E}$, furthermore in this case, by [Mil06, Corollary 8.3], the CM type can be read of from the decomposition of $\pi_{\mathfrak{P}} O_{\mathcal{E}}$ (as the set of ϕ factoring through $\mathcal{E}_v$ for the places $v \mid \pi_{\mathfrak{P}}$).

We briefly explain how to find all abelian varieties with CM in $\mathbb{F}_q$.

1. Compute the characteristic polynomial of π using the Taniyama-Shimura formula. This gives the isogeny class.
2. Find an abelian variety $A/\mathbb{F}_q$ in the isogeny class (ie such that $\mathrm{End}^0(A) := \mathrm{End}(A) \otimes \mathbb{Q} = \mathcal{E}$) by point counting (first sampling some random points and testing if they are of the correct order).
3. Check if $\mathrm{End}(A)$ is maximal and if not compute an isogeny that increases the endomorphism ring.
4. Once we have a maximal abelian variety A, compute the other ones using the action of the Shimura class group $\mathfrak{C}(\mathcal{E})$.

This is the strategy implemented in [LR13], more details are given in [ER13, § 5]. Note that at the time we did not have cyclic isogenies or modular polynomials for cyclic isogenies, so for abelian surfaces the going up phase of the algorithm did not always succeed. From the structure of the isogeny graph of Section 5.6.4, ℓ-isogenies are enough to get maximal real multiplication, but afterwards we need cyclic I-isogenies. If I is not principal, this requires modular polynomial for not necessarily principally polarised abelian varieties with real multiplication, see Section 5.7.

The vertical method for endomorphism rings computations

We briefly explain how to compute the endomorphism ring, following [FL08], and then how to go up. We already know that $\mathrm{End}(A) \subset \mathbb{Z}[\pi, \overline{\pi}]$, and $\mathrm{End}(A)$ is stable under the Rosati involution (ie under the complex conjugation). Taking an appropriate basis of $O_{\mathcal{E}}/\mathbb{Z}[\pi, \overline{\pi}]$, we reduce to check if an endomorphism $\alpha \in O_{\mathcal{E}}$ is in $\mathrm{End}(A)$.

Localising, we may assume that α is of order ℓ^m in $O_{\mathcal{E}}/\mathbb{Z}[\pi, \overline{\pi}]$. So $\ell^m\alpha$ is a polynomial in π and $\overline{\pi}$, and is an endomorphism of A, and by the universal property of the isogeny $[\ell^m]$, α is in $\mathrm{End}(A)$ if and only if $(\ell^m\alpha)(A[\ell^m]) = 0$.

In good cases $\#p \nmid O_{\mathcal{E}}/\mathbb{Z}[\pi, \overline{\pi}]$ ([FL08, Proposition 3.7] states that this is always true when $q = p > 3$ and $g = 2$), so $\ell \neq p$ and we may check that $(\ell^m\alpha)(A[\ell^m]) = 0$ by computing it on a basis of the ℓ^m-torsion. Note that since $q\overline{\pi} \in \mathbb{Z}[\pi]$ (we can check this looking at χ_π), $[\mathbb{Z}[\pi, \overline{\pi}] : \mathbb{Z}[\pi]]$ is of index a power of p, so if ℓ is prime to p we may also express $\ell^m\alpha$ as a polynomial of the Frobenius (up to replacing α by $p^x\alpha$). This makes it slightly easier to evaluate (but evaluating $\overline{\pi}$ is not too hard, we just compute $\overline{\pi}(P) = q\pi^{-1}(P)$).

Of course we first test $\ell^{m-1}\alpha$, then $\ell^{m-2}\alpha$ and so on. This requires to compute a basis of $A[\ell]$, $A[\ell^2]$ and so on, which we do using Section 5.6.2. The extension $\mathbb{F}_{q^d}$ defining the geometric points of $A[\ell]$ can be computed from $\chi_\pi \mod \ell$. Then, if $A(\mathbb{F}_{q^d})$ does not contain $A[\ell^2]$, the points of $A[\ell^m]$ are defined over $\mathbb{F}_{q^{d\ell(m-1)}}$. Indeed if $P \in A[\ell^2]$, $\pi^d(P) = P + Q$ with $Q \in A[\ell]$ so $\pi^{\ell d} = P + \ell Q = P$, so all geometric points of $A[\ell^2]$ are defined over $\mathbb{F}_{q^{\ell d}}$ and we conclude by induction. See also [FL08, Proposition 6.3] for a converse when $\mathrm{End}(A)$ is maximal.

Note that if we know $O = \mathrm{End}(A)$ already, we may consider $A[\ell]$ as an $O/\ell O$ module, so look at the order of $\pi \in O/\ell O$. This typically gives smaller bounds on d: [ER13, § 5.5.2], [FL08, Proposition 6.2]. If α is not zero on $A[\ell^m]$, we let i be the smallest integer such that α is not zero on $A[\ell^i]$, then $\alpha(A[\ell^i]) \subset A[\ell]$ and this image gives a candidate for an isogeny to try to increase the endomorphism ring. We refer to [ER13, § 5] for more tricks.

The horizontal method for endomorphism rings computations

Another strategy is to use the horizontal method to compute endomorphisms [BS09; Bis11]. Looking at the Shimura class groups of the lattice of orders between $O_{\mathcal{E}}$ and $\mathbb{Z}[\pi, \overline{\pi}]$, we can look at a group relation (of primes not dividing the index $[O_{\mathcal{E}} : \mathbb{Z}[\pi, \overline{\pi}]]$) valid in one order O but not in its suborders. Starting from a candidate A, on which we already know that $\mathrm{End}(A) \supset O_1$, O_1 a suborder of O, we can follow the isogeny cycle induced by the relation valid on $\mathfrak{C}(O)$ but not $\mathfrak{C}(O_1)$, and then check whether we get back to A or to another variety B. In the later case we know that $\mathrm{End}(A) \not\supset O$, and we can try to find a common ℓ-isogeny (this time with ℓ dividing the index) from B

and A to close the isogeny cycle. This strategy was implemented in [BLR11]. This paper was not published because at the time computing the ℓ-isogeny going up required to compute points in $A[\ell]$ using Section 5.6.2 followed by the isogeny formula of Chapter 4. So it was not really worthwhile compared to the horizontal method (the main gain was that we did not need to compute $A[\ell^m]$ if ℓ^m divides the index, only $A[\ell]$). But now that we have an efficient way to evaluate modular polynomials (especially for abelian surfaces), we should revisit this article: the common isogeneous target is simply given by the gcd of the two modular polynomials evaluated at A and B.

The CRT algorithm

With this strategy we have the following CRT algorithm to compute an irreducible component of $\mathcal{A}_{g,\Phi}$ over $\mathcal{F}^r$ (assuming $g = 2$ for simplicity, the generalisation is immediate): call a prime $\mathfrak{p}_0 \in O_{\mathcal{F}^r}$ a good CRT prime if it is of degree 1 over $\mathbb{Q}$ and it splits completely in $O_{\mathcal{E}^r}$ into $\mathfrak{p}_0 = \mathfrak{p}\overline{\mathfrak{p}}$, and $\mathfrak{p}$ is an ordinary prime of degree 1 in $\mathfrak{H}_E$. Then we may try to find one A with CM by $(O_{\mathcal{E}}, \Phi)$ using the strategy above [LR13, § 2 and § 3], and find the others using the action of the type norm [LR13, § 4]. By sieving on the good CRT primes, we may assume that the index $[O_{\mathcal{E}} : \mathbb{Z}[\pi, \overline{\pi}]]$ is not divisible by too large primes [LR13, § 5.1 and § 6]. We then do a CRT reconstruction of the class polynomial in $\mathcal{F}^r$, doing a LLL step to reconstruct a coefficient $c_i \in \mathcal{F}^r$ from its value in $O_{\mathcal{F}^r}/I$, I the ideal given by the CRT [LR13, § 5.3]. Alternatively, we could work over both each CRT prime $\mathfrak{p}_0$ and their Galois conjugate to get the values $O_{\mathcal{F}^r}/N$, then we just need to do a rational reconstruction over $\mathbb{Q}$ (in the Dihedral case this amount to working with both classes of CM types). But since the CRT step is expansive, it is faster to do the LLL computation, this gains a factor two.

Unfortunately, to get a quasi-linear algorithm to compute the class polynomial, we would need a quasi-linear algorithm to compute its reduction modulo p at each of the CRT primes. But even for abelian surfaces, simply sampling A in the correct isogeny class takes too long, see [LR13, § 5.1]. If we have explicit equation for the Humbert surface (parametrizing real multiplication by $O_{\mathcal{F}}$), we can speed up this step by sampling inside it, but this is still not quite enough to get a quasi-linear algorithm (from back of the enveloppe computations).

Broadly, if the image of the type norm of $\mathrm{Cl}(O_{\mathcal{E}^r})$ in the Shimura class group is of cardinal N, then $N = [\mathfrak{H}_E : \mathcal{E}^r]$ is the degree of the class polynomial. By Cebotarev density theorem we expect a density of $1/N$ prime to split completely in $\mathfrak{H}_E$. Less heuristically, under GRH we can bound the minimal prime splitting totally by $O(\log^2 \Delta_{\mathfrak{H}_E})$ [LO77; Bac90]. Since $\Delta_{\mathfrak{H}_E} = \Delta_{\mathcal{E}^r}^N$ (see [BGL11, § 6.4]), this gives a bound $p = O(N^2)$ for the minimal prime. If the class polynomials are of height H, then we need (neglecting log factors) $\widetilde{O}(H)$ primes, so the largest prime is $\widetilde{O}(N^2 + HN)$.

I don't know of a general height bound for Shimura class polynomials, but in dimension 1 (ie for quadratic imaginary fields) we have $N, H = \widetilde{O}(\sqrt{\Delta_K})$ so the class polynomial is of size $\widetilde{O}(\Delta_K)$. In dimension 2 (ie for quartic CM fields), we have $N = \widetilde{O}(\sqrt{\Delta_0 \Delta_1})$ where $\Delta_0 = \Delta_{\mathcal{F}}$ and $\Delta_1 = \Delta_{K/\mathcal{F}}$. A bound on H given in [Str10, § II.11] using [GL12] is $\widetilde{O}(\Delta_1^{3/2}\Delta_0^{5/2})$, but in practice it seems to be $\widetilde{O}(\Delta_1^{1/2}\Delta_0^{1/2})$ [Str10, Appendix 3], so once again $N \approx H$. Heuristically we expect this to hold again in higher dimension: all points of $\mathcal{A}_{g,\Phi}$ have the same Faltings height by [Col93], and we expect this height to be small compared to the degree by Colmez' conjecture on the Faltings height of CM points [Col93; AGH+18; YZ18], so $H = \widetilde{O}(N)$. In any case, we have $N = O(H)$, so the largest prime will be $\widetilde{O}(HN)$, ie the size of the class polynomial.

But then in dimension g we expect by Honda-Tate to have roughly $O(p^{g(g+1)/4})$ isogeny classes, hence we expect an isogeny class to be of size roughly $O(p^{g(g+1)/4})$ so sampling already costs $O(p^{g(g+1)/4})$ tries.

The CRT algorithm is optimal if $g = 1$. Indeed we don't even need to sample in the isogeny class (but this does speed things up), since we have $p = \widetilde{O}(\Delta_K) = \widetilde{O}(HN)$ and there are $\widetilde{O}(\sqrt{\Delta_K})$ maximal curves already. So directly sampling for E with maximal CM by $O_{\mathcal{E}}$ can be done in time $\widetilde{O}(N)$, then reconstructing the class polynomial modulo p costs $\widetilde{O}(N)$ operations in $\mathbb{F}_p$, and the full class polynomial is reconstructed using $O(H)$ CRT primes, this gives a $\widetilde{O}(HN)$ algorithm. We refer to [Sut11; ES10] for optimisations using isogeny classes and smaller class invariants.

7.4 USING p-ADIC LIFTS TO COMPUTE THE CLASS POLYNOMIALS

Let $\mathfrak{P}$ be a prime of $O_{\mathfrak{H}_E}$ of ordinary reduction, $q := N(\mathfrak{P}) = p^{f_{\mathfrak{P}}}$, $\mathfrak{p} = \mathfrak{P} \cap O_{\mathcal{F}^r}$ and $p = \mathfrak{P} \cap \mathbb{Z}$. We can then compute the class polynomials over $\mathbb{F}_q$, as in Section 7.3 and then use p-adic lifting, using the tools of Chapter 6, rather than via a CRT. For simplicity, we detail the algorithm in the case $g = 2$, but the extension to arbitrary g is immediate.

First we need to find a maximal abelian variety $A/\mathbb{F}_q$, ie one CM point over the residue field $\mathbb{F}_q = O_{\mathfrak{H}_E}/\mathfrak{P}$. Then we use isogenies or modular polynomials to find the N other ones, using small generators of the class group, hence isogenies of small degree (prime to th e index). This second step is quasi linear in the size of the class polynomial over $\mathbb{F}_q$ ([LR13, § 6]). Finally we lift all of them to $\mathbb{Q}_{\mathfrak{P}}$ at precision $m = O(\log H/\log q)$. This is done in time quasi-linear in the precision thanks to Chapter 6. We then reconstruct the class polynomial in $\mathbb{Q}_{\mathfrak{p}}$, and recognize coefficients in $\mathcal{F}^r$ by doing a rational reconstruction (using eg a variant of [NSV11; NS16]). We could bypass this LLL step of dimension $g+1$ by computing the class polynomial in $\mathbb{Q}_{\mathfrak{p}}$ and its Galois conjugate, so that the reconstruction in $\mathcal{F}^r$ boils down to an interpolation.

A key difference of this approach with [GHK+06; CKL08] (which use 2-adic and 3-adic lifting respectively) is that the authors only lift one abelian surface with CM, which gives one root of the class polynomial. They then recover the class polynomial via an LLL computation. Even using a fast LLL variant which is quasi-linear in the needed precision like [NSV11; NS16], this LLL step is not quasi-linear in the required dimension, which depends on N, the degree of the class polynomial, see Remark 5.3.13. By contrast our method reconstructs the class polynomial from all its roots (in other words all the Galois conjugates under $\mathrm{Gal}(\mathfrak{H}_E/\mathcal{E}^r)$), which is more efficient.

To get a quasi-linear algorithm to compute the class polynomials, we first need that p is sufficiently small that computing the Φ_p modular polynomials for lifting does not dominate the complexity step. Alternatively, since we control exactly the Galois action in the CM case, we could use Φ_ℓ modular polynomials for lifting whenever the Galois image $\sigma^i A$ is related to A by an ℓ-isogeny, as discussed in Section 6.3.

From the discussion of Section 7.3, the real difficulty is the initialisation step, ie find a maximal A over $\mathbb{F}_q$. If we have an algorithm to find a maximal abelian variety in time q^C, then we need to find $\mathfrak{P}$ such that $p^{f_{\mathfrak{P}}}$ is in $O((NH)^{1/C})$, where NH is the size of the class polynomials (ie N is the degree and H the height). Depending on $\mathfrak{H}_E$ and C, such a $\mathfrak{P}$ may or may not exist. If we look for a p which stays inert in $\mathfrak{H}_E$. Then p is small, but $q = p^N$ is too large. If we look for p which splits completely, we have seen in Section 7.3 that the smallest such p is heuristically in $O(N)$, and there is a proved bound under GRH of $O(N^2)$, which is too large.

A solution to the initialisation problem is to cheat and start with an abelian variety $A/\mathbb{F}_q$ (and p sufficiently small), and then compute the class polynomials associated to $\mathrm{End}(A)$ (but then we do not have much control on the order). This is the assumption in [GHK+06; CKL08].

A more honest solution would be to somehow recursively compute a stratification of $\mathcal{A}_g$ by moduli spaces of codimension 1 in each other (and containing the CM points), so that sampling could be done more efficiently, ie at the end we would sample in a dimension 1 variety (as is the case for the CM method for elliptic curves) rather than in dimension $g(g+1)/2$.

Another strategy that I would like to explore is, rather than looking at primes of ordinary reduction, to look at primes of supersingular or even superspecial reduction. Then we could try to construct A directly over $\mathbb{F}_q$ as a product of supersingular elliptic curves (or isogenous to such a product). This could help solve the initialisation step both for the CRT and p-adic approaches (but in the p-adic approach this would complicate the lifting).

7.5 CONCLUSION AND PERSPECTIVES

The global strategy to compute class polynomials extend to compute the class polynomials for the more general class fields given by [Rob21, Theorem 9.4.1] (ie class fields of level $\mathfrak{b}$): the moduli space is of dimension 0, so we enumerate all possibilities and evaluate suitable modular invariants at high enough complex precision, or high enough p-adic precision, or high enough "CRT precision".

To get quasi-linear algorithms in the complex analytic approach, we would need a way to evaluate modular invariants of level N (level depending on the class polynomial we want to compute). See Section 5.7 for an approach.

In the p-adic and CRT approaches, since we know how to evaluate isogenies on points, once we have a starting abelian variety (A, P) with P of $\mathfrak{b}$-torsion, we could use isogenies (of degrees prime to $\mathfrak{b}$) to compute the other ones, moving P along the isogenies. Likewise, lifting P does not pose problems (see Remark 6.5.1), so long as we can write equations for $\mathfrak{b}$-division polynomials (or simply the multiplication by $\mathfrak{b}$) and $\mathfrak{b}$ is prime to p (so the $\mathfrak{b}$-torsion is étale and lifting it via Newton iteration behaves well). The hard part is to find one such (A, P) over $\mathbb{F}_q$, as we have seen is the case when $\mathfrak{b} = 1$ already.

However when $g = 1$, the CRT method works well, ie is quasi-linear. It has been used to compute class polynomials [Sut11], and we have used it in [ERS16] to compute ray class fields of quadratic imaginary fields in quasi-linear time. This is a joint work with Enge and Seiler, unfortunately unpublished but the main ideas are in Seiler's master thesis. The main difference in [ERS16] is that while Seiler consider primes that split totally in the Hilbert class field and then compute $\mathfrak{b}$-torsion polynomials; we consider primes that split totally in the ray class

field. We can then uses the methods of Section 5.6.2 to compute the $\mathfrak{b}$-torsion points. In fact, as we have seen, it suffices to compute the $\mathfrak{b}$-point for one elliptic curve with CM, and then use isogenies both to compute the other CM curves as in Section 7.3 and to push the $\mathfrak{b}$-torsion point to get all of them. Considering primes that split totally in the ray class field, and not only in the Hilbert class field, is key for a quasi-linear algorithm.

When $g = 2$, the class polynomials have non integral coefficients, ie they live in $\mathcal{F}^r$ rather than in $O_{\mathcal{F}^r}$. Like for modular polynomials, this is due to the fact that the Igusa invariants we use are only defined away from $\chi_{10} = 0$. So the denominators correspond to primes p such that there is an A with CM by $(\mathcal{E}, \Phi)$ which reduces modulo p to a product of elliptic curves (this is in particular the case for supersingular reduction). So once we have evaluated our polynomials at some precision, we need to do a rational reconstruction step rather than just an integral reconstruction. But we do have a good control on this denominator, see [Str10, § II.9] and the refinements in [LV14].

For higher g, it would be interesting to have an integral version of the class polynomials, by interpreting the denominator as a certain modular value, as we did in Section 5.3.6 for the modular polynomials. It would be sufficient to define a canonical differential basis on our abelian varieties with CM, compatible with the Galois action. Clearly we should take a basis w_A on A such that the action of $O_{\mathcal{E}}$ is diagonal on w_A (given by the CM type Φ), but this is not quite enough to normalize w_A.

As we saw in Section 7.3 the height bound on the class polynomials $\widetilde{O}(\Delta_1^{3/2}\Delta_0^{5/2})$ for $g = 2$ is pessimistic compared to thee bound $\widetilde{O}(\Delta_1^{1/2}\Delta_0^{1/2})$ observed in practice. It is possible we could prove the improved height bounds (without relying on Colmez conjecture) using the refinements of [LV14] for the denominators and then refining the arguments in [Str10, § II.11] which bound the numerators. (Streng's height estimation is dominated by the estimated bound $\widetilde{O}(\Delta_1^{3/2}\Delta_0^{5/2})$ on denominators of [GL12], but even a refined bound of $\widetilde{O}(\Delta_1^{1/2}\Delta_0^{1/2})$ for the the height of the denominators would only give a height of $\widetilde{O}(\Delta_1^{1/2}\Delta_0^{3/2} + \Delta_0\Delta_1^{3/4})$ for the class polynomials using [Str10, § II.11].)

Class polynomials are just an exemple of Shimura varieties (of PEL type). I would have liked to call this Chapter "Computing integral models of Shimura varieties", but unfortunately I don't know of efficient ways to compute them (even in small dimension), apart of course from modular polynomials and class polynomials. The reason is that in dimension > 0, we cannot enumerate all points, and while evaluation-interpolation works, to get fast interpolation we require to sample points in the variety in a controlled way. Otherwise relations are found by linear algebra (either on Fourier coefficients as in [Gru10], or on some evaluations as we did in [MR20b] to compute some Humbert surfaces), but this is not quasi-optimal. This look like a challenging (hence fun!) problem.

Even on the moduli of abelian surfaces there are lots of interesting Shimura varieties to compute: Hilbert/Humbert surfaces, Shimura curves, generalised Humbert varieties…We refer to [Elk08; EK14; Gru10] for some explicit computations. Likewise, once we have equations for Hilbert surfaces (we do have them for small discriminants), we could try to compute the families of abelian variety with real multiplication. For instance we could use Chapter 5 to compute the endomorphism $\sqrt{d}$ on the generic point of the surface (or via evaluation/interpolation of the computation of $\sqrt{d}$ on geometric points, ie over some abelian surfaces in this family).

PERSONAL BIBLIOGRAPHY

[BCR10] G. Bisson, R. Cosset, and D. Robert. "AVIsogenies". Magma package devoted to the computation of isogenies between abelian varieties. 2010. URL: https://www.math.u-bordeaux.fr/~damienrobert/avisogenies/. Free software (LGPLv2+), registered to APP (reference IDDN.FR.001.440011.000.R.P.2010.000.10000). Latest version 0.6, released on 2012-11-28. (Cit. on pp. 3, 7, 12, 52, 119).

[BCR11] G. Bisson, R. Cosset, and D. Robert. "On the Practical Computation of Isogenies of Jacobian Surfaces". 2011. URL: https://www.math.u-bordeaux.fr/~damienrobert/avisogenies/. In preparation. (Cit. on pp. 3, 8, 12, 119).

[BLR11] G. Bisson, K. E. Lauter, and D. Robert. "Using horizontal isogenies to find hyperelliptic curves of cryptographic interest". 2011. In preparation. (Cit. on pp. 3, 12, 144).

[CR15] R. Cosset and D. Robert. "An algorithm for computing (ℓ,ℓ)-isogenies in polynomial time on Jacobians of hyperelliptic curves of genus 2". In: *Mathematics of Computation* 84.294 (Nov. 2015), pp. 1953–1975. DOI: 10.1090/S0025-5718-2014-02899-8. URL: http://www.normalesup.org/~robert/pro/publications/articles/niveau.pdf. HAL: hal-00578991, eprint: 2011/143. (Cit. on pp. 3, 7, 9–11, 18, 19, 32, 46, 72, 73, 80, 89).

[DJR+17] A. Dudeanu, D. Jetchev, D. Robert, and M. Vuille. "Cyclic Isogenies for Abelian Varieties with Real Multiplication". Oct. 2017. URL: http://www.normalesup.org/~robert/pro/publications/articles/cyclic.pdf. HAL: hal-01629829. (Cit. on pp. 3, 9, 11, 32, 74, 75).

[ER13] A. Enge and D. Robert. "Computing class polynomials in genus 2". Apr. 2013. URL: http://www.normalesup.org/~robert/pro/publications/reports/2013-04-class_poly_g2.pdf (cit. on pp. 141, 143).

[ERS16] A. Enge, D. Robert, and G. Seiler. "A CRT approach to computing ray class fields of imaginary quadratic fields". 2016. URL: http://www.normalesup.org/~robert/pro/publications/articles/rayclass.pdf. In preparation. (Cit. on pp. 9, 12, 145).

[FLR11] J.-C. Faugère, D. Lubicz, and D. Robert. "Computing modular correspondences for abelian varieties". In: *Journal of Algebra* 343.1 (Oct. 2011), pp. 248–277. DOI: 10.1016/j.jalgebra.2011.06.031. arXiv: 0910.4668 [cs.SC]. URL: http://www.normalesup.org/~robert/pro/publications/articles/modular.pdf. HAL: hal-00426338. (Cit. on pp. 3, 7, 10–12, 89, 90, 133, 134).

[IMR+14] S. Ionica, C. Martindale, D. Robert, and M. Streng. "Isogeny graphs of ordinary abelian surfaces over a finite field". Mar. 2014. In preparation. (Cit. on pp. 3, 8, 12, 121).

[KPR20] J. Kieffer, A. Page, and D. Robert. "Computing isogenies from modular equations between Jacobians of genus 2 curves". Oct. 2020. arXiv: 2001.04137 [math.AG]. URL: http://www.normalesup.org/~robert/pro/publications/articles/modular_isogenies_g2.pdf. HAL: hal-02436133. (Cit. on pp. 3, 9, 11, 77–79, 86, 98, 107–113, 118, 125, 126).

[KNR+20a] M. Kirschmer, F. Narbonne, C. Ritzenthaler, and D. Robert. "FromLatticesToModularForms". Computation of modular forms in the isogeny class spanned by products of elliptic curves. Apr. 2020. URL: https://gitlab.inria.fr/roberdam/fromlatticestomodularforms (cit. on p. 3).

[KNR+20b] M. Kirschmer, F. Narbonne, C. Ritzenthaler, and D. Robert. "Spanning the isogeny class of a power of an elliptic curve". Accepted for publication at Mathematics of Computation. Apr. 2020. arXiv: 2004.08315. URL: http://www.normalesup.org/~robert/pro/publications/articles/algebraic_obstruction.pdf. HAL: hal-02554714. (Cit. on pp. 3, 7, 8, 11, 12, 53, 76, 122, 124).

[LR13] K. E. Lauter and D. Robert. "Improved CRT Algorithm for Class Polynomials in Genus 2". In: *ANTS X — Proceedings of the Tenth Algorithmic Number Theory Symposium*. Ed. by E. W. Howe and K. S. Kedlaya. Vol. 1. The Open Book Series. Berkeley: Mathematical Sciences Publisher, Nov. 2013, pp. 437–461. DOI: 10.2140/obs.2013.1.437. URL: http://www.normalesup.org/~robert/pro/publications/articles/classCRT.pdf. Slides: 2012-07-ANTS-SanDiego.pdf (30min, International Algorithmic Number Theory Symposium (ANTS-X), July 2012, San Diego, USA), HAL: hal-00734450, eprint: 2012/443. (Cit. on pp. 3, 8, 10, 12, 141, 143–145).

[LR10] D. Lubicz and D. Robert. "Efficient pairing computation with theta functions". In: ed. by G. Hanrot, F. Morain, and E. Thomé. Vol. 6197. Lecture Notes in Comput. Sci. 9th International Symposium, Nancy, France, ANTS-IX, July 19-23, 2010, Proceedings. Springer–Verlag, July 2010. DOI: 10.1007/978-3-642-14518-6_21. URL: http://www.normalesup.org/~robert/pro/publications/articles/pairings.pdf. Slides: 2010-07-ANTS-Nancy.pdf (30min, International Algorithmic Number Theory Symposium (ANTS-IX), July 2010, Nancy), HAL: hal-00528944. (Cit. on pp. 3, 8, 10, 11, 18, 32, 56, 59).

[LR12] D. Lubicz and D. Robert. "Computing isogenies between abelian varieties". In: *Compositio Mathematica* 148.5 (Sept. 2012), pp. 1483–1515. DOI: 10.1112/S0010437X12000243. arXiv: 1001.2016 [math.AG]. URL: http://www.normalesup.org/~robert/pro/publications/articles/isogenies.pdf. HAL: hal-00446062. (Cit. on pp. 3, 7, 9–12, 18, 19, 32, 35, 42, 59, 71, 80, 89).

[LR15a] D. Lubicz and D. Robert. "A generalisation of Miller's algorithm and applications to pairing computations on abelian varieties". In: *Journal of Symbolic Computation* 67 (Mar. 2015), pp. 68–92. DOI: 10.1016/j.jsc.2014.08.001. URL: http://www.normalesup.org/~robert/pro/publications/articles/optimal.pdf. HAL: hal-00806923, eprint: 2013/192. (Cit. on pp. 3, 8, 10, 11, 18, 32–34, 51, 56–59).

[LR15b] D. Lubicz and D. Robert. "Computing separable isogenies in quasi-optimal time". In: *LMS Journal of Computation and Mathematics* 18 (1 Feb. 2015), pp. 198–216. DOI: 10.1112/S146115701400045X. arXiv: 1402.3628. URL: http://www.normalesup.org/~robert/pro/publications/articles/rational.pdf. HAL: hal-00954895. (Cit. on pp. 3, 9, 11, 69, 73, 80).

[LR16] D. Lubicz and D. Robert. "Arithmetic on Abelian and Kummer Varieties". In: *Finite Fields and Their Applications* 39 (May 2016), pp. 130–158. DOI: 10.1016/j.ffa.2016.01.009. URL: http://www.normalesup.org/~robert/pro/publications/articles/arithmetic.pdf. HAL: hal-01057467, eprint: 2014/493. (Cit. on pp. 3, 4, 9–11, 32, 34, 48–51).

[LR20a] D. Lubicz and D. Robert. "Faster isogenies via a fast change of level". Dec. 2020. In preparation. (Cit. on pp. 3, 9, 11).

[LR20b] D. Lubicz and D. Robert. "Linear representation of endomorphisms of Kummer varieties". Dec. 2020. URL: http://www.normalesup.org/~robert/pro/publications/articles/action.pdf. In preparation. (Cit. on pp. 3, 9, 11, 12, 48, 136, 137).

[MR20a] A. Maiga and D. Robert. "Computing the canonical lift of genus 2 curves in odd characteristic". Dec. 2020. URL: http://www.normalesup.org/~robert/pro/publications/articles/canonical_lift_g2.pdf. In preparation. (Cit. on pp. 3, 9, 11, 12, 132, 134, 135, 137).

[MR21] A. Maiga and D. Robert. "Computing the 2-adic canonical lift of genus 2 curves". Accepted for publication at Proceedings of the 7th International Conference on Mathematics and Computing (ICMC 2021). Jan. 2021. URL: http://www.normalesup.org/~robert/pro/publications/articles/canonical_lift_g2_p2.pdf. HAL: hal-03119147. (Cit. on pp. 3, 9, 11, 12, 86, 134, 137).

[MR19] E. Milio and D. Robert. "Denominators of modular polynomials on Hilbert surfaces". June 2019. In preparation. (Cit. on pp. 3, 8, 11, 91, 99).

[MR20b] E. Milio and D. Robert. "Modular polynomials on Hilbert surfaces". In: *Journal of Number Theory* 216 (Nov. 2020), pp. 403–459. DOI: 10.1016/j.jnt.2020.04.014. URL: https://www.sciencedirect.com/science/article/abs/pii/S0022314X20301402. HAL: hal-01520262, Reproducible archive: https://data.mendeley.com/datasets/yy3bty5ktk/1. (Cit. on pp. 8, 11, 12, 86, 91–93, 96, 98–100, 146).

[Rob10] D. Robert. "Theta functions and cryptographic applications". PhD thesis. Université Henri-Poincarré, Nancy 1, France, July 2010. URL: http://www.normalesup.org/~robert/pro/publications/academic/phd.pdf. Slides: 2010-07-Phd-Nancy.pdf (1h, Nancy), TEL: tel-00528942. (Cit. on pp. 12, 18–20, 22–24, 26–28, 30, 31, 33–37, 42, 43, 50, 51, 59, 60, 87, 90).

[Rob13] D. Robert. "Computing cyclic isogenies using real multiplication". (Notes). ANR Peace meeting, Paris. Apr. 2013. URL: http://www.normalesup.org/~robert/pro/publications/notes/2013-04-Peace-Paris-Cyclic-Isogenies.pdf (cit. on pp. 9, 74, 75).

[Rob15] D. Robert. "Isogenies, Polarisations and Real Multiplication". Modular Forms and Curves of Low Genus: Computational Aspects, ICERM, Providence, USA. Sept. 2015. URL: http://www.normalesup.org/~robert/pro/publications/slides/2015-09-Providence-ICERM.pdf (cit. on p. 122).

[Rob17] D. Robert. *Guide to Pairing-Based Cryptography*. 2017. URL: https://www.worldcat.org/title/guide-to-pairing-based-cryptography/oclc/971264380. Chapter 3 on « Pairings » with Sorina Ionica, and Chapter 10 on « Choosing Parameters » with Sylvain Duquesne, Nadia El Mrabet, Safia Haloui and Franck Rondepierre (cit. on pp. 9, 11, 55, 57, 61, 62).

[Rob21] D. Robert. *General theory of abelian varieties and their moduli spaces*. Jan. 2021. URL: http://www.normalesup.org/~robert/pro/publications/books/avtheory.pdf. Draft version. (Cit. on pp. 6, 7, 11, 18–20, 32, 36, 38, 55, 56, 59, 62, 64, 65, 85, 86, 91, 120, 126, 127, 141–143, 145).

[SR11] A. Shamir and D. Robert. "Comparative power attacks on Edwards curves". 2011. In preparation. (Cit. on p. 3).

[Kie20a] J. Kieffer. "Degree and height estimates for modular equations on PEL Shimura varieties". 2020. arXiv: 2001.04138 [math.AG]. HAL: hal-02436057. (Cit. on pp. 9, 11, 91–93, 96–100).

[Kie20b] J. Kieffer. "Evaluating modular polynomials in genus 2". 2020. arXiv: 2010.10094 [math.NT]. HAL: hal-02971326. (Cit. on pp. 9, 11, 12, 94, 95, 97, 98, 101–103, 124).

[Kie20c] J. Kieffer. "Sign choices in the AGM for genus two theta constants". 2020. arXiv: 2010.07579 [math.NT]. HAL: hal-02967220. (Cit. on pp. 9, 11, 94, 95, 124, 125).

[KPR20] J. Kieffer, A. Page, and D. Robert. "Computing isogenies from modular equations between Jacobians of genus 2 curves". Oct. 2020. arXiv: 2001.04137 [math.AG]. URL: http://www.normalesup.org/~robert/pro/publications/articles/modular_isogenies_g2.pdf. HAL: hal-02436133. (Cit. on pp. 3, 9, 11, 77–79, 86, 98, 107–113, 118, 125, 126).

[MR20a] A. Maiga and D. Robert. "Computing the canonical lift of genus 2 curves in odd characteristic". Dec. 2020. URL: http://www.normalesup.org/~robert/pro/publications/articles/canonical_lift_g2.pdf. In preparation. (Cit. on pp. 3, 9, 11, 12, 132, 134, 135, 137).

[MR21] A. Maiga and D. Robert. "Computing the 2-adic canonical lift of genus 2 curves". Accepted for publication at Proceedings of the 7th International Conference on Mathematics and Computing (ICMC 2021). Jan. 2021. URL: http://www.normalesup.org/~robert/pro/publications/articles/canonical_lift_g2_p2.pdf. HAL: hal-03119147. (Cit. on pp. 3, 9, 11, 12, 86, 134, 137).

[Mil15a] E. Milio. "A quasi-linear time algorithm for computing modular polynomials in dimension 2". In: *LMS Journal of Computation and Mathematics* 18.1 (2015), pp. 603–632. arXiv: 1411.0409 (cit. on pp. 8, 11, 12, 86, 92, 100).

[Mil15b] E. Milio. "Calcul de polynômes modulaires en dimension 2". Thèse de doctorat dirigée par Enge, Andreas et Robert, Damien; Mathematiques pures, Bordeaux 2015. PhD thesis. 2015. URL: http://www.theses.fr/2015BORD0285 (cit. on pp. 92, 97, 100).

[MR19] E. Milio and D. Robert. "Denominators of modular polynomials on Hilbert surfaces". June 2019. In preparation. (Cit. on pp. 3, 8, 11, 91, 99).

[MR20b] E. Milio and D. Robert. "Modular polynomials on Hilbert surfaces". In: *Journal of Number Theory* 216 (Nov. 2020), pp. 403–459. DOI: 10.1016/j.jnt.2020.04.014. URL: https://www.sciencedirect.com/science/article/abs/pii/S0022314X20301402. HAL: hal-01520262, Reproducible archive: https://data.mendeley.com/datasets/yy3bty5ktk/1. (Cit. on pp. 8, 11, 12, 86, 91–93, 96, 98–100, 146).

BIBLIOGRAPHY

[Abe20] S. Abelard. "Counting points on hyperelliptic curves with explicit real multiplication in arbitrary genus". In: *Journal of Complexity* 57 (2020), p. 101440 (cit. on pp. 12, 79, 115–117).

[ACL20] S. Abelard, A. Couvreur, and G. Lecerf. "Sub-quadratic time for riemann-roch spaces: case of smooth divisors over nodal plane projective curves". In: *Proceedings of the 45th International Symposium on Symbolic and Algebraic Computation*. 2020, pp. 14–21 (cit. on p. 18).

[ACL21] S. Abelard, A. Couvreur, and G. Lecerf. "Efficient computation of Riemann-Roch spaces for plane curves with ordinary singularities". 2021. URL: https://hal.archives-ouvertes.fr/hal-03110135/document (cit. on p. 18).

[AGS19a] S. Abelard, P. Gaudry, and P.-J. Spaenlehauer. "Counting points on genus-3 hyperelliptic curves with explicit real multiplication". In: *The Open Book Series* 2.1 (2019), pp. 1–19 (cit. on p. 118).

[AGS19b] S. Abelard, P. Gaudry, and P.-J. Spaenlehauer. "Improved complexity bounds for counting points on hyperelliptic curves". In: *Foundations of Computational Mathematics* 19.3 (2019), pp. 591–621 (cit. on pp. 12, 79, 116, 134).

[AH01] L. M. Adleman and M.-D. Huang. "Counting points on curves and abelian varieties over finite fields". In: *Journal of Symbolic Computation* 32.3 (2001), pp. 171–189 (cit. on pp. 116, 134).

[AL86] L. M. Adleman and H. W. Lenstra. "Finding irreducible polynomials over finite fields". In: *Proceedings of the eighteenth annual ACM symposium on Theory of computing*. 1986, pp. 350–355 (cit. on p. 103).

[AW21] J. Alman and V. V. Williams. "A refined laser method and faster matrix multiplication". In: *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM. 2021, pp. 522–539 (cit. on p. 113).

[AGH+18] F. Andreatta, E. Z. Goren, B. Howard, and K. M. Pera. "Faltings heights of abelian varieties with complex multiplication". In: *Annals of Mathematics* 187.2 (2018), pp. 391–531 (cit. on p. 144).

[Bac90] E. Bach. "Explicit bounds for primality testing and related problems". In: *Math. Comp.* 55.191 (1990), pp. 355–380. ISSN: 0025-5718. DOI: 10.2307/2008811 (cit. on p. 144).

[BGG+17] S. Ballentine, A. Guillevic, E. L. García, C. Martindale, M. Massierer, B. Smith, and J. Top. "Isogenies for point counting on genus two hyperelliptic curves with maximal real multiplication". In: *Algebraic geometry for coding theory and cryptography*. Springer, 2017, pp. 63–94 (cit. on p. 120).

[BS19] R. Barbulescu and S. Shinde. "A classification of ECM-friendly families using modular curves". 2019. URL: https://hal.inria.fr/hal-01822144/ (cit. on p. 6).

[Bel04] K. Belabas. "A relative van Hoeij algorithm over number fields". In: *Journal of Symbolic Computation* 37.5 (2004), pp. 641–668 (cit. on p. 106).

[BDL+20] D. Bernstein, L. De Feo, A. Leroux, and B. Smith. "Faster computation of isogenies of large prime degree". 2020. arXiv: 2003.10118 (cit. on pp. 81, 139).

[BS07] D. Bernstein and J. Sorenson. "Modular exponentiation via the explicit Chinese remainder theorem". In: *Mathematics of Computation* 76.257 (2007), pp. 443–454 (cit. on p. 104).

[BL04] C. Birkenhake and H. Lange. *Complex abelian varieties*. Second. Vol. 302. Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]. Berlin: Springer-Verlag, 2004, pp. xii+635. ISBN: 3-540-20488-1 (cit. on pp. 19, 20, 28).

[Bis11] G. Bisson. "Endomorphism Rings in Cryptography". PhD thesis. 2011 (cit. on pp. 5, 143).

[BS09] G. Bisson and A. Sutherland. "Computing the endomorphism ring of an ordinary elliptic curve over a finite field". In: *Journal of Number Theory* (2009) (cit. on pp. 5, 143).

[BBB+18] D. Boneh, J. Bonneau, B. Bünz, and B. Fisch. "Verifiable delay functions". In: *Annual International Cryptology Conference*. Springer. 2018, pp. 757–788 (cit. on p. 2).

[BMS+08] A. Bostan, F. Morain, B. Salvy, and E. Schost. "Fast algorithms for computing isogenies between elliptic curves". In: *Mathematics of Computation* 77.263 (2008), pp. 1755–1778 (cit. on pp. 5, 76, 77).

[BDD+19] L. Brieulle, L. De Feo, J. Doliskani, J.-P. Flori, and É. Schost. "Computing isomorphisms and embeddings of finite fields". In: *Mathematics of Computation* 88.317 (2019), pp. 1391–1426 (cit. on pp. 6, 103).

[BGL11] R. Bröker, D. Gruenewald, and K. Lauter. "Explicit CM theory for level 2-structures on abelian surfaces". In: *Algebra & Number Theory* 5.4 (2011), pp. 495–528. arXiv: 0910.1848 (cit. on p. 144).

[BL09] R. Bröker and K. Lauter. "Modular polynomials for genus 2". In: *LMS J. Comput. Math.* 12 (2009), pp. 326–339. ISSN: 1461-1570. arXiv: 0804.1565 (cit. on p. 91).

[BLS12] R. Bröker, K. Lauter, and A. Sutherland. "Modular polynomials via isogeny volcanoes". In: *Mathematics of Computation* 81.278 (2012), pp. 1201–1231. arXiv: 1001.0402 (cit. on p. 5).

[BJW17] E. H. Brooks, D. Jetchev, and B. Wesolowski. "Isogeny graphs of ordinary abelian varieties". In: *Research in Number Theory* 3.1 (2017), p. 28 (cit. on pp. 8, 121).

[Can16] L. Candelori. "The transformation laws of algebraic theta functions". 2016. arXiv: 1609.04486 (cit. on pp. 41, 76).

[Can20] L. Candelori. "The algebraic functional equation of Riemann's theta function". In: *Annales de l'Institut Fourier*. Vol. 70. 2. 2020, pp. 809–830. arXiv: 1512.04415 (cit. on pp. 28, 41).

[Can94] D. G. Cantor. "On the analogue of the division polynomials for hyperelliptic curves." In: *Journal für die reine und angewandte Mathematik* 1994.447 (1994), pp. 91–146 (cit. on p. 79).

[Car03] R. Carls. "Generalized AGM sequences and approximation of canonical lifts". PhD thesis. Apr. 2003. URL: http://www.math.leidenuniv.nl/carls (cit. on p. 133).

[Car07] R. Carls. "Canonical coordinates on the canonical lift". In: *J. Ramanujan Math. Soc.* 22.1 (2007), pp. 1–14 (cit. on p. 133).

[CKL08] R. Carls, D. Kohel, and D. Lubicz. "Higher-dimensional 3-adic CM construction". In: *J. Algebra* 319.3 (2008), pp. 971–1006. ISSN: 0021-8693. DOI: 10.1016/j.jalgebra.2007.11.016 (cit. on pp. 133, 145).

[CL08a] R. Carls and D. Lubicz. "A p-adic quasi-quadratic time and quadratic space point counting algorithm". In: *International Mathematics Research Notices* (2008) (cit. on pp. 7, 133–136, 139).

[CEL20] X. Caruso, E. Eid, and R. Lercier. "Fast computation of elliptic curve isogenies in characteristic two". 2020. arXiv: 2003.06367 (cit. on pp. 78, 108).

[CRV16] X. Caruso, D. Roe, and T. Vaccon. "Division and slope factorization of p-adic polynomials". In: *Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation*. 2016, pp. 159–166 (cit. on p. 131).

[CRV18] X. Caruso, D. Roe, and T. Vaccon. "ZpL: a p-adic precision package". In: *Proceedings of the 2018 ACM International Symposium on Symbolic and Algebraic Computation*. 2018, pp. 119–126 (cit. on p. 137).

[CF+96] J. W. S. Cassels, E. V. Flynn, et al. *Prolegomena to a middlebrow arithmetic of curves of genus 2*. Vol. 230. Cambridge University Press, 1996 (cit. on pp. 7, 52, 80).

[CLM+18] W. Castryck, T. Lange, C. Martindale, L. Panny, and J. Renes. "CSIDH: an efficient post-quantum commutative group action". In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2018, pp. 395–427 (cit. on pp. 2, 6).

[Cel19] T. O. Celik. "A Thomae-like Formula: Algebraic Computations of Theta Constants". 2019. arXiv: 1901.08459 (cit. on p. 53).

[CN90] C.-L. Chai and P. Norman. "Bad reduction of the Siegel moduli scheme of genus two with Γo (p)-level structure". In: *American Journal of Mathematics* (1990), pp. 1003–1071 (cit. on pp. 86, 106, 132).

[CLG09] D. Charles, K. Lauter, and E. Goren. "Cryptographic hash functions from expander graphs". In: *Journal of Cryptology* 22.1 (2009), pp. 93–113. ISSN: 0933-2790 (cit. on pp. 2, 6).

[Cho06] T. Y. Chow. "You could have invented spectral sequences". In: *Notices of the AMS* 53 (2006), pp. 15–19 (cit. on p. 58).

[Cv00] C. Ciliberto and G. van der Geer. "The moduli space of abelian varieties and the singularities of the theta divisor". In: *Surveys in differential geometry*. Vol. 7. Surv. Differ. Geom. Int. Press, Somerville, MA, 2000, pp. 61–81 (cit. on p. 126).

[CFG18] F. Cléry, C. Faber, and G. van der Geer. "Covariants of binary sextics and modular forms of degree 2 with character". 2018. arXiv: 1803.05624 [math.AG] (cit. on p. 112).

[CFG20] F. Cléry, C. Faber, and G. van der Geer. "Concomitants of Ternary Quartics and Vector-valued Siegel and Teichmüller Modular Forms of Genus Three". 2020. arXiv: 1908.04248 [math.AG] (cit. on p. 126).

[CFv17] F. Cléry, C. Faber, and G. van der Geer. "Covariants of binary sextics and vector-valued Siegel modular forms of genus two". In: *Math. Ann.* 369.3-4 (2017), pp. 1649–1669 (cit. on pp. 111, 112).

[Col93] P. Colmez. "Périodes des variétés abéliennes à multiplication complexe". In: *Annals of Mathematics* (1993), pp. 625–683 (cit. on p. 144).

[Cos11] R. Cosset. "Application des fonctions thêta à la cryptographie sur courbes hyperelliptiques". PhD thesis. 2011 (cit. on p. 53).

[CHK19] E. Costa, D. Harvey, and K. Kedlaya. "Zeta functions of nondegenerate hypersurfaces in toric varieties via controlled reduction in p-adic cohomology". In: *The Open Book Series* 2.1 (2019), pp. 221–238 (cit. on p. 139).

[CMS+17] E. Costa, N. Mascot, J. Sijsling, and J. Voight. "Rigorous computation of the endomorphism ring of a Jacobian". 2017. arXiv: 1705.09248 (cit. on p. 77).

[CJL+17] C. Costello, D. Jao, P. Longa, M. Naehrig, J. Renes, and D. Urbanik. "Efficient compression of SIDH public keys". In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2017, pp. 679–706 (cit. on p. 64).

[CLN16] C. Costello, P. Longa, and M. Naehrig. "Efficient algorithms for supersingular isogeny Diffie-Hellman". In: *Advances in Cryptology*. Springer. 2016. URL: https://ecc2017.cs.ru.nl/slides/ecc2017-costello.pdf (cit. on pp. 2, 6, 10).

[CS20] C. Costello and B. Smith. "The supersingular isogeny problem in genus 2 and beyond". In: *International Conference on Post-Quantum Cryptography*. Springer. 2020, pp. 151–168 (cit. on p. 3).

[Cou94] J. Couveignes. "Quelques calculs en théorie des nombres". PhD thesis. 1994 (cit. on p. 108).

[Cou96] J. Couveignes. "Computing l-isogenies using the p-torsion". In: *Algorithmic Number Theory* (1996), pp. 59–65 (cit. on p. 108).

[CL08b] J. Couveignes and R. Lercier. "Galois invariant smoothness basis". In: *Algebraic geometry and its applications* (2008) (cit. on p. 6).

[CL09] J. Couveignes and R. Lercier. "Elliptic periods for finite fields". In: *Finite fields and their applications* 15.1 (2009), pp. 1–22 (cit. on p. 6).

[Cou06] J. M. Couveignes. "Hard Homogeneous Spaces." In: *IACR Cryptology ePrint Archive* 2006 (2006), p. 291 (cit. on pp. 2, 6).

[Cou20] J.-M. Couveignes. "Enumerating number fields". In: *Annals of Mathematics* 192.2 (2020), pp. 487–497 (cit. on p. 102).

[CE14] J.-M. Couveignes and T. Ezome. "Computing functions on Jacobians and their quotients". In: *LMS Journal of Computation and Mathematics* 18.1 (2014), pp. 555–577. arXiv: 1409.0481 (cit. on pp. 11, 12, 19, 40, 52, 53, 70, 75, 77, 80).

[CK12] J.-M. Couveignes and J.-G. Kammerer. "The geometry of flex tangents to a cubic curve and its parameterizations". In: *Journal of Symbolic Computation* 47.3 (2012), pp. 266–281 (cit. on p. 4).

[CL13] J.-M. Couveignes and R. Lercier. "Fast construction of irreducible polynomials over finite fields". In: *Israel Journal of Mathematics* 194.1 (2013), pp. 77–105 (cit. on p. 6).

[De 17] L. De Feo. *Mathematics of Isogeny Based Cryptography*. 2017. arXiv: 1711.04062 (cit. on p. 2).

[DHP+16] L. De Feo, C. Hugounenq, J. Plût, and É. Schost. "Explicit isogenies in quadratic time in any characteristic". In: *LMS Journal of Computation and Mathematics* 19.A (2016), pp. 267–282 (cit. on pp. 12, 108).

[DJP14] L. De Feo, D. Jao, and J. Plût. "Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies". In: *Journal of Mathematical Cryptology* 8.3 (2014), pp. 209–247 (cit. on pp. 2, 6, 80).

[DKS18] L. De Feo, J. Kieffer, and B. Smith. "Towards practical key exchange from ordinary isogeny graphs". In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2018, pp. 365–394. arXiv: 1809.07543 (cit. on pp. 2, 6).

[DKL+20] L. De Feo, D. Kohel, A. Leroux, C. Petit, and B. Wesolowski. "SQISign: compact post-quantum signatures from quaternions and isogenies". In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2020, pp. 64–93 (cit. on p. 2).

[DMP+19] L. De Feo, S. Masson, C. Petit, and A. Sanso. "Verifiable Delay Functions from Supersingular Isogenies and Pairings". 2019. eprint: CryptologyePrintArchive, Report2019/166. URL: https://eprint.iacr.org/2019/166.pdf (cit. on pp. 2, 6).

[DR73] P. Deligne and M. Rapoport. "Les schémas de modules de courbes elliptiques". In: *Modular functions of one variable, II (Proc. Internat. Summer School, Univ. Antwerp, Antwerp, 1972)*. 1973, 143–316. Lecture Notes in Math., Vol. 349 (cit. on p. 118).

[DM69] P. Deligne and D. Mumford. "The irreducibility of the space of curves of given genus". In: *Publications Mathématiques de l'Institut des Hautes Études Scientifiques* 36.1 (1969), pp. 75–109 (cit. on p. 122).

[DA70] M. Demazure and M. Artin. *Schémas en groupes (SGA3)*. Springer Berlin, Heidelberg, New York, 1970 (cit. on p. 127).

[DT08] C. Diem and E. Thomé. "Index calculus in class groups of non-hyperelliptic curves of genus three". In: *Journal of Cryptology* 21.4 (2008), pp. 593–611 (cit. on p. 3).

[DH76] W. Diffie and M. Hellman. "New directions in cryptography". In: *IEEE Transactions on information Theory* 22.6 (1976), pp. 644–654 (cit. on p. 1).

[DIK06] C. Doche, T. Icart, and D. Kohel. "Efficient scalar multiplication by isogeny decompositions". In: *Public Key Cryptography-PKC 2006* (2006), pp. 191–206 (cit. on p. 6).

[Dup06] R. Dupont. "Moyenne arithmetico-geometrique, suites de Borchardt et applications". In: *These de doctorat, Ecole polytechnique, Palaiseau* (2006) (cit. on pp. 6, 92–95, 100, 103, 124, 125).

[Eid20] É. Eid. "Fast computation of hyperelliptic curve isogenies in odd characteristic". 2020. arXiv: 2009.12180 [math.AG] (cit. on p. 78).

[Elk92] N. Elkies. "Explicit isogenies". In: *manuscript, Boston MA* (1992) (cit. on p. 5).

[Elk97] N. Elkies. "Elliptic and modular curves over finite fields and related computational issues". In: *Computational perspectives on number theory: proceedings of a conference in honor of AOL Atkin, September 1995, University of Illinois at Chicago*. Vol. 7. Amer Mathematical Society. 1997, p. 21 (cit. on p. 5).

[EK14] N. Elkies and A. Kumar. "K3 surfaces and equations for Hilbert modular surfaces". In: *Algebra & Number Theory* 8.10 (2014), pp. 2297–2411 (cit. on p. 146).

[Elk08] N. D. Elkies. "Shimura curve computations via K3 surfaces of Néron–Severi rank at least 19". In: *International Algorithmic Number Theory Symposium*. Springer. 2008, pp. 196–211 (cit. on p. 146).

[Eng09a] A. Enge. "Computing modular polynomials in quasi-linear time". In: *Math. Comp* 78.267 (2009), pp. 1809–1824 (cit. on pp. 5, 92).

[Eng09b] A. Enge. "The complexity of class polynomial computation via floating point approximations". In: *Mathematics of Computation* 78.266 (2009), pp. 1089–1107 (cit. on p. 142).

[EGT09] A. Enge, P. Gaudry, and E. Thomé. "An $L(1/3)$ Discrete Logarithm Algorithm for Low Degree Curves". In: *Imprint* (2009) (cit. on p. 3).

[ES10] A. Enge and A. Sutherland. "Class invariants by the CRT method, ANTS IX: Proceedings of the Algorithmic Number Theory 9th International Symposium". In: *Lecture Notes in Computer Science* 6197 (July 2010), pp. 142–156 (cit. on pp. 5, 144).

[ET14] A. Enge and E. Thomé. "Computing class polynomials for abelian surfaces". In: *Experimental Mathematics* 23.2 (2014), pp. 129–145 (cit. on p. 142).

[FC90] G. Faltings and C.-L. Chai. *Degeneration of abelian varieties*. Ergebnisse der Mathematik und ihrer Grenzgebiete (3) 22. Springer-Verlag, Berlin, 1990 (cit. on pp. 41, 86).

[Feo10] L. de Feo. "Algorithmes Rapides pour les Tours de Corps Finis et les Isogénies". PhD thesis. Ecole Polytechnique X, Dec. 2010. URL: http://hal.inria.fr/tel-00547034/en (cit. on p. 108).

[Fly90] E. Flynn. "The Jacobian and formal group of a curve of genus 2 over an arbitrary ground field". In: *Mathematical Proceedings of the Cambridge Philosophical Society*. Vol. 107. 03. Cambridge Univ Press. 1990, pp. 425–441 (cit. on pp. 52, 80).

[FGH00] M. Fouquet, P. Gaudry, and R. Harley. "An extension of Satoh's algorithm and its implementation". In: (2000) (cit. on p. 131).

[FM02] M. Fouquet and F. Morain. "Isogeny volcanoes and the SEA algorithm". In: *Algorithmic number theory (Sydney, 2002)*. Vol. 2369. Lecture Notes in Comput. Sci. Berlin: Springer, 2002, pp. 276–291. DOI: 10.1007/3-540-45455-1_23 (cit. on p. 5).

[FL08] D. Freeman and K. Lauter. "Computing endomorphism rings of Jacobians of genus 2 curves over finite fields". In: *Algebraic geometry and its applications* (2008), pp. 29–66 (cit. on p. 143).

[GHS02] S. Galbraith, F. Hess, and N. Smart. "Extending the GHS Weil descent attack". In: *Advances in Cryptology—EUROCRYPT 2002*. Springer. 2002, pp. 29–44 (cit. on p. 6).

[Gau04] P. Gaudry. "Algorithmes de comptage de points d'une courbe définie sur un corps fini". 2004. URL: http://www.loria.fr/~gaudry/publis/pano.pdf (cit. on p. 129).

[Gau07] P. Gaudry. "Fast genus 2 arithmetic based on Theta functions". In: *Journal of Mathematical Cryptology* 1.3 (2007), pp. 243–265 (cit. on p. 6).

[Gau09] P. Gaudry. "Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem". In: *Journal of Symbolic Computation* 44.12 (2009), pp. 1690–1702 (cit. on p. 3).

[GG03] P. Gaudry and N. Gurel. "Counting points in medium characteristic using Kedlaya's algorithm". In: *Experimental Mathematics* 12.4 (2003), pp. 395–402 (cit. on p. 139).

[GHK+06] P. Gaudry, T. Houtmann, D. Kohel, C. Ritzenthaler, and A. Weng. "The 2-adic CM method for genus 2 curves with application to cryptography". In: *Advances in cryptology—ASIACRYPT 2006*. Vol. 4284. Lecture Notes in Comput. Sci. Berlin: Springer, 2006, pp. 114–129. DOI: 10.1007/11935230_8 (cit. on pp. 91, 133, 145).

[GL09] P. Gaudry and D. Lubicz. "The arithmetic of characteristic 2 Kummer surfaces and of elliptic Kummer lines". In: *Finite Fields and Their Applications* 15.2 (2009), pp. 246–260 (cit. on p. 3).

[GKS11] P. Gaudry, D. R. Kohel, and B. A. Smith. "Counting Points on Genus 2 Curves with Real Multiplication". In: *ASIACRYPT*. Ed. by D. H. Lee and X. Wang. Vol. 7073. Lecture Notes in Computer Science. Springer, 2011, pp. 504–519. ISBN: 978-3-642-25384-3 (cit. on pp. 12, 115).

[GS12] P. Gaudry and É. Schost. "Genus 2 point counting over prime fields". In: *Journal of Symbolic Computation* 47.4 (2012), pp. 368–400 (cit. on pp. 12, 100, 113).

[GL12] E. Z. Goren and K. E. Lauter. "Genus 2 curves with complex multiplication". In: *International Mathematics Research Notices* 2012.5 (2012), pp. 1068–1142 (cit. on pp. 86, 144, 146).

[GD64] A. Grothendieck and J. Dieudonné. "Eléments de géométrie algébrique". In: *Publ. math. IHES* 20.24 (1964), p. 1965 (cit. on p. 128).

[Gro62] A. Grothendieck. *Fondements de la géométrie algébrique: extraits du Séminaire Bourbaki, 1957-1962*. Secrétariat mathématique, 1962 (cit. on p. 127).

[Gru10] D. Gruenewald. "Computing Humbert surfaces and applications". In: *Arithmetic, Geometry, Cryptography and Codint Theory 2009* (2010), pp. 59–69 (cit. on pp. 91, 146).

[Gun63] K.-B. Gundlach. "Die Bestimmung der Funktionen zur Hilbertschen Modulgruppe des Zahlkörpers $\mathbb{Q}(\sqrt{5})$". In: *Math. Annalen* 152 (1963), pp. 226–256 (cit. on p. 86).

[Gun65] K.-B. Gundlach. "Die Bestimmung der Funktionen zu einigen Hilbertschen Modulgruppen". In: *Journal für die reine und angewandte Mathematik* 220 (1965) (cit. on p. 86).

[HPS] G. Hanrot, A. Pellet-Mary, and D. Stehlé. "Reconstructing an element of a number field from any of its canonical embeddings". Private communication (cit. on p. 106).

[Har02] R. Harley. *Asymptotically optimal p-adic point-counting*. Email at the Number Theory List. 2002 (cit. on p. 129).

[HHL09] A. W. Harrow, A. Hassidim, and S. Lloyd. "Quantum algorithm for linear systems of equations". In: *Physical review letters* 103.15 (2009), p. 150502 (cit. on p. 10).

[Har07] D. Harvey. "Kedlaya's algorithm in larger characteristic". In: *Int. Math. Res. Notices* (2007) (cit. on p. 139).

[HSV06] F. Hess, N. Smart, and F. Vercauteren. "The Eta pairing revisited". In: *IEEE Transactions on Information Theory* 52.10 (2006), pp. 4595–4602 (cit. on p. 57).

[Hes02] F. Hess. "Computing Riemann–Roch spaces in algebraic function fields and related topics". In: *Journal of Symbolic Computation* 33.4 (2002), pp. 425–445 (cit. on pp. 9, 18).

[How95] E. Howe. "Principally polarized ordinary abelian varieties over finite fields". In: *American Mathematical Society* 347.7 (1995) (cit. on p. 122).

[Hua18] M.-D. A. Huang. "Trilinear maps for cryptography". 2018. arXiv: 1803.10325 [cs.CR] (cit. on pp. 13, 64, 75).

[Hua19] M.-D. A. Huang. "Trilinear maps for cryptography II". 2019. arXiv: 1810.03646 [cs.CR] (cit. on pp. 13, 64, 75).

[Igu60] J.-i. Igusa. "Arithmetic variety of moduli for genus two". In: *Annals of Mathematics* (1960), pp. 612–649 (cit. on pp. 86, 112, 137).

[Igu64] J.-i. Igusa. "On the graded ring of theta-constants". In: *American Journal of Mathematics* 86.1 (1964), pp. 219–246 (cit. on p. 95).

[Igu66] J.-i. Igusa. "On the graded ring of theta-constants (II)". In: *American Journal of Mathematics* 88.1 (1966), pp. 221–236 (cit. on p. 95).

[IT14] S. Ionica and E. Thomé. "Isogeny graphs with maximal real multiplication." In: *IACR Cryptology ePrint Archive* 2014 (2014), p. 230 (cit. on pp. 115, 121).

[JD11] D. Jao and L. De Feo. "Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies". In: *International Workshop on Post-Quantum Cryptography*. Springer. 2011, pp. 19–34 (cit. on pp. 2, 6, 80).

[JW15] D. Jetchev and B. Wesolowski. "On graphs of isogenies of principally polarizable abelian surfaces and the discrete logarithm problem". 2015. arXiv: 1506.00522 (cit. on p. 6).

[Jon93] A. J. de Jong. "The moduli spaces of polarized abelian varieties". In: *Mathematische Annalen* 295.1 (1993), pp. 485–503 (cit. on p. 32).

[JKB+18] B. W. Jordan, A. G. Keeton, B. Poonen, E. M. Rains, N. Shepherd-Barron, and J. T. Tate. "Abelian varieties isogenous to a power of an elliptic curve". In: *Compos. Math.* 154.5 (2018), pp. 934–959 (cit. on p. 124).

[Kan18] E. Kani. "Elliptic subcovers of a curve of Genus 2 II. The refined Humbert invariant". In: *Journal of Number Theory* 193 (2018), pp. 302–335 (cit. on p. 99).

[Kan19a] E. Kani. "Elliptic subcovers of a curve of genus 2. I. The isogeny defect". In: *Annales mathématiques du Québec* 43.2 (2019), pp. 281–303 (cit. on pp. 79, 99).

[Kan19b] E. Kani. *Generalized Humbert Schemes and Intersections of Humbert Surfaces*. 2019. URL: https://mast.queensu.ca/~kani/papers/interHum11.pdf (cit. on p. 99).

[KLL88] R. Kannan, A. K. Lenstra, and L. Lovász. "Polynomial factorization and nonrandomness of bits of algebraic and some transcendental numbers". In: *Mathematics of Computation* 50.181 (1988), pp. 235–250 (cit. on p. 106).

[Kat81] N. Katz. "Serre-Tate local moduli". In: *Surfaces algébriques*. Springer, 1981, pp. 138–202 (cit. on pp. 106, 111, 128, 132).

[Ked01] K. Kedlaya. "Counting points on hyperelliptic curves using Monsky-Washnitzer cohomology". 2001. arXiv: math/0105031 (cit. on pp. 5, 139).

[Ked16] K. S. Kedlaya. *p-adic cohomology*. 2016. arXiv: math/0601507 (cit. on p. 139).

[KU11] K. S. Kedlaya and C. Umans. "Fast polynomial factorization and modular composition". In: *SIAM Journal on Computing* 40.6 (2011), pp. 1767–1802 (cit. on pp. 80, 103, 106, 139).

[KM97] S. Keel and S. Mori. "Quotients by groupoids". In: *Annals of mathematics* 145.1 (1997), pp. 193–213 (cit. on p. 32).

[Kem88] G. Kempf. "Multiplication over abelian varieties". In: *American Journal of Mathematics* 110.4 (1988), pp. 765–773 (cit. on pp. 21, 33, 49).

[Kem89a] G. Kempf. "Linear systems on abelian varieties". In: *American Journal of Mathematics* 111.1 (1989), pp. 65–94 (cit. on pp. 21, 23, 29, 31–33, 88).

[Kem92] G. Kempf. "Equations of Kümmer Varieties". In: *American Journal of Mathematics* 114.1 (1992), pp. 229–232 (cit. on pp. 21, 54).

[Kem89b] G. Kempf. "Projective coordinate rings of abelian varieties". In: *Algebraic analysis, geometry and number theory* (1989), pp. 225–236 (cit. on pp. 21, 32).

[Kem90] G. R. Kempf. "Some wonderful rings in algebraic geometry". In: *Journal of Algebra* 134.1 (1990), pp. 222–224 (cit. on p. 21).

[Kem91] G. R. Kempf. *Complex abelian varieties and theta functions*. Springer Science & Business Media, 1991 (cit. on pp. 36, 76).

[Khu04] K. Khuri-Makdisi. "Linear algebra algorithms for divisors on an algebraic curve". In: *Mathematics of Computation* 73.245 (2004), pp. 333–357 (cit. on pp. 9, 18).

[Khu07] K. Khuri-Makdisi. "Asymptotically fast group operations on Jacobians of general curves". In: *Mathematics of Computation* 76.260 (2007), pp. 2213–2239 (cit. on pp. 9, 18).

[Koh96] D. Kohel. "Endomorphism rings of elliptic curves over finite fields". PhD thesis. University of California, 1996 (cit. on pp. 5, 77).

[Koh08] D. R. Kohel. "Complex multiplication and canonical lifts". In: *Algebraic Geometry And Its Applications: Dedicated to Gilles Lachaud on His 60th Birthday*. World Scientific, 2008, pp. 67–83 (cit. on pp. 132, 133, 140).

[Koi76] S. Koizumi. "Theta relations and projective normality of abelian varieties". In: *American Journal of Mathematics* (1976), pp. 865–889 (cit. on pp. 33, 72, 88).

[Kou00] A. Kouvidakis. "Theta line bundles and the determinant of the Hodge bundle". In: *Transactions of the American Mathematical Society* 352.6 (2000), pp. 2553–2568 (cit. on p. 41).

[Kup05] G. Kuperberg. "A subexponential-time quantum algorithm for the dihedral hidden subgroup problem". In: *SIAM Journal on Computing* 35.1 (2005), pp. 170–188 (cit. on p. 2).

[Lab16] H. Labrande. "Explicit computation of the Abel-Jacobi map and its inverse". PhD thesis. 2016 (cit. on pp. 6, 12, 93–95, 97, 100, 118, 124, 125, 142).

[Lab18] H. Labrande. "Computing Jacobi's theta in quasi-linear time". In: *Mathematics of Computation* 87.311 (2018), pp. 1479–1508 (cit. on pp. 94, 103).

[LT16] H. Labrande and E. Thomé. "Computing theta functions in quasi-linear time in genus two and above". In: *LMS Journal of Computation and Mathematics* 19.A (2016), pp. 163–177 (cit. on p. 94).

[LO77] J. C. Lagarias and A. M. Odlyzko. "Effective versions of the Chebotarev density theorem". In: Academic press, New York, 1977, pp. 409–464 (cit. on p. 144).

[LV16] P. Lairez and T. Vaccon. "On p-adic differential equations with separation of variables". In: *Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation*. 2016, pp. 319–323 (cit. on p. 78).

[Lan58] S. Lang. "Reciprocity and Correspondences". In: *American Journal of Mathematics* 80.2 (1958), pp. 431–440 (cit. on pp. 55, 59).

[LV14] K. Lauter and B. Viray. "Denominators of Igusa class polynomials". In: *Publications mathématiques de Besançon* 2 (2014), pp. 5–29 (cit. on p. 146).

[LL03] R. Lercier and D. Lubicz. "Counting Points on Elliptic Curves over Finite Fields of Small Characteristic in Quasi Quadratic Time". In: *Advances in Cryptology—EUROCRYPT '2003*. Ed. by E. Biham. Lecture Notes in Computer Science. Springer-Verlag, May 2003 (cit. on p. 133).

[LL06] R. Lercier and D. Lubicz. "A quasi-quadratic time algorithm for hyperelliptic curve point counting". In: *Ramanujan J.* 12.3 (2006), pp. 399–423 (cit. on pp. 133, 136).

[LS08] R. Lercier and T. Sirvent. "On Elkies subgroups of ℓ-torsion points in elliptic curves defined over a finite field." In: *Journal de théorie des nombres de Bordeaux* 20.3 (2008), pp. 783–797 (cit. on pp. 12, 78, 111).

[LLG+20] R. Lercier, Q. Liu, E. L. García, and C. Ritzenthaler. "Reduction type of smooth quartics". 2020. arXiv: 1803.05816 [math.NT] (cit. on p. 4).

[MR08] V. Maillot and D. Rössler. "On the determinant bundles of abelian schemes". In: *Compositio Mathematica* 144.2 (2008), pp. 495–502 (cit. on p. 41).

[Mes72] W. Messing. "The crystals associated to Barsotti-Tate groups". In: *The crystals associated to Barsotti-Tate groups: with applications to abelian schemes*. Springer, 1972, pp. 112–149 (cit. on pp. 127, 128).

[Mes01] J.-F. Mestre. *Lettre à Gaudry et Harley*. 2001. URL: http://www.math.jussieu.fr/mestre (cit. on pp. 132, 133).

[Mes02] J.-F. Mestre. *Notes of a talk given at the Cryptography Seminar Rennes*. 2002. URL: http://www.math.univ-rennes1.fr/crypto/2001-02/mestre.ps (cit. on pp. 133, 135, 136).

[Mil20] E. Milio. "Computing isogenies between Jacobians of curves of genus 2 and 3". In: *Mathematics of Computation* 89.323 (2020), pp. 1331–1364. arXiv: 1709.06063 (cit. on pp. 12, 80).

[Mil86] V. Miller. "Short programs for functions on curves". In: *Unpublished manuscript* 97 (1986), pp. 101–102 (cit. on pp. 57, 61).

[Mil04] V. S. Miller. "The Weil Pairing, and Its Efficient Calculation". In: *J. Cryptology* 17.4 (2004), pp. 235–261. DOI: 10.1007/s00145-004-0315-8 (cit. on p. 57).

[Mil91] J. Milne. *Abelian varieties*. 1991. URL: http://www.jmilne.org/math/CourseNotes/av.html (cit. on p. 68).

[Mil06] J. S. Milne. *Complex multiplication*. 2006. URL: https://www.jmilne.org/math/CourseNotes/cm.html (cit. on p. 143).

[MN17a] P. Molin and C. Neurohr. "Computing period matrices and the Abel-Jacobi map of superelliptic curves". 2017. arXiv: 1707.07249 (cit. on pp. 95, 125).

[MN17b] P. Molin and C. Neurohr. *hcperiods: Arb and Magma packages for periods of superelliptic curves*. 2017 (cit. on pp. 95, 125).

[MGE12] B. Moonen, G. van der Geer, and B. Edixhoven. *Abelian varieties*. Book project, 2012. URL: https://www.math.ru.nl/~bmoonen/research.html#bookabvar (cit. on p. 86).

[Mor95] F. Morain. "Calcul du nombre de points sur une courbe elliptique dans un corps fini: aspects algorithmiques". In: *J. Théor. Nombres Bordeaux* 7 (1995), pp. 255–282 (cit. on p. 5).

[Mor85] L. Moret-Bailly. *Pinceaux de variétés abéliennes*. Société mathématique de France, 1985 (cit. on p. 41).

[Mor90] L. Moret-Bailly. "Sur l'équation fonctionnelle de la fonction thêta de Riemann". In: *Compositio Mathematica* 75.2 (1990), pp. 203–217 (cit. on p. 41).

[Mum66] D. Mumford. "On the equations defining abelian varieties. I". In: *Invent. Math.* 1 (1966), pp. 287–354 (cit. on pp. 3, 6, 18, 20–28, 30, 31, 33, 36, 38, 43, 45, 87).

[Mum67a] D. Mumford. "On the equations defining abelian varieties. II". In: *Invent. Math.* 3 (1967), pp. 75–135 (cit. on pp. 3, 18, 20, 21, 32, 36, 41, 87).

[Mum67b] D. Mumford. "On the equations defining abelian varieties. III". In: *Invent. Math.* 3 (1967), pp. 215–244 (cit. on pp. 3, 18, 20, 21, 36).

[Mum69] D. Mumford. "Varieties defined by quadratic equations". In: *Questions on Algebraic Varieties (CIME, III Ciclo, Varenna, 1969)* (1969), pp. 29–100 (cit. on pp. 21, 50).

[Mum70a] D. Mumford. *Abelian varieties*. Tata Institute of Fundamental Research Studies in Mathematics, No. 5. Published for the Tata Institute of Fundamental Research, Bombay, 1970, pp. viii+242 (cit. on pp. 19, 20, 65).

[Mum70b] D. Mumford. *The structure of the moduli spaces of curves and abelian varieties*. Mathematics Institute, University of Warwick, 1970 (cit. on p. 32).

[Mum83] D. Mumford. *Tata lectures on theta I*. Vol. 28. Progress in Mathematics. With the assistance of C. Musili, M. Nori, E. Previato and M. Stillman. Boston, MA: Birkhäuser Boston Inc., 1983, pp. xiii+235. ISBN: 3-7643-3109-7 (cit. on pp. 20, 28, 36).

[Mum84] D. Mumford. *Tata lectures on theta II*. Vol. 43. Progress in Mathematics. Jacobian theta functions and differential equations, With the collaboration of C. Musili, M. Nori, E. Previato, M. Stillman and H. Umemura. Boston, MA: Birkhäuser Boston Inc., 1984, pp. xiv+272. ISBN: 0-8176-3110-0 (cit. on pp. 20, 50, 53).

[Mum91] D. Mumford. *Tata lectures on theta III*. Vol. 97. Progress in Mathematics. With the collaboration of Madhav Nori and Peter Norman. Boston, MA: Birkhäuser Boston Inc., 1991, pp. viii+202. ISBN: 0-8176-3440-1 (cit. on pp. 21, 36).

[MFK94] D. Mumford, J. Fogarty, and F. Kirwan. *Geometric invariant theory*. Vol. 34. Springer Science & Business Media, 1994 (cit. on pp. 32, 127).

[NR19] M. Naehrig and J. Renes. "Dual isogenies and their application to public-key compression for isogeny-based cryptography". In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2019, pp. 243–272 (cit. on p. 64).

[Nar18] A. K. Narayanan. "Fast computation of isomorphisms between finite fields using elliptic curves". In: *International Workshop on the Arithmetic of Finite Fields*. Springer. 2018, pp. 74–91 (cit. on p. 6).

[NS16] A. Neumaier and D. Stehlé. "Faster LLL-type reduction of lattice bases". In: *Proceedings of the ACM on International Symposium on Symbolic and Algebraic Computation*. 2016, pp. 373–380 (cit. on pp. 95, 102, 105, 106, 142, 145).

[NSV11] A. Novocin, D. Stehlé, and G. Villard. "An LLL-reduction algorithm with quasi-linear time complexity". In: *Proceedings of the forty-third annual ACM symposium on Theory of computing*. 2011, pp. 403–412 (cit. on pp. 95, 102, 142, 145).

[Oor71] F. Oort. "Finite group schemes, local moduli for abelian varieties, and lifting problems". In: *Compositio Mathematica* 23.3 (1971), pp. 265–296 (cit. on pp. 32, 127, 128).

[21] *PARI/GP version* 2.14. The PARI Group. Univ. Bordeaux, 2021. URL: http://pari.math.u-bordeaux.fr/ (cit. on p. 3).

[Per16] S. Perna. "Heat equation for theta functions and vector-valued modular forms". 2016. arXiv: 1510.03384 [math.AG] (cit. on p. 126).

[Poloo] A. Polishchuk. "Determinant bundles for abelian schemes". In: *Compositio Mathematica* 121.3 (2000), pp. 221–245 (cit. on p. 41).

[Ray85] M. Raynaud. "Hauteurs et isogénies". In: *Astérisque* 127 (1985), pp. 199–234 (cit. on p. 99).

[Ric36] F. Richelot. "Essai sur une méthode générale pour déterminer la valeur des intégrales ultra-elliptiques, fondée sur des transformations remarquables de ces transcendantes". In: *C. R. Acad. Sci. Paris* 2 (1836), pp. 622–627 (cit. on p. 133).

[Ric37] F. Richelot. "De transformatione Integralium Abelianorum primiordinis commentation". In: *J. reine angew. Math.* 16 (1837), pp. 221–341 (cit. on p. 133).

[Rito3] C. Ritzenthaler. "Problèmes arithmétiques relatifs à certaines familles de courbes sur les corps finis". PhD thesis. Université Denis Diderot Paris VII, June 2003 (cit. on pp. 133, 136).

[Rit10] C. Ritzenthaler. "Explicit computations of Serre's obstruction for genus-3 curves and application to optimal curves". In: *LMS Journal of Computation and Mathematics* 13 (2010), pp. 192–207 (cit. on p. 122).

[RSo6] A. Rostovtsev and A. Stolbunov. "Public-key cryptosystem based on isogenies". In: *International Association for Cryptologic Research. Cryptology ePrint Archive* (2006). eprint: http://eprint.iacr.org/2006/145 (cit. on pp. 2, 6).

[SSTo3] T. Satoh, B. Skjernaa, and Y. Taguchi. "Fast Computation of Canonical Lifts of Elliptic curves and its Application to Point Counting". In: *Finite Fields and Their Applications* 9.1 (2003), pp. 89–101 (cit. on p. 132).

[Satoo] T. Satoh. "The canonical lift of an ordinary elliptic curve over a finite field and its point counting". In: *J. Ramanujan Math. Soc.* 15.4 (2000), pp. 247–270 (cit. on pp. 5, 128, 129, 131, 132).

[Sch84] A. Schönhage. "Factorization of univariate integer polynomials by Diophantine approximation and an improved basis reduction algorithm". In: *International Colloquium on Automata, Languages, and Programming*. Springer. 1984, pp. 436–447 (cit. on p. 106).

[Sch85] R. Schoof. "Elliptic curves over finite fields and the computation of square roots mod p". In: *Mathematics of computation* 44.170 (1985), pp. 483–494 (cit. on p. 5).

[Sch95] R. Schoof. "Counting points on elliptic curves over finite fields". In: *J. Théor. Nombres Bordeaux* 7.1 (1995), pp. 219–254 (cit. on pp. 5, 77, 107).

[Sea13] G. J. Seal. *Tensors, monads and actions*. 2013. arXiv: 1205.0101 [math.CT] (cit. on p. 1).

[Ser75] J.-P. Serre. *Groupes algébriques et corps de classes*. 2nd. Vol. 7. Publications de l'Institut de mathématique de l'Université de Nancago. Hermann, 1975 (cit. on p. 61).

[Ser01] J. Serre. "Appendix to Geometric methods for improving the upper bounds on the number of rational points on algebraic curves over finite fields, by K". In: *Lauter. J. Algebraic Geom* 10.1 (2001), pp. 30–36 (cit. on p. 122).

[She08] N. Shepherd-Barron. "Thomae's formulae for non-hyperelliptic curves and spinorial square roots of theta-constants on the moduli space of curves". In: (2008) (cit. on p. 53).

[Sho94] P. W. Shor. "Algorithms for quantum computation: discrete logarithms and factoring". In: *Proceedings 35th annual symposium on foundations of computer science*. Ieee. 1994, pp. 124–134 (cit. on p. 2).

[Shp96] I. E. Shparlinski. "On irreducible polynomials of small height over finite fields". In: *Applicable Algebra in Engineering, Communication and Computing* 7.6 (1996), pp. 427–431 (cit. on p. 103).

[Sil94] J. H. Silverman. *Advanced topics in the arithmetic of elliptic curves*. Vol. 151. Graduate Texts in Mathematics. New York: Springer-Verlag, 1994, pp. xiv+525. ISBN: 0-387-94328-5 (cit. on p. 141).

[Sma03] N. Smart. "An analysis of Goubin's refined power analysis attack". In: *Cryptographic Hardware and Embedded Systems-CHES 2003* (2003), pp. 281–290 (cit. on p. 6).

[Smi08] B. Smith. "Isogenies and the discrete logarithm problem in Jacobians of genus 3 hyperelliptic curves". In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2008, pp. 163–180. arXiv: 0806.2995 [math.NT] (cit. on p. 6).

[Stacks] T. Stacks Project Authors. *Stacks Project*. https://stacks.math.columbia.edu. 2018 (cit. on p. 11).

[Sto10] A. Stolbunov. "Constructing public-key cryptographic schemes based on class group action on a set of isogenous elliptic curves". In: *Advances in Mathematics of Communications* 4.2 (2010), p. 215 (cit. on pp. 2, 6).

[Str10] M. Streng. "Complex multiplication of abelian surfaces". Proefschrift. Universiteit Leiden, 2010 (cit. on pp. 85, 86, 95, 142, 144, 146).

[Sug14] K.-i. Sugiyama. "On a generalization of Deuring□s results". In: *Finite Fields and Their Applications* 26 (2014), pp. 69–85 (cit. on p. 143).

[Sut11] A. Sutherland. "Computing Hilbert class polynomials with the Chinese remainder theorem". In: *Mathematics of Computation* 80.273 (2011), pp. 501–538 (cit. on pp. 5, 144, 145).

[Sut13] A. Sutherland. "On the evaluation of modular polynomials". In: *The Open Book Series* 1.1 (2013), pp. 531–555 (cit. on p. 103).

[SZ17] A. Sutherland and D. Zywina. "Modular curves of prime-power level with infinitely many rational points". In: *Algebra & Number Theory* 11.5 (2017), pp. 1199–1229 (cit. on p. 6).

[Tat66] J. Tate. "Endomorphisms of abelian varieties over finite fields". In: *Inventiones mathematicae* 2.2 (1966), pp. 134–144 (cit. on p. 140).

[Tes06] E. Teske. "An elliptic curve trapdoor system". In: *Journal of cryptology* 19.1 (2006), pp. 115–133 (cit. on p. 6).

[Tia20] S. Tian. "Translating the discrete logarithm problem on Jacobians of genus 3 hyperelliptic curves with (ℓ, ℓ, ℓ)-isogenies". 2020. arXiv: 2007.03172 [math.AG] (cit. on p. 12).

[Tra14] C. Tran. "Formules d'addition sur les jacobiennes de courbes hyperelliptiques: application à la cryptographie". PhD thesis. Rennes 1, 2014 (cit. on p. 8).

[Tui16] J. Tuitman. "Counting points on curves using a map to $\boldsymbol{P}^1$". In: *Mathematics of Computation* 85.298 (2016), pp. 961–981 (cit. on p. 139).

[VV09] B. Vallée and A. Vera. "Probabilistic analyses of lattice reduction algorithms". In: *The LLL Algorithm*. Springer, 2009, pp. 71–143 (cit. on p. 103).

[VHI06] W. Van Dam, S. Hallgren, and L. Ip. "Quantum algorithms for some hidden shift problems". In: *SIAM Journal on Computing* 36.3 (2006), pp. 763–778 (cit. on p. 2).

[Van98] P. Van Wamelen. "Equations for the Jacobian of a hyperelliptic curve". In: *Transactions of the American Mathematical Society* 350.8 (1998), pp. 3083–3106 (cit. on p. 52).

[Vél71] J. Vélu. "Isogénies entre courbes elliptiques". In: *Compte Rendu Académie Sciences Paris Série A-B* 273 (1971), A238–A241 (cit. on pp. 5, 66, 77).

[VPV01] F. Vercauteren, B. Preneel, and J. Vandewalle. "A memory efficient version of Satoh's algorithm". In: *International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2001, pp. 1–13 (cit. on p. 129).

[Vil18] G. Villard. "On computing the resultant of generic bivariate polynomials". In: *Proceedings of the 2018 ACM International Symposium on Symbolic and Algebraic Computation*. 2018, pp. 391–398 (cit. on p. 113).

[Wat69] W. Waterhouse. "Abelian varieties over finite fields". In: *Ann. Sci. Ecole Norm. Sup* 2.4 (1969), pp. 521–560 (cit. on p. 124).

[YZ18] X. Yuan and S.-W. Zhang. "On the averaged Colmez conjecture". In: *Annals of Mathematics* 187.2 (2018), pp. 533–638 (cit. on p. 144).

[ZSP+18] G. H. Zanon, M. A. Simplicio, G. C. Pereira, J. Doliskani, and P. S. Barreto. "Faster isogeny-based compressed key agreement". In: *International Conference on Post-Quantum Cryptography*. Springer. 2018, pp. 248–268 (cit. on p. 64).

[ZLR10] A. Zykin, G. Lachaud, and C. Ritzenthaler. "Jacobians among abelian threefolds: A formula of klein and a question of serre". In: *Doklady Mathematics*. Vol. 81. 2. Springer. 2010, pp. 233–235. arXiv: 0802.4017 (cit. on p. 122).