Elliptic Nets and Elliptic Curves

BY

Katherine E. Stange

B. Math., University of Waterloo, 2001

M. Sc., Brown University, 2003

Submitted in partial fulfillment of the requirements
for the Degree of Doctor of Philosophy in the
Department of Mathematics at Brown University

Providence, Rhode Island

May 2008

This dissertation by Katherine E. Stange is accepted in its present form by the Department of Mathematics as satisfying the dissertation requirement for the degree of Doctor of Philosophy.

Date _____ _____
Joseph H. Silverman, Director

Recommended to the Graduate Council

Date _____ _____
Alexander Goncharov, Reader

Date _____ _____
Stephen Lichtenbaum, Reader

Date _____ _____
Joseph H. Silverman, Reader

Approved by the Graduate Council

Date _____ _____
Sheila Bonde
Dean of the Graduate School

Vita

Katherine E. Stange was assembled in North Bay, Ontario, Canada in 1978. She was further processed at the University of Waterloo, where she was labelled “Bachelor of Mathematics” in 2001.

Acknowledgements

There are a great many people who have helped me along the long road that led, finally, to this thesis. First among these is my advisor, Joseph Silverman. Thank you for your incredible patience and for your enthusiastic telling of wonderful mathematical tales every Wednesday morning. And thank you for your confidence in me. It is hard to put into words exactly what an advisor does, because it happens between the practical matters of correcting papers and answering confusions. It has to do with learning what mathematics is, and it is why we care about our mathematical genealogy. I'm proud to have you as a mathematical father.

This work was supported for two years by the National Sciences and Engineering Research Council of Canada in the form of a post-graduate scholarship, for which I am very grateful. I also wish to thank Microsoft Research for providing me with a one-semester internship working with Kristin Lauter in San Diego, California. Thank you, Kristin, for your amazing generosity, infectious curiosity and your confidence in my work.

Thank you to Audrey, Carol, Doreen, Larry and Natalie. Without each of you, I wouldn't have ever found my way long enough to get to the math at all. Doreen and Natalie, you deserve special thanks for your labours on my behalf, and for your love, both of which were above and beyond, and for which I owe you more than I can give.

Thank you to my readers, Stephen Lichtenbaum and Alexander Goncharov. Thank you to Michael Rosen for your interest and useful discussions. Thank you to Alf van der Poorten for your support and enthusiasm. To Thomas Banchoff, for teaching me about teaching. To George Daskalopoulos, who was always especially kind. Thank you to everyone in the Brown mathematics department for being a good family to the graduate students.

Thank you to Christine Swart, Michael Scott, Graeme Everest, Gary McGuire, Peter Stevenhagen, Nelson Stephens and Noam Elkies for your interest, discussions and support. And to many others who took the time to discuss this work with me.

I want to thank my high school math teacher, Mr. Garrett, who is his own grandpa, and who always answered my interest with excitement. It was in your class, 15 years ago, that I decided to do this today. Thank you also to my teachers at the University of Waterloo, especially Peter Hoffman, Ken Davidson, Brian Forrest and Cam Stewart. Thank you also to Murat for your support.

Thank you to Donald Knuth for LaTeX, and to Mistress LaSpliffe and the New York Times for providing the necessary procrastination.

These are the people and things who made the mathematics possible. However, many of you I also count among my friends. And I must also thank those, not necessarily mathematically involved, whose friendship was every bit as important to the success of my crazy quest to become a mathematician as the mathematics itself.

This includes all of the mathematics department graduate students, who each in their way contributed to my upbringing, mathematical and otherwise. Thank you to Panayotis for being a loveably argumentative Greek. To Dan for putting us into song. To Steve for telling us jungle tales. To Mikey G for balls. To Alina for the challenge. To Michelle for her generosity. To Karen. To Graeme. To the Mikes, Ben, Hatice, Steffan (for being Canadian), Ebru, Hyun, and Yu.

To Michael I owe more than thanks, for you are family.

To Rafe, for wonderful times. I miss you.

To those in the wider community of Brown whose paths I crossed and learned from. In particular, thank you to Marie. To Joanna and Anita, for your open and accepting friendship. To Becca, for fascinating conversation. To Kevin, for the Fez.

To Lionel, for letting me in.

To David, for the experiences, the filters, the words.

To all those I rode with. If graduate school was a journey, I did it by bicycle. Thank you to the ECCC for the haybales and portajohns and all the rest. To Providence Bicycle. To Joe for swearing at me when I got the townlines. To Forest for trying to ride me off your wheel while eating ice cream, and putting up with me when I hung on. To the Brown Cycling Club for suffering my dictatorship. A special thank you to Bob and Bikeworks for your incredible support and enthusiasm while I tried my legs at racing. And thank you to Preston, for every time you agreed to go longer.

Thank you also to all those, named and nameless, whom I met in my travels abroad. Your strange customs but universal generosity were a gift. To Yousra, Ahmed, Denis, Victor, Vika, Katya, Tatyana Makarovna, Cody, Galina, Alain, Barbara, Patrick, Hanna, Spring. Especially to Yulia and Lera, and to Tenzin, Dukgyal and Nordron. Thank you to Laura, Ruxandra, Dan, Heidrun, Voichitsa, and Helmut, and especially to Oma Oma. And to many others whose names I have since lost, or never knew.

Thank you to Turtle Soup: to Meg, Helen, Kirsti, Lisa, Leah and Jessica. I wouldn't have survived high school without H.O.T. and quinzees and the Eighteenth Ruler of the Universe, without Bump the King of Dorks and the belly button tree, without torso dancing and Psycho-pro-tection, without the nickname Chubbo, without Dr. Niel Paterson or Cookie Dough. These things, ironically, made me sane. And without the freedom and creativity of our strange community, without all of you to convince me that anything was possible, I wouldn't have believed I could do it.

And finally, there are those for whom thanks are simply inadequate.

For Jonathan: How you managed to stay convinced that I was a mathematician I don't know. But it is only thanks to that delusion that I really am. You've taught me anew how to love mathematics, and whenever that relationship was going poorly, your bad puns laughed me back from the brink. You've been my co-conspirator in a shameless neverending childhood. You've given me mathematics, but you've given me much more: you've given me the world to do it in. I can't find any words besides

these[1]: I love you.

For Oma: Your love, your warmth, your interest, and your stories, have given me a cherished sense of connection between what I do now and where I come from. And thank you for the baked apples.

For Christiaan, who derailed any attempt I ever had to be serious: My life would have been impoverished without the Zappa, the sizzling floor, the Hee Haw in the Hole, without you spinning Marley on the linoleum. You taught me to read with raisins and Pavlovian conditioning. And you rode your bike with me, always just a little faster. I think it was those bicycle rides that taught me tenacity, which is still what I credit for anything I may go so far as to say I have *earned*.

And last, for my parents, mom and dad: How does one thank one's parents? Without you, I wouldn't have had anyone to count up the myriad of small boxes and wedges I drew with glee all over scraps of paper, to count those filled and those unfilled and pronounce the delightful name of the fraction I had created. Without you, I wouldn't know that six times eight is forty-eight. (With a little more of you, I might have known that seven times eight is fifty-six.) Without you, I wouldn't have been drawn wide-eyed into thinking about whether there were more integers than there were even integers. Or whether the universe was infinite, or how many different kinds of tastebuds we had. Without you, I wouldn't have had the confidence to take 'contest math.' And without you, I wouldn't have been happy and laughing and warm and strong when I did. Hell, without you, I wouldn't even *exist*.

[1] With the possible exception of certain Frank Zappa lyrics.

Contents

List of Tables

List of Figures

List of Examples

List of Algorithms

Part I

Overture

Chapter 1

Introduction

This thesis is a retelling and elaboration of a delightful story about recurrence sequences first told in 1948. The original tale of recurrence relations and elliptic curves is due to Morgan Ward[1] [74]. He himself, in turn, was retelling (and enhancing) an older story due to Edouard Lucas [41, 42], about linear recurrence sequences. To properly introduce our topic, then, we must start with Lucas and Ward.

1.1 Lucas' story

Lucas, writing in the first volume of the *American Journal of Mathematics* in 1878, was interested in symmetric functions of the roots of quadratic equations. Consider a quadratic equation

$$x^2 - Px + Q = 0,$$

with roots a and b (so $P = a + b$ and $Q = ab$). Then, define the symmetric functions

$$U_n = \frac{a^n - b^n}{a - b}, \quad \text{and} \quad V_n = a^n + b^n.$$

Lucas demonstrated that these remarkable functions have two special properties. First, they are generated by recurrence relations:

$$U_n = PU_{n-1} - QU_{n-2}, \quad \text{and} \quad V_n = PV_{n-1} - QV_{n-2}. \tag{1.1}$$

(A quick way to see this is to first show $a^{n+2} = Pa^{n+1} - Qa^n$ and $b^{n+2} = Pa^{n+1} - Qb^n$.) And second, they are defined by 'circular functions':

$$U_n = P^{\frac{n-1}{2}} \left(\frac{\sin(nx)}{\sin(x)} \right), \quad \text{and} \quad V_n = P^{\frac{n+1}{2}} \left(\frac{\cos(nx)}{\cos(x)} \right).$$

The recurrence relations for V_n and U_n can be considered instances of the addition/subtraction formulæ for trigonometric functions.

[1] See [39] for an overview of Ward's mathematical work.

Table 1.1: Special cases of Lucas sequences

(P,Q)	U_n	V_n
$(1,-1)$	Fibonacci Numbers	Lucas Numbers
$(2,-1)$	Pell Numbers	Pell-Lucas Numbers
$(1,-2)$	Jacobsthal Numbers	Jacobsthal-Lucas Numbers
$(3,2)$	Mersenne Numbers	2^n+1
$(b+1,b)$	multiples of Cunningham Numbers	

Examples of Lucas sequences are shown in Table 1.1. The Fibonacci numbers and Mersenne numbers in particular need no introduction: Lucas sequences in general and these special cases in particular have been a source of study for hundreds of years. Mathematicians, including Lucas, have continued to unearth connections to diverse corners of mathematics, and so it is that Lucas wrote

> Ce memoire à pour objet l'étude des fonctions symétriques des racines d'une équation du second degré, et son application à la théorie des nombres premiers. Nous indiquons dès le commencement, l'analogie complète de ces fonctions symétriques avec les fonctions circulaires et hyperboliques; nous montrons ensuite la liaison qui existe entre ces fonctions symétriques et les théories des déterminants, des combinaisons, des fractions continues, de la divisibilité, des diviseurs quadratiques, des radicaux continus, de la division de circonférence, de l'analyse indéterminée du second degré, des residus quadratiques, de la décomposition des grands nombres en facteurs premiers, etc. Cette méthode est le point du départ d'une étude plus complète, des propriétés des fonctions symétriques des racines d'une équation algébrique, de degré quelconque, à coefficients commensurables, dans leurs rapports avec les théories des fonctions elliptiques et abélienne, des résidus potentiels, et de l'analyse indeterminee [sic] des degrés supérieurs. [2][42, p.184]

Ward took up the challenge.[3]

1.2 Ward's story

Ward studied integral sequences satisfying a certain recurrence relation. We generalise his definition to arbitrary integral domains.

Definition 1.2.1. Let R be an integral domain. An *elliptic divisibility sequence* is a function $W:\mathbb{Z}\to R$ satisfying

$$W(n+m)W(n-m)W(1)^2 = W(n+1)W(n-1)W(m)^2 - W(m+1)W(m-1)W(n)^2. \qquad (1.2)$$

[2] This memoire has as its object the study of the symmetric functions of the roots of a quadratic equation, and its application to the theory of prime numbers. We indicate at the start the complete analogy of these symmetric functions with the circular and hyperbolic functions; then we show the connections between these symmetric functions and the theory of determinants, combinations, continued fractions, divisibility, quadratic divisors, continued radicals, the division of the circumference, the analysis of indeterminates of the second degree, quadratic residues, the prime factorisation of large numbers, etc. This method is the starting point of a more complete study of the properties of symmetric functions of the roots of an algebraic equation, of any degree, of rational coefficients, in their relation with the theory of elliptic and abelian functions, potential residues, and the analysis of indeterminates of higher degree.

[3] Lucas apparently claimed in other scattered hints to have found such a generalisation involving elliptic functions. This generalisation will be Ward's story, because Lucas never published anything about it. See [20, §10.1].

Ward made the assumption that $W(1)=1$ and so left the $W(1)^2$ out of the left side of this relation. By restoring that factor we obtain a homogeneous equation in the sense that if W is an elliptic divisibility sequence and c is a constant then cW is also an elliptic divisibility sequence. Thus, we loose nothing essential in our study if we, too, frequently assume for simplicity that $W(1)=1$.

Ward also was interested in integer sequences, and required in his definition that $W(n)|W(m)$ whenever $n|m$. We will not make that assumption, since we intend to work over more general rings, but we will discuss the integer case in Chapter 2.

Lucas' U_n were special values of the cyclotomic polynomials

$$\Theta_n(x)=\frac{x^n-1}{x-1}.$$

For example,

$$\Theta_1(x)=1, \qquad \Theta_2(x)=x+1, \qquad \Theta_3(x)=x^2+x+1, \qquad \Theta_4(x)=x^3+x^2+x+1.$$

In fact,

$$U_n=b^{1-n}\Theta_n(a/b).$$

The cyclotomic polynomials are those whose roots are the 'n-torsion points' on the unit circle–that is to say, the n-th roots of unity.

For an elliptic curve

$$y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6, \tag{1.3}$$

the analogous polynomials are called *division polynomials*. These are the polynomials Ψ_n in x,y that vanish exactly at the n-torsion points (and whose poles are supported at the identity). Define the usual quantities

$$\begin{aligned} b_2&=a_1^2+4a_2, \quad b_4=2a_4+a_1a_3, \\ b_6&=a_3^2+4a_6, \quad b_8=a_1^2a_6+4a_2a_6-a_1a_3a_4+a_2a_3^2-a_4^2. \end{aligned} \tag{1.4}$$

The first few division polynomials are

$$\begin{aligned} &\Psi_1=1, \qquad \Psi_2=2y+a_1x+a_3, \\ &\Psi_3=3x^4+b_2x^3+3b_4x^2+3b_6x+b_8, \\ &\Psi_4=(2y+a_1x+a_3)(2x^6+b_2x^5+5b_4x^4+10b_6x^3+10b_8x^2+(b_2b_8-b_4b_6)x+b_4b_8-b_6^2). \end{aligned} \tag{1.5}$$

Of course, giving the zeroes and poles doesn't exactly determine the polynomial. One way to fix the definition is to require the leading coefficient of Ψ_n to be n (where the variable x is assigned weight 2 and the variable y is assigned weight 3). More frequently, the polynomials are defined in the first place by giving the first four of them as above, and requiring the rest to be determined by the recurrence relations

$$\begin{aligned} \Psi_{2m+1}&=\Psi_{m+2}\Psi_m^3-\Psi_{m-1}\Psi_{m+1}^3, \\ 2y\Psi_{2m}&=\Psi_m(\Psi_{m+2}\Psi_{m-1}^2-\Psi_{m-2}\Psi_{m+1}^2). \end{aligned} \tag{1.6}$$

Morgan Ward, for his part, defined the division polynomials only over the complex numbers, using the complex analytic theory of elliptic functions. That is, consider the Ψ_n as functions of a complex variable z representing a point on an elliptic curve. Recall that a complex lattice Λ determines an elliptic curve over $\mathbb{C}$, and that the Weierstrass sigma function $\sigma : \mathbb{C} \to \mathbb{C}$ can be used to 'build' elliptic functions.

Definition 1.2.2. Fix a lattice $\Lambda \in \mathbb{C}$ corresponding to an elliptic curve E. For $n \in \mathbb{Z}$, define a function Ψ_n on $\mathbb{C}$ in the variable z as follows:

$$\Psi_n(z;\Lambda) = \frac{\sigma(nz;\Lambda)}{\sigma(z;\Lambda)^{n^2}}$$

(For $n = 0$, set $\Psi_n = 0$.)

It is then a straightforward exercise to show, using complex function theory, that, first, the Ψ_1, Ψ_2, Ψ_3 and Ψ_4 agree with (1.5), and second, that the Ψ_n satisfy (1.2) (in the variable n). Note that equations (1.6) are special cases of (1.2). Ward went on to show that every elliptic divisibility sequence satisfying certain conditions arises from an elliptic curve.

Theorem 1.2.1 ([74, Thm 12.1]). *If W is an elliptic divisibility sequence of integers satisfying $W(1) = 1$, $W(2)W(3) \neq 0$, and $W(2)|W(4)$, then there exists an elliptic curve (given by a lattice Λ) and a complex constant z such that*

$$W(n) = \Psi_n(z;\Lambda) = \frac{\sigma(nz;\Lambda)}{\sigma(z;\Lambda)^{n^2}}.$$

Suppose the lattice Λ determines an elliptic curve E and z determines a point $P \in E$. We will say W is the elliptic divisibility sequence associated to E and P and write $W_{E,P}(n)$ for the sequence.

Thus, we have much the same story as Lucas told sixty years earlier:

elliptic divisibility sequences	$\leftrightarrow$	Lucas sequences (U_n)
recurrence (1.2)	$\leftrightarrow$	recurrence (1.1)
division polynomials	$\leftrightarrow$	cyclotomic polynomials
elliptic functions	$\leftrightarrow$	trigonometric functions
elliptic curve	$\leftrightarrow$	multiplicative group

Since a singular cubic curve has a group law on its non-singular points, and in the multiplicative case, this becomes $\mathbb{G}_m$, it is perhaps not entirely surprising that the Lucas sequences U_n satisfy the elliptic divisibility sequence recurrence (1.2); in fact, Ward first encountered the equation in Lucas' paper [42, p.204].

Much work has been done on elliptic divisibility sequences since they were defined by Ward: they are a natural starting point in the study of non-linear recurrence sequences, and have a host of interesting properties arising from this interplay of number theory and geometry. For example, it is a result of Silverman that each term beyond some finite bound is divisible by some prime which is not a divisor of any previous term [62]; further conjectures in this vein have recently opened approaches to Hilbert's tenth problem for the rationals [12]. There have also been applications to cryptography [61] and partial

difference equations [32]. Chapter 2 surveys the background results about elliptic divisibility sequences used in this thesis.

1.3 Moving to higher rank

The purpose of this thesis is to introduce and study a generalisation of elliptic divisibility sequences to higher dimension and to general fields. An *elliptic net* is a function $W : A \to R$ from a finite rank free abelian group A to an integral domain R satisfying the property

$$\begin{aligned} &W(p+q+s)\,W(p-q)\,W(r+s)\,W(r) \\ &\quad + W(q+r+s)\,W(q-r)\,W(p+s)\,W(p) \\ &\quad + W(r+p+s)\,W(r-p)\,W(q+s)\,W(q) = 0 \end{aligned} \tag{1.7}$$

for all $p, q, r, s \in A$. If $A = R = \mathbb{Z}$, this is an equivalent definition of an elliptic divisibility sequence. By the *rank* of an elliptic net we shall mean the rank of A.

As is the case for rank one, elliptic nets of any rank are closely tied to elliptic curves. Part II of this thesis makes this correspondence explicit. Let K be any field. We generalise the concept of division polynomials to that of *net polynomials* $\Psi_{\mathbf{v}} \in E^n(K)$ for $\mathbf{v} \in \mathbb{Z}^n$, and show that these polynomials (actually rational functions) generate all elliptic nets from n-tuples of points $\mathbf{P} = (P_1, \ldots, P_n) \in E^n$. That is, the function $W_{E,\mathbf{P}} : \mathbb{Z}^n \to K$ given by

$$W_{E,\mathbf{P}}(\mathbf{v}) = \Psi_{\mathbf{v}}(\mathbf{P}) \tag{1.8}$$

is an elliptic net, and all elliptic nets from $\mathbb{Z}^n$ to K arise in this manner for some choice of curve E over K and n-tuple $\mathbf{P} \in E(K)^n$. Certain elliptic nets (called *singular*) arise from singular cubic Weierstrass curves.

The hope of defining such higher rank elliptic divisibility sequences via a recurrence was briefly discussed in correspondence by Noam Elkies, James Propp and Michael Somos in 2001 [55]. As we shall see in Section 2.2, over $\mathbb{Q}$, elliptic divisibility sequences arise as the denominators of the multiples $[n]P$ of the point P on E. Around the same time, Graham Everest, Victor Miller, Peter Rogers, Nelson Stephens and Thomas Ward mused about the collection of denominators of the points $[n]P + [m]Q$ (as n and m vary) and their number-theoretical properties [19, 21, 56].

Any rank one subnet of an elliptic net (i.e., the restriction to any line passing through the origin in the array of numbers) is an elliptic divisibility sequence. Christine Swart [68] and Alf van der Poorten [73] have also studied *translated elliptic divisibility sequences*, which correspond in this context to collections of terms of the form $W((a, nb))$ in an elliptic net defined on $A = \mathbb{Z}^2$ for fixed a and b as n varies (i.e., the restriction to any line not necessarily passing through the origin). Marco Streng studies a slightly different generalisation for elliptic curves with complex multiplication [67].

To be precise about the relationship given by equation (1.8), let us set some terminology. Suppose K is a field. We call a set of non-zero points $\{P_1, \ldots, P_n\}$ on the non-singular part C_0 of a cubic Weierstrass curve C *appropriate* if $P_i \neq \pm P_j$ for any $i \neq j$ and if $[2]P_1$ and $[3]P_1$ are nonzero in the case $n = 1$.

An elliptic net is called *degenerate* if it vanishes at any of the standard basis vectors or their pairwise sums or differences in $\mathbb{Z}^n$ (or, if $n = 1$, at 2 or 3). We call two elliptic nets W_1 and W_2 *scale equivalent* if $W_1(\mathbf{v}) = f(\mathbf{v})W_2(\mathbf{v})$ for some quadratic function $f : A \to K^*$. Finally, a change of variables is *unihomothetic* if it is of the form

$$\begin{aligned} x' &= x + r, \\ y' &= y + sx + t. \end{aligned}$$

Theorem 1.3.1 (Introductory version of Theorem 9.2.1)**.** *For each field K, there is an explicit bijection*

$$\left\{ \begin{array}{l} \textit{scale equivalence classes of} \\ \textit{non-degenerate elliptic nets} \\ W : \mathbb{Z}^n \to K \textit{ for some } n \end{array} \right\} \longleftrightarrow \left\{ \begin{array}{l} \textit{tuples } (C, P_1, \ldots, P_m) \textit{ for some } m, \textit{ where } C \\ \textit{is a cubic curve in Weierstrass form over } K, \\ \textit{considered modulo unihomothetic changes} \\ \textit{of variables, and such that } \{P_i\} \in C_0(K)^m \\ \textit{is appropriate} \end{array} \right\}.$$

These are partially ordered sets and the bijection preserves the ordering: elliptic nets are ordered by a natural notion of restriction (for any inclusion of their domains) and tuples of points are ordered by the subgroup ordering on the groups they generate. The bijection takes a net of rank k to a tuple with k points.

The proof of this bijection is the primary purpose of Part II. The structure of the proof is roughly as follows. The difficult part of the argument is to create the elliptic net from the elliptic curve. To do so over the complex numbers using Weierstrass' σ function is relatively straightforward: in this way we define functions $\Omega_{\mathbf{v}}$ and show that they form an elliptic net using complex function theory. We then wish to create the *net polynomials* $\Psi_{\mathbf{v}}$ analogous to the division polynomials, which should agree with the $\Omega_{\mathbf{v}}$ over $\mathbb{C}$. It is necessary to understand the recursive structure of an elliptic net sufficiently to do two things: first, give a generating collection of initial terms (a *baseset*) from which every other term can be calculated with applications of the recurrence relation; and second, control the amount of division required in such calculations to draw conclusions about the form of general terms as functions of the initial terms. We define the $\Psi_{\mathbf{v}}$ in general by calculating them explicitly from the $\Omega_{\mathbf{v}}$ on a baseset, and defining the rest recursively. Then, equality of the $\Omega_{\mathbf{v}}$ and $\Psi_{\mathbf{v}}$ on the baseset gives us equality on the entire elliptic net. Through our understanding of the recursive structure, we then proceed to determine conditions on the shape of the $\Psi_{\mathbf{v}}$ as polynomials in certain variables. In the rank one case, we can define an elliptic net associated with any elliptic curve over any field: we can do this because the division polynomials have $\mathbb{Z}$ coefficients and do not vanish modulo primes p. We show the corresponding conditions for net polynomials using the recurrence structure and some number-theoretical arguments, and this allows us to define an elliptic net from any curve and tuple of points over any field using a geometrical argument. Finally, we derive formulæ for going in the other direction, from net to curve, and, collecting our results, we have the bijection stated in Theorem 1.3.1.

In the course of proving the curve-net bijection, we discover numerous properties of elliptic nets and net polynomials. Among these is a Laurentness property.

Theorem 1.3.2 (Introductory version of Theorem 4.4.1)**.** *The terms of an elliptic net are generated*

by the recurrence relation from a finite set of initial terms. Furthermore, the terms are Laurent polynomials in a set of initial terms of size 4 *for rank one, and size no larger than* $3^n - 1$ *for rank* $n > 1$.

We also show that the net polynomials exist for any elliptic curve over any field, and have a certain restrictive form. In the following theorem δ and γ_4 are defined for the elliptic scheme in the theorem analogously to the usual way Δ and c_4 are defined for elliptic curves.

Theorem 1.3.3 (Introductory version of Theorem 7.1.1). *Let*

$$f(x,y) = y^2 + \alpha_1 xy + \alpha_3 y - x^3 - \alpha_2 x^2 - \alpha_4 x - \alpha_6$$

define an elliptic scheme $E_{\mathbb{Z}}$ *over the ring* $R = \mathbb{Z}[\alpha_1, \ldots, \alpha_6]$ *localised at* (δ) *and* (γ_4). *Let* $n \geq 1$.

There exist rational functions $\Psi_{\mathbf{v}}$ *on* $E_{\mathbb{Z}}^n$ *for each* $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{Z}^n$ *satisfying the following properties:*

1. *The* $\Psi_{\mathbf{v}}$ *satisfy the recurrence* (1.7) *in terms of* $\mathbf{v}$.
2. $\Psi_{\mathbf{v}} = 1$ *whenever* $\mathbf{v} = \mathbf{e}_i$ *for some* $1 \leq i \leq n$ *or* $\mathbf{v} = \mathbf{e}_i + \mathbf{e}_j$ *for some* $1 \leq i < j \leq n$. *(Here* $\mathbf{e}_1, \ldots, \mathbf{e}_n$ *represent the standard basis vectors in* $\mathbb{Z}^n$.*)*
3. *For each* $j = 1, \ldots, n$, *let* $p_j : E^n \to E$ *be the projection onto the* j*-th factor, and let* $s : E^n \to E$ *be summation of all factors. Then*

$$\operatorname{div}(\Psi_{\mathbf{v}}) = ([v_1] \times \ldots \times [v_n])^* s^*(\mathcal{O}) - \sum_{1 \leq k < j \leq n} v_k v_j (p_k^* \times p_j^*) s^*(\mathcal{O}) - \sum_{k=1}^{n} \left(2v_k^2 - \sum_{j=1}^{n} v_k v_j \right) p_k^*(\mathcal{O}).$$

4. *The* $\Psi_{\mathbf{v}}$ *can be expressed as polynomials in the ring*

$$\mathcal{R}_n = \mathbb{Z}[\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_6][x_i, y_i]_{i=1}^n \left[(x_i - x_j)^{-1} \right]_{1 \leq i < j \leq n} \Big/ \left\langle f(x_i, y_i) \right\rangle_{i=1}^n,$$

where the x_i *and* y_i *are the rational functions* $\wp$ *and* $\wp'$ *respectively on the* i*-th component of* $E_{\mathbb{Z}}^n$.

Among the properties Ward was interested in were some he called 'symmetry properties.' Consider an elliptic divisibility sequence, such as

$$\begin{aligned} &1, 1, -3, 11, 38, 249, -2357, 8767, 496035, -3769372, -299154043, \\ &-12064147359, 632926474117, -65604679199921, -6662962874355342, \\ &-720710377683595651, 285131375126739646739, \\ &5206174703484724719135, -360421577662469237888837209, \ldots \end{aligned} \tag{1.9}$$

This sequence, when reduced modulo 11, becomes

$$\begin{aligned} &1, 1, 8, 0, 5, 7, 8, 0, 1, 9, 10, 0, 3, 7, 6, 0, 3, 1, 10, 0, 1, 10, 8, 0, 5, 4, 8, \\ &0, 1, 2, 10, 0, 3, 4, 6, 0, 3, 10, 10, 0, 1, 1, 8, 0, 5, 7, 8, 0, 1, 9, \ldots \end{aligned} \tag{1.10}$$

Sequence (1.9) is the sequence associated to the curve $y^2+y=x^3+x^2-2x$ and point $P=(0,0)$, while (1.10) is the sequence associated to the curve and point under reduction modulo eleven, i.e., curve $y^2+y=x^3+x^2+9x$ and point $\tilde{P}=(0,0)$ over $\mathbb{F}_{11}$. The point $\tilde{P}$ has order 4 since this is the index at which the first zero appears in (1.10). However, the sequence evidently has period 40, illustrating the important remark that an elliptic divisibility sequence or elliptic net is *not* a function on the points of the curve (in this case, the cyclic group generated by P).

Ward showed that elliptic divisibility sequences modulo a prime, while not necessarily periodic with the same order as the associated point, do have a sort of 'partial periodicity' pattern with respect to that order. For any prime p, let r be the smallest positive index at which the sequence is divisible by p (in (1.10), this is 4). This r is called the *rank of apparition*, and is equal to the order of the associated point on the curve over $\mathbb{F}_p$. Ward showed that for any sequence and prime p, there exist integers a and b such that

$$W(k+sr) \equiv W(k)a^{ks}b^{s^2} \mod p$$

for all s and k (see Theorem 2.6.2). In the example above $a=8$ and $b=2$.

For higher rank n, we replace the notion of a rank of apparition with a *lattice of apparition*, the sublattice of $\mathbb{Z}^n$ of indices at which the net vanishes modulo p. In general, we call the sublattice of $\mathbb{Z}^n$ where an elliptic net vanishes a *lattice of zero-apparition*, so that the lattice of apparition mod p is the same as the lattice of zero-apparition for the elliptic net considered as taking values in $\mathbb{F}_p$. Ward's result generalises as follows.

Theorem 1.3.4 (Introductory version of Theorem 10.2.3). *Suppose that $W_{E,\mathbf{P}}$ is a non-degenerate elliptic net of rank n with values in a field K and with lattice of zero-apparition Γ. For any $\mathbf{r} \in \Gamma$ and $\mathbf{k} \in \mathbb{Z}^n \backslash \Gamma$, define*

$$g : \Gamma \times (\mathbb{Z}^n \backslash \Gamma) \to K^*$$

by

$$g(\mathbf{r},\mathbf{k}) = \frac{W_{E,\mathbf{P}}(\mathbf{r}+\mathbf{k})}{W_{E,\mathbf{P}}(\mathbf{k})}.$$

Then g is a quadratic function where defined, which is affine linear in the second factor in the sense that

$$g(\mathbf{r},\mathbf{k}_1+\mathbf{k}_2) - g(\mathbf{r},\mathbf{k}_1) - g(\mathbf{r},\mathbf{k}_2) + g(\mathbf{r},\mathbf{0}) = 0.$$

The proof depends on understanding how elliptic nets for the same curve relate to one another. The following very important 'transformation property' is used in the proof of Theorem 1.3.4 and repeatedly throughout the thesis.

Theorem 1.3.5 (Introductory version of Theorem 10.1.1). *Let T be any $n \times m$ matrix. Let $\mathbf{P} \in E^m$, $\mathbf{v} \in \mathbb{Z}^n$. Then*

$$W_{E,\mathbf{P}}(T^{tr}(\mathbf{v})) = W_{E,T(\mathbf{P})}(\mathbf{v}) \prod_{i=1}^{n} W_{E,\mathbf{P}}(T^{tr}(\mathbf{e}_i))^{v_i^2 - v_i(\Sigma_{j \neq i} v_j)} \prod_{1 \leq i < j \leq n} W_{E,\mathbf{P}}(T^{tr}(\mathbf{e}_i + \mathbf{e}_j))^{v_i v_j}$$

In particular, the two elliptic nets

$$W_{E,\mathbf{P}} \circ T^{tr} : \mathbb{Z}^n \to K, \quad \textit{and} \quad W_{E,T(\mathbf{P})} : \mathbb{Z}^n \to K$$

are scale equivalent.

1.4 Deeper connections

The proofs of partial periodicity for elliptic nets demonstrated in the last section flow eventually from the theory of elliptic functions over the complex numbers (when traced to its roots, the proof is a generalisation of the method of Ward demonstrated in Theorem 2.6.2). A central paradigm to the study of elliptic nets is that the arithmetic of the nets is explained by the geometry of the underlying curves. These proofs leave unanswered the question of the underlying geometry.

In Part III of the thesis, we partially answer this question. The answer lies in the theory of generalised Jacobians, biextensions and Tate-Lichtenbaum and Weil pairings for elliptic curves. Let E be an elliptic curve and S and T two points on that curve. One can form a notion of a Jacobian for the singular curve obtained from E by identifying S and T. This is a group called the *generalised Jacobian* and is an extension of E by $\mathbb{G}_m$. Whenever a sequence of group operations on points P_i in E results in the identity $\mathcal{O}$, one may take lifts of P_i in the generalised Jacobian and perform the same operations, in which case the result lies in the fibre over $\mathcal{O}$. This 'monodromy' is in some sense the source of the partial periodicity patterns in the elliptic divisibility sequence or net.

To make this precise, it is convenient to consider the Poincaré biextension, which is a variety X lying over $E \times E$:

$$\pi : X \to E \times E$$

with a $\mathbb{G}_m$ action. It has the property that its slices $\pi^{-1}(E \times \{P\})$ and $\pi^{-1}(\{P\} \times E)$ describe all of the generalised Jacobians of E. It can be formed from the Poincaré line bundle by deleting the zero section. The collection of biextensions can be described via cocyles and coboundaries, just as the group of extensions can. These cocycles are called *factor systems*. The factor system for the Poincaré biextension can be described in terms of elliptic nets, as follows.

Theorem 1.4.1 (Introductory version of Theorem 15.1.1). *Let E be an elliptic curve. The Poincaré biextension is given by a factor system consisting of a single map*

$$\Lambda : E \times E \times E \to \mathbb{G}_m$$

given by the formula

$$\Lambda(Q_1, Q_2, Q_3) = \frac{W(\mathbf{q}_1 + \mathbf{q}_2 + \mathbf{q}_3) W(\mathbf{q}_1) W(\mathbf{q}_2) W(\mathbf{q}_3)}{W(\mathbf{q}_1 + \mathbf{q}_2) W(\mathbf{q}_2 + \mathbf{q}_3) W(\mathbf{q}_3 + \mathbf{q}_1)},$$

where in this formula, W is an elliptic net associated to E and the points $\mathbf{T} = (P_1, \ldots, P_n)$, and the $\mathbf{q}_i$ are such that $\mathbf{q}_i \cdot \mathbf{T} = Q_i$ on the curve. The resulting value of Λ is independent of the choice of $\mathbf{T}$ (so the choice of $\mathbf{T}$ can depend on Q_1, Q_2, Q_3 in general).

Since the elliptic net satisfies a recurrence relation, we can make an interesting observation.

Theorem 1.4.2 (Introductory version of Theorem 15.2.1). *Let E be an elliptic curve. The Poincaré biextension for E admits a factor system consisting of one map Λ such that*

$$\Lambda(X_1, X_4+X_2, -X_2)+\Lambda(X_2, X_4+X_3, -X_3)+\Lambda(X_3, X_4+X_1, -X_1) = 0$$

for all non-zero points $X_1, X_2, X_3, X_4 \in E$ satisfying the condition that none of the expressions

$$X_4+X_i\ (i=1,2,3), \quad X_i-X_j\ (i,j=1,2,3, i\neq j), \quad X_4+X_i+X_j\ (i,j=1,2,3, i\neq j)$$

vanishes.

Another consequence of Theorem 1.4.1 is formulæ for the Tate-Lichtenbaum and Weil pairings on the elliptic curve. These pairings are usually defined using cohomological methods, which we review in Chapter 16. In Chapter 17, we give an abstract definition of two pairings for any biextension, and show that these are the Tate-Lichtenbaum and Weil pairings for the Poincaré biextension.

Theorem 1.4.3 (Introductory version of Theorems 17.2.1 and 17.2.2). *Let Q_1, Q_2, Q_3 be points on an elliptic curve E and let W be any elliptic net associated to E and points $\mathbf{T} = (P_1, \ldots, P_n)$ such that we can find $\mathbf{q}_i \in \mathbb{Z}^n$ for which $\mathbf{q}_i \cdot \mathbf{T} = Q_i$ on the curve.*

The Tate-Lichtenbaum pairing of $Q_1 \in E[m]$ and $Q_2 \in E$ is given by

$$\tau_m(Q_1, Q_2) = \frac{W(m\mathbf{q}_1+\mathbf{q}_2+\mathbf{q}_3)W(\mathbf{q}_3)}{W(m\mathbf{q}_1+\mathbf{q}_3)W(\mathbf{q}_2+\mathbf{q}_3)} \tag{1.11}$$

and the Weil pairing of $Q_1, Q_2 \in E[m]$ is given by

$$e_m(Q_1, Q_2) = \frac{W(m\mathbf{q}_1+\mathbf{q}_2+\mathbf{q}_3)W(\mathbf{q}_1+\mathbf{q}_3)W(m\mathbf{q}_2+\mathbf{q}_3)}{W(m\mathbf{q}_1+\mathbf{q}_3)W(\mathbf{q}_2+\mathbf{q}_3)W(\mathbf{q}_1+m\mathbf{q}_2+\mathbf{q}_3)}.$$

(These formulæ are independent of $\mathbf{q}_3$ and the choice of $\mathbf{T}$.)

Finally, we return to partial periodicity properties. Let $P \in E[m]$. The Tate-Lichtenbaum self pairing $\tau_m(P,P)$ in formula (1.11), for the elliptic net associated to the basis $\mathbf{T} = (P)$, becomes

$$\left(\frac{W_{E,P}(m+2)}{W_{E,P}(2)}\right)\left(\frac{W_{E,P}(1)}{W_{E,P}(m+1)}\right),$$

from which the reader may guess that there are relations to the partial periodicity properties (m is the rank of apparition of the sequence). In general, the formulæ above can be expressed in terms of the function g of Theorem 1.3.4. We have

$$\tau_m(P,Q) = \frac{g(m\mathbf{p}, \mathbf{q}+\mathbf{s})}{g(m\mathbf{p}, \mathbf{s})},$$

and

$$e_m(P,Q) = \frac{g(m\mathbf{p}, \mathbf{q}+\mathbf{s})g(m\mathbf{q}, \mathbf{s})}{g(m\mathbf{p}, \mathbf{s})g(m\mathbf{q}, \mathbf{p}+\mathbf{s})}.$$

1.5 Cryptographic applications

Elliptic nets have several applications to elliptic curve cryptography. The first is to provide a new algorithm for computing the Tate-Lichtenbaum and Weil pairings. We focus on the Tate-Lichtenbaum pairing and remark that the Weil pairing may be computed by two applications of the algorithm for the Tate-Lichtenbaum pairing (see Theorem 17.2.2).

The use of pairings in elliptic curve cryptography was originally suggested as a means of reducing the discrete logarithm problem on an elliptic curve to the discrete logarithm problem in a finite field [45, 23], but considerable excitement and research has since been generated by public-key cryptographic applications such as Sakai, Ohgishi and Kasahara's key agreement and signature schemes [59], Joux's tripartite Diffie-Hellman key exchange [35], and Boneh and Franklin's identity-based encryption scheme [7]. Good overviews of the research include [18, 53], while a very up-to-date research bibliography can be found at [3].

The bottleneck for implementations of pairing-based cryptographic protocols is the costly computation of the pairing, which is most frequently the Tate-Lichtenbaum or Weil pairing, the former usually being more efficient. Prior to the author's work, the only polynomial time algorithm was due to Victor Miller [47, 46] (for an overview of implementions, see [17, 26]).

Theorem 17.2.1 relates elliptic nets and the Tate-Lichtenbaum pairing: it reduces the calculation of the pairing to the calculation of terms of an elliptic net. Rachel Shipsey's thesis provides a double-and-add method of calculating the n-th term of an elliptic divisibility sequence in $\log n$ time [61]. The first step to providing an elliptic net algorithm for pairings is to generalise her algorithm to elliptic nets.

The elliptic net algorithm and Miller's algorithm are both $\log n$ algorithms; the difference is in the constants. In its nascent form, the elliptic net algorithm is only somewhat slower than an optimised Miller's, especially at higher embedding degrees. The elliptic net algorithm has no cryptographically significant restrictions on the points, curves or finite fields to which it applies, and requires no inversions. One expects that the elliptic net algorithm will yield to further optimisation, possibly providing an efficient alternative to Miller's algorithm in many cases. Several groups of researchers are already working in this direction; we discuss this in Chapter 18.

The elliptic net algorithm for the Tate-Lichtenbaum pairing is an example of a paradigm which is probably best attributed to Rachel Shipsey: do arithmetic on elliptic curves via the arithmetic of elliptic nets. Shipsey's work made use of this approach to solve the elliptic curve discrete logarithm problem in certain cases already known to be cryptographically insecure. Her work inspired Kristin Lauter and the author to look more closely at the elliptic curve discrete logarithm problem in the context of elliptic nets. This joint work[4] forms the final chapter of this thesis.

The security of elliptic curve cryptography rests on the assumption that the *elliptic curve discrete logarithm problem* is hard.

Problem 1.5.1 (Elliptic Curve Discrete Logarithm Problem (ECDLP)). *Let E be an elliptic curve over a finite field K. Suppose one is given points $P, Q \in E(K)$ such that $Q \in \langle P \rangle$. Determine k such that*

[4] Performed during an internship at Microsoft Research, Redmond, Washington, September 10, 2007 - December 14, 2007.

$Q = [k]P$.

We define three hard problems in the theory of elliptic divisibility sequences (*EDS Association*, *EDS Residue* and *EDS Discrete Log*), each of which is solvable in sub-exponential time if and only if the elliptic curve discrete logarithm problem is solvable in sub-exponential time.

Problem 1.5.2 (EDS Association). *Let E be an elliptic curve over a finite field K. Suppose one is given points $P, Q \in E(K)$ such that $Q \in \langle P \rangle$, $Q \neq \mathcal{O}$, and $\mathrm{ord}(P) \geq 4$. Determine $W_{E,P}(k)$ for the value of $0 < k < \mathrm{ord}(P)$ such that $Q = [k]P$.*

Problem 1.5.3 (EDS Residue). *Let E be an elliptic curve over a finite field K. Suppose one is given points $P, Q \in E(K)$ such that $Q \in \langle P \rangle$, $Q \neq \mathcal{O}$, and $\mathrm{ord}(P) \geq 4$. Determine the quadratic residuosity of $W_{E,P}(k)$ for the value of $0 < k < \mathrm{ord}(P)$ such that $Q = [k]P$.*

Problem 1.5.4 (Width s EDS Discrete Log). *Given an elliptic divisibility sequence W and terms $W(k)$, $W(k+1)$, ..., $W(k+s-1)$, determine k.*

A perfectly periodic elliptic divisibility sequence is one which has a finite period n and whose first positive index k at which $W(k) = 0$ is $k = n$. If a periodic sequence is not perfectly periodic, then it has $n > k$. Our main result is as follows.

Theorem 1.5.5 (Introductory version of Theorem 19.8.1). *Let E be an elliptic curve over a finite field $K = \mathbb{F}_q$ of characteristic $\neq 2$. If any one of the following problems is solvable in sub-exponential time, then all of them are:*

1. *Problem 1.5.1: ECDLP*
2. *Problem 1.5.2: EDS Association for non-perfectly periodic sequences*
3. *Problem 1.5.3: EDS Residue for non-perfectly periodic sequences*
4. *Problem 1.5.4 ($s = 3$): Width 3 EDS Discrete Log for perfectly periodic sequences*

A secondary purpose of our analysis is to relate these hard problems to the MOV and Frey-Rück attacks, as well as Shipsey's attack (on curves where these apply); this relationship stems from the connection between the Tate-Lichtenbaum pairing and elliptic nets. Finally, this research raises the interesting question of when terms of an elliptic divisibility sequence or elliptic net over a finite field are quadratic residues.

1.6 Prerequisites

We will assume the reader is very familiar with elliptic curves. All of the necessary background concerning elliptic curves can be found in [63, 64]. Among the other topics we will assume familiarity with (listed with references) are: the basic concepts of homological algebra such as fibre products and cohomology [75, Chapters 1-3]; basic complex function theory [44, Chapter 1]; schemes and algebraic

varieties [29, Chapter II], particularly the theory of divisors and line bundles; valuations and number fields [52, Chapters I-II]; and the basic theory of abelian varieties [38, Chapters I-VI] [51, Chapter III], particularly duality and the Theorem of the Cube. In general the citations given include much more than is actually required. Part III, Chapters 12, 13, 14 and 16 provide detailed background on central extensions, $\mathbb{G}_m$-torsors and line bundles, generalised Jacobians, biextensions and the Weil and Tate-Lichtenbaum pairings. Part I, Chapter 2 gives all the necessary background on elliptic divisibility sequences used in the thesis.

Chapter 2

Basic properties of elliptic divisibility sequences

Our purpose in this chapter is to give an overview of the classical theory and methods of elliptic divisibility sequences. As such, we will include especially those proofs that give a flavour of the methods, and omit much of the tedium. Citations are given wherever details are missing.

2.1 Making the curve-sequence relation explicit

Ward, in relating sequences and curves in Theorem 1.2.1, gives explicit formulæ for the coefficients of the Weierstrass equation of the curve and the coordinates of the point, in terms of the initial terms of the sequence. Christine Swart gives a cleaner collection of equations for this, and it is her version we describe here. Also, although Ward concerns himself with integer sequences, his formulae and those of Swart work equally well for rationals. As in the introduction, define a change of variables of a cubic curve in Weierstrass form to be *unihomothetic* if it is of the form

$$\begin{aligned} x' &= x + r, \\ y' &= y + sx + t. \end{aligned}$$

Proposition 2.1.1 ([68, Thm 4.5.3]). *Let $W : \mathbb{Z} \to \mathbb{Q}$ be an elliptic divisibility sequence with $W(1) = 1$ and $W(2)W(3) \neq 0$. Then the family of curve-point pairs (C, P) such that $W = W_{C,P}$ is three dimensional. These are the curve and point*

$$C : y^2 + a_1 xy + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6, \qquad P = (0,0)$$

where

$$\begin{aligned} a_1 &= \frac{W(4) + W(2)^5 - 2W(2)W(3)}{W(2)^2 W(3)} \\ a_2 &= \frac{W(2)W(3)^2 + W(4) + W(2)^5 - W(2)W(3)}{W(2)^3 W(3)} \end{aligned}$$

$$a_3 = W(2), \qquad a_4 = 1, \qquad a_6 = 0$$

or any image of these under a unihomothetic change of coordinates.

Proof. See Section 8.2. □

If we apply a change of variables of the form

$$x \leftarrow u^2 x, \qquad y \leftarrow u^3 y$$

to the curve E defined by

$$y^2 + a_1 xy + a_3 y = x^3 + a_2^2 + a_4 x + a_6 \tag{2.1}$$

and point $P = (x, y) \in E$ to obtain a new curve E' and point P', then the associated elliptic divisibility sequences satisfy

$$W_{E',P'}(n) = u^{n^2-1} W_{E,P}(n). \tag{2.2}$$

This is called by some an *equivalence* of elliptic divisibility sequences. We set our own terminology later.

2.2 Relations to the group law on the elliptic curve

Suppose we define some auxiliary polynomials ϕ_m and ω_m by

$$\phi_m = x\Psi_m^2 - \Psi_{m+1}\Psi_{m-1}, \tag{2.3}$$

$$4y\omega_m = \Psi_{m+2}\Psi_{m-1}^2 - \Psi_{m-2}\Psi_{m+1}^2. \tag{2.4}$$

Then, one can check that on the curve (2.1),

$$[m]P = \left(\frac{\phi_m(P)}{\Psi_m(P)^2}, \frac{\omega_m(P)}{\Psi_m(P)^3} \right). \tag{2.5}$$

In particular, when working over $\mathbb{Q}$, and in the case of an integer sequence, whenever $\phi_m(P)$ and $\Psi_m(P)$ are relatively prime, the denominator of the x-coordinate of $[m]P$ will be exactly $W_{E,P}(m)^2$. The numerators and denominators in (2.5) may involve cancellation. There is no cancellation if $P = (0,0)$, $a_6 = 0$ and $\gcd(a_3, a_4) = 1$ [61, §4.4].[1]

2.3 More on division polynomials

The division polynomials Ψ_n have a special form.

[1] This has led some to remark that the 'correct' definition of elliptic divisibility sequences is by denominators in such a fashion. We will not join that camp.

Proposition 2.3.1 ([63, Ex 3.7] or [74, V.14]). *The division polynomials Ψ_n have a representation as polynomials in x and y with coefficients in $\mathbb{Z}[a_1,a_2,a_3,a_4,a_6]$. In particular, they are of the form*

$$\Psi_n(x,y) = \begin{cases} nx^{\frac{n^2-1}{2}} + \dots & n \text{ odd} \\ y(nx^{\frac{n^2-4}{2}} + \dots) & n \text{ even} \end{cases}.$$

Therefore, their squares Ψ_n^2 are polynomials of degree n^2-1 in the variable x alone, with coefficients in $\mathbb{Z}[a_1,a_2,a_3,a_4,a_6]$, and leading coefficient n^2. The roots of this polynomial Ψ_n^2 are exactly the x-coordinates of all n^2-1 non-zero n-torsion points on the associated elliptic curve.

2.4 Induction properties

Proposition 2.4.1. *Let $W : \mathbb{Z} \to R$ be an elliptic divisibility sequence that is nonzero at the first four terms. Then $W(-z) = -W(z)$ for any $z \in \mathbb{Z}$. In particular $W(0) = 0$. Furthermore, any two elliptic divisibility sequences $W, W' : \mathbb{Z} \to R$ that agree and are non-zero at 1,2,3 and 4, must agree everywhere.*

Proof. Our first step is to show the last statement for positively indexed terms (i.e., all positively indexed terms agree). Two particular instances of the elliptic net equation (3.1) are

$$W(2n)W(2)W(1)^2 = W(n)\left(W(n+2)W(n-1)^2 - W(n-2)W(n+1)^2\right), \tag{2.6}$$

$$W(2n+1)W(1)^3 = W(n+2)W(n)^3 - W(n-1)W(n+1)^3. \tag{2.7}$$

By induction on these equations, every subsequent positive indexed term is determined by $W(1)$, $W(2)$, $W(3)$, $W(4)$.

Now we show the first statement. Assume without loss of generality that $W(1) = 1$ (since we can scale W by a constant). First we show that $W(0) = 0$. For, consider $n = m = 0$: in this case (1.2) states that $W(0)^2 = 0$. Now we consider the statement $-W(z) = W(-z)$. Suppose $W(z+2) \neq 0$. Setting $n = 1, m = z+1$ in (1.2), we obtain $W(z+2)W(-z) = -W(z+2)W(z)$, whence $W(-z) = -W(z)$ since $W(z+2) \neq 0$. We have now shown the symmetry for $z = 0,1,2$, hence $W(-1)$ and $W(-2)$ are nonzero, and so we've shown it for $z = -3,-4$ also. Therefore we've shown it for $z = 0,1,2,3,4$. Thus, by the first part, the sequences $W'(z) = -W(-z)$ and $W(z)$ agree on the first four terms and therefore agree everywhere.

Finally, by the symmetry property just shown, the terms indexed by non-positive integers are also determined uniquely by $W(1)$, $W(2)$, $W(3)$ and $W(4)$. □

Proposition 2.4.2 ([74, Lemma 4.1]). *If W is an elliptic divisibility sequence satisfying $W(1) = 1$ and $W(2)W(3) \neq 0$, and if two consecutive terms vanish, then $W(n) = 0$ for $n \geq 4$.*

Proof. See [74, Lemma 4.1]. □

2.5 The integer case

From Proposition 2.3.1, any rational elliptic divisibility sequence can be made into an integer sequence by an appropriate equivalence of the form (2.2), clearing the denominators.

Proposition 2.5.1 ([74, Thm 4.1]). *Suppose W is an elliptic divisibility sequence satisfying $W(1)=1$, $W(2)W(3)\neq 0$ and $W(2)|W(4)$, and $W(i)\in\mathbb{Z}$ for $i=1,2,3,4$. Then, the sequence is entirely integer and for all $n,m\in\mathbb{Z}$,*

$$n|m \implies W(n)|W(m).$$

Proof. We provide a sketch. For a complete proof, see [74, Thm 4.1]. Recall equations (2.6) and (2.7):

$$W(2n)W(2)W(1)^2 = W(n)\left(W(n+2)W(n-1)^2 - W(n-2)W(n+1)^2\right),$$
$$W(2n+1)W(1)^3 = W(n+2)W(n)^3 - W(n-1)W(n+1)^3.$$

A first induction shows that all terms are integers, and $W(2)|W(2n)$ for every n. Then, a second induction shows the divisibility property in general: for this, we use the following equations (the first in the case that m is even, the second in the case that it is odd):

$$W(nm)W(2) = W\left(\tfrac{nm}{2}\right)\left(W\left(\tfrac{nm}{2}+2\right)W\left(\tfrac{nm}{2}-1\right)^2 - W\left(\tfrac{nm}{2}-2\right)W\left(\tfrac{nm}{2}+1\right)^2\right), \tag{2.8}$$

$$W(nm)W(n) = W\left(\tfrac{n(m+1)}{2}+1\right)W\left(\tfrac{n(m+1)}{2}-1\right)W\left(\tfrac{n(m-1)}{2}\right)^2 \tag{2.9}$$

$$- W\left(\tfrac{n(m-1)}{2}+1\right)W\left(\tfrac{n(m-1)}{2}-1\right)W\left(\tfrac{n(m+1)}{2}\right)^2. \tag{2.10}$$

This second induction uses Proposition 2.4.2. □

2.6 Periodicity modulo p

Definition 2.6.1. For an integer elliptic divisibility sequence W, let r denote the smallest positive integer such that $W(r)\equiv 0 \mod p$. The integer r is called the *rank of apparition of W with respect to p.*

Proposition 2.6.1 ([74, Thm 5.1]). *For any integer elliptic divisibility sequence and prime p, the rank of apparition r with respect to p exists and satisfies*

$$1 \leq r \leq 2p+1.$$

Proof. Without loss of generality, we may assume $r \geq p+3$. Then consider the p values

$$\frac{W(r-1)W(r+1)}{W(r)^2},$$

each of which is a non-zero value modulo p. By the pigeonhole principle[2], two must coincide, and we have for some $1 \leq n < m \leq p-1$,

$$\frac{W(m-1)W(m+1)}{W(m)^2} \equiv \frac{W(n-1)W(n+1)}{W(n)^2} \mod p.$$

Then, the elliptic divisibility sequence recurrence (1.2) implies

$$W(m+n)W(m-n) \equiv 0 \mod p.$$

By our assumption that $r \geq p+3$, and the fact that $m-n \leq p-2$, we conclude that $W(m-n) \not\equiv 0 \mod p$, and so

$$W(m+n) \equiv 0 \mod p.$$

But $m+n \leq 2p+1$. □

By the nice properties of the division polynomials (Proposition 2.3.1), we can reduce them modulo a prime p, and the reduced division polynomials will correspond to the elliptic curve and point reduced modulo the same prime. In particular, it will still be the case that $\Psi_n(P) \equiv 0$ modulo p if and only if $[n]\widetilde{P} = \widetilde{\mathcal{O}}$ on the reduced curve. So, if W is such that $W(1) = 1$, $W(2)W(3) \neq 0$ and $W(2)|W(4)$, then the sequence arises from some curve E and point P (by Theorem 1.2.1). In this case Shipsey [61, §4.7.2] observes that Hasse's bound on the number of points of a curve over a finite field implies that for most primes p, the rank of apparition satisfies the stronger bound

$$r \leq p+1+2\sqrt{p}.$$

Ward proves a very interesting and important 'symmetry' or 'partial periodicity' property.

Theorem 2.6.2 ([74, Thm 9.2]). *Let W be an integer elliptic divisibility sequence such that $W(1) = 1$ and $W(2)|W(4)$. Let p be an odd prime and suppose $W(2)W(3) \not\equiv 0 \mod p$. Let r be the rank of apparition of W with respect to p. Then there exist integers a, b such that for all non-negative integers k and s, we have*

$$W(k+sr) \equiv a^{ks} b^{s^2} W(k) \mod p.$$

Furthermore, the integers a and b satisfy

$$a \equiv \frac{W(r-2)}{W(r-1)W(2)}, \qquad b \equiv \frac{W(r-1)^2 W(2)}{W(r-2)} \mod p.$$

The proof uses the periodicity of the Weierstrass sigma function, and the reader is encouraged to look ahead to Chapter 5, especially equation (5.1).

Proof. By Theorem 1.2.1, W is associated to some curve E and point P. Let z be the complex coordinate of the point P, so that $P = (\wp(z), \wp'(z))$. The roots of $\Psi_n^2(x) = 0$ over $\mathbb{C}$ are of the form

$$\zeta = \wp(\omega/n)$$

[2] My advisor is fond of boosting the confidence of his struggling graduate students by asserting that his own thesis consisted in large part of a single application of pigeonhole principle. For this, and his rumoured – but surely feigned – occasional confusion over the correct definition of a topology, we are ever grateful.

where ω is a period of the Weierstrass $\wp$ function. By Proposition 2.3.1, the polynomial $\Psi_n^2(x)$ has leading coefficient n^2 and coefficients in $\mathbb{Z}[a_1, a_2, a_3, a_4, a_6]$. Hence, $\Psi_n^2(x)$ is a well-defined polynomial of degree n^2 modulo p for any $p \nmid n$.

Now, assume for the moment that $p \nmid r$.

Let K be the number field obtained by adjoining all the roots of $\Psi_r^2(x)$ and let $\mathfrak{p}$ be a prime of K that divides p. Then Ψ_r^2 splits in the finite field $K/\mathfrak{p}$. Since its value at P is zero, $\wp(z)$ is a root modulo $\mathfrak{p}$, i.e.,

$$\wp(z) \equiv \wp(\omega/r) \mod \mathfrak{p}$$

for some period ω. Thus, the sequence W under consideration agrees modulo $\mathfrak{p}$ with the sequence $W'_n = \Psi_n(\omega/r)$. Since W modulo $\mathfrak{p}$ reduces to integers modulo p (i.e., its image is in $\mathbb{Q}/(p) \subset K/\mathfrak{p}$), it suffices to replace W in our consideration with W' and show the formulæ of the theorem modulo $\mathfrak{p}$.

The formula of the Theorem now results from a calculation using the period relation (5.1) of the Weierstrass σ function:

$$\begin{aligned}
\frac{\Psi_{k+sr}\left(\frac{\omega}{r}\right)}{\Psi_k\left(\frac{\omega}{r}\right)} &= \frac{\sigma\left((k+sr)\frac{\omega}{r}\right)}{\sigma\left(k\frac{\omega}{r}\right)}\sigma\left(\frac{\omega}{r}\right)^{-2rsk-r^2s^2} \\
&= \lambda(s\omega)e^{\eta(s\omega)\left(k\frac{s\omega}{r}+s\frac{\omega}{2}\right)}\sigma\left(\frac{\omega}{r}\right)^{-2rsk-r^2s^2} \\
&= \left(\sigma\left(\frac{\omega}{r}\right)^{-2r}e^{\eta(\omega)\frac{\omega}{r}}\right)^{ks}\left(\lambda(\omega)\sigma\left(\frac{\omega}{r}\right)^{-r^2}e^{\eta(\omega)\frac{\omega}{2}}\right)^{s^2}.
\end{aligned}$$

For the case when $p|r$, there are some additional difficulties, and the reader should consult [74, Thm 9.2]. Finally, note that the final statement of the theorem (the formulæ for a and b) follows immediately from the existence of a and b. □

Ayad and Swart generalise partial periodicity to the case of prime power moduli [2, Thm C] [68, Thm 5.1.3]. Their proofs have the additional attraction that they require only the recurrence relation and not the underlying elliptic curve relationship.

Our interest in Ward's original proof is to demonstrate a strategy that we will apply later: first, show that the functions in question (in this case the division polynomials) have a nice form (i.e., they are defined with $\mathbb{Z}$ coefficients and reduce modulo p without becoming trivial); second, verify the property of interest (in this case the periodicity property) in the complex analytic case; third, using the information from step one, transport the property to the finite field (or other field) case.

For a wealth of periodicity properties of elliptic divisibility sequences modulo primes and powers of primes, see Swart [68].

Part II

Moving to higher rank

Chapter 3

Elliptic nets

We now introduce the main character of our story: the elliptic net.

3.1 Definitions and properties

Definition 3.1.1. Let A be a free finitely-generated abelian group, and let R be an integral domain. An *elliptic net* is any map $W : A \to R$ such that the following recurrence holds for all $p, q, r, s \in A$.

$$\begin{aligned} &W(p+q+s)\,W(p-q)\,W(r+s)\,W(r) \\ &\quad + W(q+r+s)\,W(q-r)\,W(p+s)\,W(p) \\ &\quad + W(r+p+s)\,W(r-p)\,W(q+s)\,W(q) = 0 \end{aligned} \tag{3.1}$$

We refer to the rank of A as the *rank* of the elliptic net . Elliptic nets of rank one are elliptic divisibility sequences, since (1.2) is a special case of (3.1). That the converse holds is a consequence of the Curve-Net Theorem (Theorem 9.2.1), but should be susceptible of an elementary proof. Unfortunately the author does not know of one.

Nelson Stephens has pointed out in conversation that another form of the elliptic net recurrence relation is

$$\begin{aligned} &W(a+b)\,W(a-c)\,W(c+d)\,W(c-d) \\ &\quad + W(a+c)\,W(a-c)\,W(d+b)\,W(d-b) \\ &\quad + W(a+d)\,W(a-d)\,W(b+c)\,W(b-c) = 0 \end{aligned} \tag{3.2}$$

where each term corresponds to a cyclic permutation of (bcd). This can be seen by the substitution $s \leftarrow 2a$, $r \leftarrow b-a$, $p \leftarrow c-a$, $q \leftarrow d-a$. Unfortunately, this is slightly less general, since in this case s is necessarily even.

Proposition 3.1.1. *Let $W : A \to R$ be an elliptic net. Then $W(-z) = -W(z)$ for any $z \in A$. In particular $W(0) = 0$.*

Proof. First we show that $W(0) = 0$. Consider $p = q = r = s = 0$: in this case (3.1) states that $3W(0)^4 = 0$. Now we consider the general statement. If $W(-z) = W(z) = 0$, we are done. If not, then without loss of generality, assume $W(z) \neq 0$. Setting $p = q = z, r = s = 0$ in (3.1), we obtain $0 + W(z)^4 + W(z)^3 W(-z) = 0$, whence $W(-z) = -W(z)$. □

Proposition 3.1.2. *Suppose that $B \subset A$ is a subgroup of A. Then the restriction of an elliptic net $W : A \to R$ to B is also an elliptic net.*

We refer to this elliptic net as *the subnet associated to B* and write $W|B$.

3.2 Examples

We begin with some trivial examples, easily verified by a simple calculation.

Example 3.2.1. The map $W : \mathbb{Z}^n \to R$ defined by $W(\mathbf{v}) = 0$ for all $\mathbf{v}$ is an elliptic net.

Example 3.2.2. The map $W : \mathbb{Z}^n \to \mathbb{Z}^n$ defined by $W(\mathbf{v}) = \mathbf{v}$ is an elliptic net. The elliptic net $W' : \mathbb{Z} \to \mathbb{Z}^n$ defined by $W'(n) = W(0, \ldots, 0, n)$ is an elliptic subnet of W (this follows from this example and Example 3.2.1).

Since there is a projection $\mathbb{Z}^n \to \mathbb{Z}$ onto any coordinate, we have also some other examples.

Example 3.2.3. For any $1 \leq i \leq n$, the map $W_i : \mathbb{Z}^n \to \mathbb{Z}$ defined by $W(\mathbf{v}) = v_i$ is an elliptic net.

Since there is a ring homomorphism $\mathbb{Z} \to R$ for any ring R, the above give basic examples for all rings R. If c is a constant, and W is an elliptic net, so is cW.

Now we consider some more interesting examples.

Example 3.2.4 (Due to M. Ward [74]). Consider the elliptic net $W : \mathbb{Z} \to \mathbb{Z}$ given by

$$W(n) = \left(\frac{n}{3}\right) = \begin{cases} 0 & 3|n \\ 1 & n \equiv 1 \mod 3 \\ -1 & n \equiv -1 \mod 3 \end{cases}$$

The proof that this is an elliptic net is as follows. First, the Legendre symbol is multiplicative, so each term of the recurrence relation (3.1) takes value $-1, 0, 1$ according to the modulo 3 residue of the product of the indices of the term. Therefore, we analyse the modulo 3 residue of the three products $(p+q+s)(p-q)(r+s)r$, $(q+r+s)(q-r)(p+s)p$ and $(r+p+s)(r-p)(q+s)q$. By Example 3.2.2, we know that these three products add up to zero. We claim that at least one of them is zero. For, if not, then none of p, q, r, $p-q$, $q-r$ or $r-p$ is divisible by 3. So p, q, r are distinct nonzero elements of $\mathbb{Z}/3\mathbb{Z}$, a contradiction. Therefore at least one product vanishes modulo 3; we claim that the other two products are either both divisible by 3, or else distinct as residues modulo 3. For, if they were the same and non-zero modulo 3, then their sum would not be divisible by 3, a contradiction. Hence, the residues of the three products are either $\{0, 0, 0\}$ or $\{-1, 0, 1\}$ in some permutation. Thus the values of $W(n)$ satisfy the recurrence relation.

There are more interesting examples in the case of $R = \mathbb{C}$.

Example 3.2.5 (Lucas, as described by [74]). Fix some $x \in \mathbb{C}$ and consider the sequence

$$W(n) = \frac{\sin(nx)}{\sin(x)} = \frac{e^{inx} - e^{-inx}}{e^{ix} - e^{-ix}}.$$

It is a tedious computation to verify that W is an elliptic net. For any $a, b \in \mathbb{C}$ satisfying $ab = 1$, there is some $x \in \mathbb{R}$ such that $a = e^{ix}, b = e^{-ix}$. If $a, b, c \in \mathbb{C}$ satisfy $c^2 = ab$, then we define W' to be

$$W'(n) = c^{1-n}\frac{a^n - b^n}{a - b}.$$

It follows that this is an elliptic net also, by considering $a' = a/c, b' = b/c$. This family includes the elliptic net

$$W_F(n) = i^{n-1}F_n,$$

where F_n are the Fibonacci numbers , defined by $F_1 = F_2 = 1, F_n = F_{n-1} + F_{n-2}$ and arising from a, b roots of $x^2 - x - 1$. It also includes the elliptic net

$$W_M(n) = \sqrt{2}^{1-n}(2^n - 1)$$

related to the Mersenne numbers $M_n = 2^n - 1$.

From the equation $x^2 + 3x + 1$, we obtain $a, b = \frac{3 \pm \sqrt{5}}{2}$ and the elliptic net

$$W_{2F}(n) = F_{2n}$$

of the even-indexed Fibonacci numbers. The first few terms of this net are

$$1, 3, 8, 21, 55, 144, 377, 987, 2584, 6765, 17711, 46368, 121393, 317811, 832040, \ldots$$

Proposition 3.2.1. *Fix n, and R an integral domain. If* $\mathbf{v} \in \mathbb{Z}^n$ *is such that* $W(\mathbf{v}) = 0$ *for all elliptic nets* $W : \mathbb{Z}^n \to R$*, then* $\mathbf{v} = 0$.

Proof. I claim that for any m, k, there is an elliptic net $W : \mathbb{Z} \to \mathbb{Z}$ (chosen depending on m, k) such that $W(k) \notin (m)$. First, if $k \notin (m)$, then $W(n) = n$ will do. Next, recall that $\gcd(F_{2n}, n) = 1$ for all $n \neq 5$ [72, p.84]. We also have that $(\frac{5}{3}) = -1$. So it suffices to choose among $W(n) = n$, $W(n) = F_{2n}$ or $W(n) = \left(\frac{n}{3}\right)$ (Examples 3.2.2, 3.2.5, 3.2.4).

Let $\phi : \mathbb{Z} \to R$ be the map given by $r \mapsto r \cdot 1$. Let (m) be the kernel of this map, where $m \in \mathbb{Z}$. Suppose that $\mathbf{v} \neq 0$. Then there is some non-zero coordinate v_i. Choose $W : \mathbb{Z} \to \mathbb{Z}$ as above so that $W(v_i) \notin (m)$. Then $\phi(W(W_i(\mathbf{v}))) \neq 0$ for the elliptic net $\phi \circ W \circ W_i : \mathbb{Z}^n \to R$ (see Example 3.2.3 for definition of W_i). □

The most interesting examples must wait until we have proven much of the general theory. But as a preview, here is an illustration of a portion of an elliptic net of rank 2 that is typical of those studied in this thesis.

Example 3.2.6. In a way to be defined in Chapter 5, the following elliptic net is associated to the elliptic curve $y^2 + y = x^3 + x^2 - 2x$ and the points $P = (0, 0)$, $Q = (1, 0)$ on that curve. The array shows

the first quadrant of the elliptic net, with the origin ($W(0,0) = 0$) in the lower left. For example, $W(3,2) = -13$.

4335	5959	12016	−55287	23921	1587077	−7159461
94	479	919	−2591	13751	68428	424345
−31	53	−33	−350	493	6627	48191
−5	8	−19	−41	−151	989	−1466
1	3	−1	−13	−36	181	−1535
1	1	2	−5	7	89	−149
0	1	1	−3	11	38	249

Chapter 4

The joy of induction

4.1 Proofs by induction

Induction is surely the favourite pastime of any recurrence relation; we will find it convenient to set some notation and terminology.

Definition 4.1.1. Let I be a group, called the *indexing group*, whose elements are called *indices*. The *term associated to i* is the symbol T_i. Equations in a finite number of the T_i using addition, subtraction and multiplication of the terms are called *recurrence relations*.

Fix a set $\mathcal{R}$ of recurrence relations. We say that an index $i \in I$ is *implied by* a set $\mathcal{J} \subset I$ if either $i \in \mathcal{J}$ or some element r of $\mathcal{R}$ satisfies the following conditions:

1. There is exactly one occurence of T_i among the terms of r.
2. The other terms of r are indexed by elements of $\mathcal{J}$.
3. All the terms in the monomial with T_i are indexed by non-zero elements of I.

Let $S \subset I$ be a finite set. We say that i is *S-integrally implied by $\mathcal{J}$* if in addition the monomial containing T_i consists only of T_i and terms indexed by S. A set $K \subset I$ is *(S-integrally) implied by* the set $\mathcal{J}$ if every index in K is (S-integrally) implied by $\mathcal{J}$.

A set $B \subset I$ *is an (S-integral) baseset for T* if the following condition holds: For each index $i \in I$, there is a finite sequence $\mathcal{J}_0 \subset \mathcal{J}_1 \subset \cdots \subset \mathcal{J}_n$ such that $B = \mathcal{J}_0$, $i \in \mathcal{J}_n$ and for each $1 \leq k \leq n$, $\mathcal{J}_k$ is (S-integrally) implied by $\mathcal{J}_{k-1}$.

In our case, the indexing set will be the group $A \cong \mathbb{Z}^n$ for some n. The set $\mathcal{R}$ will be all instances of the recurrence relation

$$T_{\mathbf{p}+\mathbf{q}+\mathbf{s}} T_{\mathbf{p}-\mathbf{q}} T_{\mathbf{r}+\mathbf{s}} T_{\mathbf{r}} + T_{\mathbf{q}+\mathbf{r}+\mathbf{s}} T_{\mathbf{q}-\mathbf{r}} T_{\mathbf{p}+\mathbf{s}} T_{\mathbf{p}} + T_{\mathbf{r}+\mathbf{p}+\mathbf{s}} T_{\mathbf{r}-\mathbf{p}} T_{\mathbf{q}+\mathbf{s}} T_{\mathbf{q}} = 0$$

(which is just (3.1)) as $\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s}$ range over A.

In the induction proofs of the next section, the game is to show that all elements are implied or S-integrally implied by a baseset. To do this, we will construct the sets $\mathcal{J}_1, \mathcal{J}_2, \ldots$ by induction. At each stage, we show that an index is implied simply by stating which is the relevant element of $\mathcal{R}$.

In generating terms from a baseset using induction in this way, one never obtains a zero term as a polynomial in the baseset terms. For, if we did, this term would be zero in all elliptic nets, a fact contradicting Proposition 3.2.1. Therefore, condition 3 above really is sufficient to avoid division by zero in the context of these abstract proofs.

The set $\mathcal{R}$ is an image of a homomorphism of $\mathbb{Z}$-modules $F : A^4 \to A^{12}$, and as such forms a $\mathbb{Z}$-module itself. That is to say, the recurrence relation (3.1) is given by the 12 indices in A whose terms are related. This will be written as

$$F(\mathbf{p},\mathbf{q},\mathbf{r},\mathbf{s}) = [\ \mathbf{p}+\mathbf{q}+\mathbf{s},\mathbf{p}-\mathbf{q},\mathbf{r}+\mathbf{s},\mathbf{r}\ \mid\ \mathbf{q}+\mathbf{r}+\mathbf{s},\mathbf{q}-\mathbf{r},\mathbf{p}+\mathbf{s},\mathbf{p}\ \mid\ \mathbf{r}+\mathbf{p}+\mathbf{s},\mathbf{r}-\mathbf{p},\mathbf{q}+\mathbf{s},\mathbf{q}\ ].$$

In the case of column vector notation, for example if

$$\mathbf{p} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \mathbf{q} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \mathbf{r} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \mathbf{s} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix},$$

then we can write a recurrence compactly as an array, such as

$$\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array} \left[\begin{array}{cccc|cccc|cccc} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 \end{array}\right]. \tag{4.1}$$

In this notation, the terms to the left of the square braces correspond to the columns of p, q, r and s, while the indices of the terms of the recurrence appear as the columns within the square braces.

To demonstrate that an index i is (S-integrally) implied by a set of indices S, it suffices to write down an appropriate such array. We remark that any array of the form (4.1) is a recurrence if each row is a recurrence. Therefore we may construct examples row-by-row.

One final note. By Proposition 3.1.1, when an index i is implied, the index $-i$ is also taken to be immediately implied. This will often be implicit.

The following definition will sometimes be used to order the inductions.

Definition 4.1.2. Let

$$N(\mathbf{v}) = \max_{i=1,\ldots,n} |v_i|$$

be the *norm* of the vector $\mathbf{v}$ or the term $W(\mathbf{v})$.

4.2 Basesets for ranks 1 and 2

The rank one case is a result of Morgan Ward.

Theorem 4.2.1 (Ward [74, Thm 4.1]). *Let $W : \mathbb{Z} \to R$ be an elliptic net. Then all $W(v)$ are polynomials in*

$$\left\{ W(1), W(1)^{-1}, W(2), W(3), \frac{W(4)}{W(2)} \right\}$$

with $\mathbb{Z}$-coefficients.

In particular, if $W(1) = 1$, the $W(i)$ are integers for $i = 2,3,4$ and $W(2)$ divides $W(4)$, then the elliptic net consists entirely of integers.

For the rank two case, we require a lemma.

Lemma 4.2.2. *Let $W : \mathbb{Z}^2 \to R$ be an elliptic net. Then all $W(\mathbf{z})$ are polynomials with integer coefficients in*

$$B = \{W(\mathbf{v}) : N(\mathbf{v}) \leq 4\} \cup \{W(1,0)^{-1}, W(0,1)^{-1}, W(1,1)^{-1}\}.$$

Proof. Let

$$S = \{(1,0), (0,1), (1,1)\}.$$

The proof proceeds by induction on the norm. Clearly any $W(\mathbf{v})$ for $\mathbf{v} \in B$ is a polynomial with integer coefficients in terms of B. Now suppose that all terms with indices of norm less than $N_0 \geq 4$ are such polynomials. Call the set of such indices T. Suppose $\mathbf{v}$ is an index of norm N_0. We will construct the recurrence demonstrating that $\mathbf{v}$ is S-integrally implied by T.

We construct examples row-by-row. For each $i = 1,2$, let w_i be defined by

$$w_i = \begin{cases} v_i/2, & v_i \text{ even} \\ (v_i+1)/2, & v_i \text{ odd} \end{cases}.$$

Case I: $\mathbf{v}$ has exactly one odd entry and one even entry. For the odd entry, we use the row

$$F(w_i, w_i - 1, 0, 0) = \left[\; v_i, 1, 0, 0 \;\middle|\; w_i - 1, w_i - 1, w_i, w_i \;\middle|\; w_i, -w_i, w_i - 1, w_i - 1 \; \right].$$

For the even entry, we use the row

$$F(w_i, w_i, 1, 0) = \left[\; v_i, 0, 1, 1 \;\middle|\; w_i + 1, w_i - 1, w_i, w_i \;\middle|\; w_i + 1, -w_i + 1, w_i, w_i \; \right].$$

Case II: $\mathbf{v}$ has exactly two odd entries. For the first odd entry, use

$$F(w_i, w_i - 1, 0, 0) = \left[\; v_i, 1, 0, 0 \;\middle|\; w_i - 1, w_i - 1, w_i, w_i \;\middle|\; w_i, -w_i, w_i - 1, w_i - 1 \; \right].$$

For the second odd entry, use

$$F(w_i, w_i - 1, 1, 0) = \left[\; v_i, 1, 1, 1 \;\middle|\; w_i, w_i - 2, w_i, w_i \;\middle|\; w_i + 1, -w_i + 1, w_i - 1, w_i - 1 \; \right].$$

Case III: $\mathbf{v}$ has two even entries. For the first even entry, use

$$F(w_i, w_i - 1, 0, 1) = \left[\; v_i, 1, 1, 0 \;\middle|\; w_i, w_i - 1, w_i + 1, w_i \;\middle|\; w_i + 1, -w_i, w_i, w_i - 1 \; \right].$$

For the second even entry, use

$$F(w_i, w_i, 1, 0) = \left[\; v_i, 0, 1, 1 \;\middle|\; w_i + 1, w_i - 1, w_i, w_i \;\middle|\; w_i + 1, -w_i + 1, w_i, w_i \; \right].$$

For even v_i, either $|v_i| \leq 2$ or $|v_i| > 3$. In the former case, $|w_i| + 1 \leq 2 < N_0$. In the latter case, we have $|w_i| + 1 \leq (|v_i| + 2)/2 < |v_i| \leq N_0$. For odd $\mathbf{v}_i$, either $|v_i| \leq 3$ or $|v_i| > 4$. In the former case $|w_i| + 2 \leq 4 \leq N_0$. In the latter case, we have $|w_i| + 2 \leq (|v_i| + 5)/2 < |v_i| \leq N_0$.

Therefore all the vectors in the recurrence have norm less than N_0 with the exception of $\mathbf{v}$. In the monomial of $\mathbf{v}$ in the recurrence, the other indices are $(1,0)$, $(0,1)$ or $(1,1)$. This demonstrates that $\mathbf{v}$ is integrally implied by the set of indices of norm strictly less than N_0, and we are done. □

Theorem 4.2.3. *Let $W : \mathbb{Z}^2 \to R$ be an elliptic net. Then all $W(\mathbf{v})$ are polynomials in*

$$\left\{ W(1,1), W(1,0), W(0,1), W(1,1)^{-1}, W(1,0)^{-1}, W(0,1)^{-1}, W(2,1), W(1,2), \right.$$
$$\left. W(2,0), W(0,2), \frac{W(0,2)W(2,1)W(1,0) - W(0,1)W(2,0)W(1,2)}{W(0,1)^3 W(2,1) - W(1,0)^3 W(1,2)} \right\}$$

with $\mathbb{Z}$-coefficients.

In particular, if $W(1,0) = W(0,1) = W(1,1) = 1$, the terms $W(2,0)$, $W(0,2)$, $W(1,2)$, $W(2,1)$ are integers and $W(2,1) - W(1,2)$ divides $W(0,2)W(2,1) - W(2,0)W(1,2)$, then all terms of the elliptic net are integers.

Proof. We have the recurrence

$$\begin{array}{cccc|cccc|cccc|cccc} 0 & 1 & 1 & 0 & 1 & -1 & 1 & 1 & 2 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 \end{array}.$$

And so

$$W(1,-1) = \frac{W(0,1)^3 W(2,1) - W(1,0)^3 W(1,2)}{W(1,1)^3}. \tag{4.2}$$

Let

$$S = \{(1,0), (0,1), (1,1), (1,-1)\}.$$

Let

$$B = \{\mathbf{v} \in \mathbb{Z}^2 : N(\mathbf{v}) \leq 4\}.$$

By Lemma 4.2.2, it is only required to verify the truth of the statement for $\mathbf{v}$ of norm less than or equal to 4. To do so, we demonstrate term-by-term that B is S-integrally implied by the set

$$\{(1,0), (0,1), (1,1), (2,0), (0,2), (2,1), (1,2)\}.$$

We list the relevant recurrences in order. It is assumed at each step that the calculation of $W(a,b)$ also calculated $W(-a,-b) = -W(a,b)$. We prefix each recurrence by the element whose index it is meant to imply.

$$W(2,-1): \quad \begin{array}{cccc|cccc|cccc|cccc} -1 & 0 & 1 & 1 & 0 & -1 & 2 & 1 & 2 & -1 & 0 & -1 & 1 & 2 & 1 & 0 \\ 1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 \end{array}.$$

$$W(1,-2): \quad \begin{array}{cccc|cccc|cccc|cccc} 0 & 1 & 1 & 0 & 1 & -1 & 1 & 1 & 2 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ -1 & -1 & 0 & 0 & -2 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & 1 & -1 & -1 \end{array}.$$

$$W(2,2):\ \begin{matrix}1&1&-1&0\\1&2&1&-1\end{matrix}\left[\begin{matrix}2&0&-1&-1\\2&-1&0&1\end{matrix}\middle|\begin{matrix}0&2&1&1\\2&1&0&1\end{matrix}\middle|\begin{matrix}0&-2&1&1\\1&0&1&2\end{matrix}\right]. \tag{4.3}$$

Note that in (4.3), the calculation of $W(2,2)$ requires division by $W(1,-1)$.

$$W(2,-2):\ \begin{matrix}1&1&-1&0\\-1&-2&-1&1\end{matrix}\left[\begin{matrix}2&0&-1&-1\\-2&1&0&-1\end{matrix}\middle|\begin{matrix}0&2&1&1\\-2&-1&0&-1\end{matrix}\middle|\begin{matrix}0&-2&1&1\\-1&0&-1&-2\end{matrix}\right].$$

At this point we have implied all indices of norm at most 2.

$$W(3,0):\ \begin{matrix}2&1&0&0\\0&0&1&0\end{matrix}\left[\begin{matrix}3&1&0&0\\0&0&1&1\end{matrix}\middle|\begin{matrix}1&1&2&2\\1&-1&0&0\end{matrix}\middle|\begin{matrix}2&-2&1&1\\1&1&0&0\end{matrix}\right].$$

$$W(3,1):\ \begin{matrix}2&1&0&0\\1&0&1&0\end{matrix}\left[\begin{matrix}3&1&0&0\\1&1&1&1\end{matrix}\middle|\begin{matrix}1&1&2&2\\1&-1&1&1\end{matrix}\middle|\begin{matrix}2&-2&1&1\\2&0&0&0\end{matrix}\right].$$

$$W(3,2):\ \begin{matrix}2&1&0&0\\1&1&1&0\end{matrix}\left[\begin{matrix}3&1&0&0\\2&0&1&1\end{matrix}\middle|\begin{matrix}1&1&2&2\\2&0&1&1\end{matrix}\middle|\begin{matrix}2&-2&1&1\\2&0&1&1\end{matrix}\right].$$

$$W(3,3):\ \begin{matrix}2&1&1&0\\2&1&0&0\end{matrix}\left[\begin{matrix}3&1&1&1\\3&1&0&0\end{matrix}\middle|\begin{matrix}2&0&2&2\\1&1&2&2\end{matrix}\middle|\begin{matrix}3&-1&1&1\\2&-2&1&1\end{matrix}\right].$$

Simply by switching top rows with bottom rows, we similarly calculate $W(0,3)$, $W(1,3)$, and $W(2,3)$. And by putting negatives on the second rows, we imply the indices $(1,-3)$, $(2,-3)$, $(3,-3)$, $(3,-2)$ and $(3,-1)$. Note that some of these calculations appear to require division by $W(1,-1)$. However, in each case that this occurs, the other two terms are divisible either by $W(1,-1)$ or $W(2,-2)$. But $W(1,-1)$ divides $W(2,-2)$ by (4.3). We have now implied all indices with norm at most 3.

$$W(4,0):\ \begin{matrix}2&1&0&1\\0&0&1&0\end{matrix}\left[\begin{matrix}4&1&1&0\\0&0&1&1\end{matrix}\middle|\begin{matrix}2&1&3&2\\1&-1&0&0\end{matrix}\middle|\begin{matrix}3&-2&2&1\\1&1&0&0\end{matrix}\right].$$

$$W(4,1):\ \begin{matrix}3&2&1&-1\\0&0&0&1\end{matrix}\left[\begin{matrix}4&1&0&1\\1&0&1&0\end{matrix}\middle|\begin{matrix}2&1&2&3\\1&0&1&0\end{matrix}\middle|\begin{matrix}3&-2&1&1\\1&0&1&0\end{matrix}\right].$$

$$W(4,2):\ \begin{matrix}3&2&1&-1\\1&1&1&0\end{matrix}\left[\begin{matrix}4&1&0&1\\2&0&1&1\end{matrix}\middle|\begin{matrix}2&1&2&3\\2&0&1&1\end{matrix}\middle|\begin{matrix}3&-2&1&2\\2&0&1&1\end{matrix}\right].$$

$$W(4,3):\ \begin{matrix}2&2&1&0\\2&1&0&0\end{matrix}\left[\begin{matrix}4&0&1&1\\3&1&0&0\end{matrix}\middle|\begin{matrix}3&1&2&2\\1&1&2&2\end{matrix}\middle|\begin{matrix}3&-1&2&2\\2&-2&1&1\end{matrix}\right].$$

$$W(4,4):\ \begin{matrix}3&2&1&-1\\2&2&1&0\end{matrix}\left[\begin{matrix}4&1&0&1\\4&0&1&1\end{matrix}\middle|\begin{matrix}2&1&2&3\\3&1&2&2\end{matrix}\middle|\begin{matrix}3&-2&1&2\\3&-1&2&2\end{matrix}\right].$$

Again by switching top rows with bottom rows, we similarly calculate $W(0,4)$, $W(1,4)$, $W(2,4)$ and $W(3,4)$. And by putting negatives on the second rows, we imply the indices $(1,-4)$, $(2,-4)$, $(3,-4)$, $(4,-4)$, $(4,-3)$, $(4,-2)$ and $(4,-1)$. As before, the resulting division by $W(1,-1)$ can be accounted for. We have demonstrated the calculation of all terms of index with norm at most 4. The calculations, with attention to the case of (4.3), demonstrate the first statement. The second statement follows immediately. □

4.3 Basesets for ranks $n \geq 3$

We now extend the results of the last section to arbitrary dimension.

Lemma 4.3.1. *Let $n \geq 3$. Suppose that $W : \mathbb{Z}^n \to R$ is an elliptic net. Let*

$$B_n = \{\mathbf{v} \in \mathbb{Z}^n : v_n = 0\} \cup \{\mathbf{v} \in \mathbb{Z}^n : N(\mathbf{v}) \leq 1\}.$$

Let

$$S_n = \{\mathbf{v} \in \mathbb{Z}^n : N(\mathbf{v}) = 1\}.$$

Then B_n is an S_n-integral baseset for W.

Proof. We prove this by induction on the norm. The base case is trivial: any $\mathbf{v} \in \mathbb{Z}^n$ with $N(\mathbf{v}) \leq 1$ is in B_n. We will show that any $\mathbf{v} \in \mathbb{Z}^n$ with $N(\mathbf{v}) = N_0$ is S_n-integrally implied by the set

$$K_{N_0} = \{\mathbf{v} \in \mathbb{Z}^n : N(\mathbf{v}) < N_0\}.$$

First, consider such a $\mathbf{v}$ with $v_n = 1$. We will construct a recurrence row-by-row. Note that the following is a recurrence for $\mathbb{Z}$:

$$F(1,0,0,0) = [\ 1,1,0,0\ \mid\ 0,0,1,1\ \mid\ 1,-1,0,0\ ].$$

Using this as the n-th row, it is now only necessary to create for each $1 \leq i \leq n-1$ a recurrence of the form

$$[\ v_i, \$, \%, \%\ \mid\ *, *, \#, \#\ \mid\ \#, \#, *, *\ ],$$

such that

1. The # must have absolute value less than N_0,
2. The % and \$ must have absolute value less than or equal to 1,
3. The $*$ and % must not be all zero in a given column.

Let

$$w_i = \begin{cases} v_i/2, & v_i \text{ even} \\ (v_i+1)/2, & v_i \text{ odd} \end{cases}.$$

Case I: If the i-th entry of $\mathbf{v}$ is even and $w_i \geq 0$, we may use either

$$F(w_i - 1, w_i, 0, 1) = [\ v_i, -1, 1, 0\ \mid\ w_i + 1, w_i, w_i, w_i - 1\ \mid\ w_i, -w_i + 1, w_i + 1, w_i\ ],$$

or

$$F(w_i, w_i + 1, 1, -1) = [\ v_i, -1, 0, 1\ \mid\ w_i + 1, w_i, w_i - 1, w_i\ \mid\ w_i, -w_i + 1, w_i, w_i + 1\ ].$$

Case II: If the i-th entry of $\mathbf{v}$ is even and $w_i < 0$, we may use either

$$F(w_i, w_i - 1, 0, 1) = [\ v_i, 1, 1, 0\ \mid\ w_i, w_i - 1, w_i + 1, w_i\ \mid\ w_i + 1, -w_i, w_i, w_i - 1\ ],$$

or

$$F(w_i+1, w_i, 1, -1) = [\ \ v_i, 1, 0, 1\ \ |\ \ w_i, w_i-1, w_i, w_i+1\ \ |\ \ w_i+1, -w_i, w_i-1, w_i\ \ ].$$

Case III: If the i-th entry of $\mathbf{v}$ is odd, we may use either

$$F(w_i, w_i, 0, -1) = [\ \ v_i, 0, -1, 0\ \ |\ \ w_i-1, w_i, w_i-1, w_i\ \ |\ \ w_i-1, -w_i, w_i-1, w_i\ \ ],$$

or

$$F(w_i, w_i, 1, -1) = [\ \ v_i, 0, 0, 1\ \ |\ \ w_i, w_i-1, w_i-1, w_i\ \ |\ \ w_i, -w_i+1, w_i-1, w_i\ \ ].$$

Condition (1) is clearly satisfied.

Now we verify condition (2). In case I, if $w_i = 0$, then the largest #-entry is of size at most $1 < N_0$. Still for case I, if $w_i > 0$, then the #-entry of largest absolute value is $|w_i| < |v_i| \leq N_0$. In case II, $w_i < 0$ and the #-entry of largest absolute value is $-w_i$ and $|-w_i| < |v_i| \leq N_0$. In case III, if $w_i > 0$, then $v_i > 0$, and the largest #-entry is of absolute value $|w_i| = |(v_i+1)/2| \leq |v_i| \leq N_0$. If $v_i = 1$, then the second $\leq$ sign is actually a strict $<$. If $v_i > 1$, then the first $\leq$ sign is actually a strict $<$. In case III, if $w_i = 0$, then the largest #-entry is of size at most $1 < N_0$. In case III, if $w_i < 0$, then the #-entry of largest size is $w_i - 1$. Then $|v_i| \geq 3$. So $|w_i - 1| = |(v_i - 1)/2| < |v_i| \leq N_0$.

It remains to check condition (3). If $n \geq 3$, by the choices given above, we may guarantee that the first two $*$-columns satisfy this. For the other columns, the only cases of difficulty are in case I when $w_i = 0$ and hence $v_i = 0$, or in case III when $w_i = 0$ or 1 and hence $v_i = -1$ or 1. But if $v_i \in \{-1, 0, 1\}$ for all i, then we are trying to create a recurrence for an element of the baseset, which is not the case.

This demonstrates how to imply any index $\mathbf{v}$ with norm N_0 and $v_n = 1$. By Proposition 3.1.1 we also have all indices with $v_n = -1$. We will now show how to imply any index $\mathbf{v}$ of norm N_0 by induction on the size of the last term. Suppose that all indices with $|v_n| < M_0$ have been implied for some $M_0 \geq 1$.

We now show how to imply an index $\mathbf{v}$ with $v_n = M_0$ (and hence by Proposition 3.1.1 with $v_n = -M_0$). If M_0 is even, we can use as the n-th row the recurrence

$$F(w_i, w_i, 0, 0) = [\ \ v_i, 0, 0, 0\ \ |\ \ w_i, w_i, w_i, w_i\ \ |\ \ w_i, -w_i, w_i, w_i\ \ ].$$

If M_0 is odd, we can use

$$F(w_i, w_i-1, 0, 0) = [\ \ v_i, 1, 0, 0\ \ |\ \ w_i-1, w_i-1, w_i, w_i\ \ |\ \ w_i, -w_i, w_i-1, w_i-1\ \ ].$$

Note that $v_n \geq 2$ so the last eight indices of these recurrences are non-zero. It now suffices to create for each $1 \leq i \leq n-1$ a recurrence of the form

$$[\ \ v_i, \#, \#, \#\ \ |\ \ *, *, *, *\ \ |\ \ *, *, *, *\ \ ],$$

such that the $*$ may be anything at all, and the # may be anything of norm less than or equal to 1 so long as in a given column they are not all zero. Recall that $n \geq 3$. If we have at least one even v_i for $i < n$ or if v_n is odd, then the recurrences given in the first part will suffice for these criteria. If the v_i are all odd for $i < n$ and v_n is even, then it suffices to have the additional possibility

$$F(w_i, w_i-1, 1, 0) = [\ \ v_i, 1, 1, 1\ \ |\ \ w_i, w_i-2, w_i, w_i\ \ |\ \ w_i+1, -w_i+1, w_i-1, w_i-1\ \ ].$$

□

Lemma 4.3.2. *Let $W : \mathbb{Z}^3 \to R$ be an elliptic net. Let*

$$S_3 = \{\mathbf{v} \in \mathbb{Z}^3 : N(\mathbf{v}) = 1\}.$$

Then S_3 is an S_3-integral baseset for W.

Proof. We demonstrate that the terms $W(2,0,0)$, $W(0,2,0)$, $W(2,1,0)$, $W(1,2,0)$ are S_3-integrally implied by the set S_3. To do so, we simply list the relevant recurrences. For $W(2,0,0)$, one may use $\mathbf{p} = (1,0,0)$, $\mathbf{q} = (1,0,-1)$, $\mathbf{r} = (0,1,0)$, $\mathbf{s} = (0,0,1)$ and we obtain

$$\left[\begin{array}{cccc|cccc|cccc} 2 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & -1 & 1 & 0 & 1 & 0 & 0 & -1 \end{array}\right]. \tag{4.4}$$

For $W(2,1,0)$, we use $\mathbf{p} = (1,0,1)$, $\mathbf{q} = (1,0,0)$, $\mathbf{r} = (0,0,1)$, $\mathbf{s} = (0,1,-1)$ and obtain

$$\left[\begin{array}{cccc|cccc|cccc} 2 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & -1 & 0 & 1 & 1 & 0 & -1 & 0 \end{array}\right]. \tag{4.5}$$

For $W(0,2,0)$ and $W(1,2,0)$ we may interchange the rows appropriately.

Now recall from the proof of Theorem 4.2.3, that

$$W(1,-1) = \frac{W(0,1)^3 W(2,1) - W(1,0)^3 W(1,2)}{W(1,1)^3}.$$

Therefore all the terms in the set in the statement of Theorem 4.2.3 are S_3-integrally implied by S_3. Therefore the theorem states that all terms of the form $W(*,*,0)$ are S_3-integrally implied by S_3. Then by Lemma 4.3.1, all of $\mathbb{Z}^3$ is S_3-integrally implied by S_3. □

Theorem 4.3.3. *Let $n \geq 3$. Let $W : \mathbb{Z}^n \to R$ be an elliptic net. Let*

$$S_n = \{\mathbf{v} \in \mathbb{Z}^n : N(\mathbf{v}) = 1\}.$$

Then S_n is an S_n-integral baseset for W. That is, all terms of W are Laurent polynomials with coefficients in $\mathbb{Z}$ in the terms indexed by S_n.

Define

$$S'_n = \{\mathbf{v} \in \mathbb{Z}^n : N(\mathbf{v}) = 1 \text{ and } v_i = 0 \text{ for at least one } i\}.$$

If $n \geq 4$, then S_n is S'_n-integrally implied by S'_n.

Proof. The case $n = 3$ is Lemma 4.3.2. To prove the general case we can induct using Lemma 4.3.1. Define the set

$$R_{k-1} = \{\mathbf{v} \in \mathbb{Z}^k : v_k = 0, N(\mathbf{v}) = 1\} \subset S_k,$$

and from the statement of Lemma 4.3.1,

$$B_k = \{\mathbf{v} \in \mathbb{Z}^k : v_k = 0\} \cup \{\mathbf{v} \in \mathbb{Z}^k : N(\mathbf{v}) \leq 1\}.$$

If the theorem statement holds for $n = k-1$, then all terms of $W : \mathbb{Z}^k \to R$ indexed by $\mathbf{v}$ with $v_k = 0$ are R_{k-1}-integrally implied by R_{k-1}. So all of B_k is S_k-integrally implied by S_k. By Lemma 4.3.1, all of $\mathbb{Z}^k$ is S_k-integrally implied by B_k, and hence S_k-integrally implied by S_k.

Now we show the second statement. Let $n \geq 4$. Note that we have the following recurrences:

$$\begin{aligned}
F(1,0,0,0) &= [\ 1,1,0,0 \mid 0,0,1,1 \mid 1,-1,0,0\ ],\\
F(0,0,0,1) &= [\ 1,0,1,0 \mid 1,0,1,0 \mid 1,0,1,0\ ],\\
F(1,1,1,-1) &= [\ 1,0,0,1 \mid 1,0,0,1 \mid 1,0,0,1\ ],\\
F(0,0,-1,1) &= [\ 1,0,0,-1 \mid 0,1,1,0 \mid 0,-1,1,0\ ].
\end{aligned}$$

By choosing either these or their corresponding negatives, we can S'_n-integrally imply any term of $S_n \backslash S'_n$ from baseset S'_n. □

4.4 Laurentness

While our primary interest in Theorems 4.2.1, 4.2.3 and 4.3.3 is their use in the proof of Theorem 9.2.1, the Laurentness properties we've shown may be of independent interest. We collect them into a single statement here.

Theorem 4.4.1. *The terms of an elliptic net are generated by the recurrence relation from a finite set of initial terms. Furthermore, the terms are Laurent polynomials in a finite set of initial terms. Let*

$$S_n = \{\mathbf{v} \in \mathbb{Z}^n : \max_{i=1,\ldots,n} |v_i| = 1\},$$

$$S'_n = S_n \cap \{\mathbf{v} \in \mathbb{Z}^n : v_i = 0 \text{ for at least one } i\}.$$

The terms of an elliptic net of rank n are Laurent polynomials in the following variables and coefficients:

1. *For $n = 1$:*

$$\text{Variables: } W(1), W(2); \quad \text{Coefficients: } \mathbb{Z}[W(3), W(4)]$$

2. *For $n = 2$:*

$$\begin{aligned}
&\text{Variables: } W(1,1), W(1,0), W(0,1), W(0,1)^3 W(2,1) - W(1,0)^3 W(1,2);\\
&\text{Coefficients: } \mathbb{Z}[W(1,0), W(0,1), W(1,1), W(2,1), W(1,2), W(2,0), W(0,2)]
\end{aligned}$$

3. *For $n = 3$:*

$$\text{Variables: } S_3; \quad \text{Coefficients: } \mathbb{Z}$$

4. *For $n \geq 4$:*

$$\text{Variables: } S'_n; \quad \text{Coefficients: } \mathbb{Z}[S_n \backslash S'_n]$$

The results for $n = 1$ and $n = 2$ are sharp in the sense that the result does not hold for a strictly smaller set of variables or coefficients.

Chapter 5

Elliptic nets over the complex numbers

Fix an elliptic curve E over a field K. Our purpose now is to define functions $\Omega_{\mathbf{v}} : E^n \to K$ for all $\mathbf{v} \in \mathbb{Z}^n$. We wish to do so in such a way that the map $W_{E,\mathbf{P}} : \mathbb{Z}^n \to K$ given by fixing $\mathbf{P} \in E^n$ and defining

$$W_{E,\mathbf{P}}(\mathbf{v}) = \Omega_{\mathbf{v}}(\mathbf{P})$$

is an elliptic net. Our strategy is to do so first for elliptic curves over the complex numbers. Accordingly, we set $K = \mathbb{C}$ for this section and consider the complex uniformization of E. We associate to E a lattice $\Lambda \subset \mathbb{C}$ and consider points $z \in \mathbb{C}/\Lambda$ as points on E.

5.1 Elliptic functions over $\mathbb{C}$

For a complex lattice Λ, let $\eta : \Lambda \to \mathbb{C}$ be the quasi-period homomorphism, and define $\lambda : \Lambda \to \{\pm 1\}$ by

$$\lambda(\omega) = \begin{cases} 1 & \text{if } \omega \in 2\Lambda, \\ -1 & \text{if } \omega \notin 2\Lambda. \end{cases}$$

Recall that the Weierstrass sigma function $\sigma : \mathbb{C}/\Lambda \to \mathbb{C}$ satisfies the following transformation formula for all $z \in \mathbb{C}$ and $\omega \in \Lambda$:

$$\sigma(z+\omega;\Lambda) = \lambda(\omega) e^{\eta(\omega)(z+\frac{1}{2}\omega)} \sigma(z;\Lambda) \tag{5.1}$$

Definition 5.1.1. Fix a lattice $\Lambda \in \mathbb{C}$ corresponding to an elliptic curve E. For $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{Z}^n$, define a function $\Omega_{\mathbf{v}}$ on $\mathbb{C}^n$ in variables $\mathbf{z} = (z_1, \ldots, z_n)$ as follows:

$$\Omega_{\mathbf{v}}(\mathbf{z};\Lambda) = \frac{\sigma(v_1 z_1 + \ldots + v_n z_n;\Lambda)}{\prod_{i=1}^{n} \sigma(z_i;\Lambda)^{2v_i^2 - \Sigma_{j=1}^{n} v_i v_j} \prod_{\substack{1 \le i,j \le n \\ i \ne j}} \sigma(z_i + z_j;\Lambda)^{v_i v_j}} \tag{5.2}$$

(If $\mathbf{v} = \mathbf{0}$, we set $\Omega_{\mathbf{v}} \equiv 0$.) In particular, we have for each $v \in \mathbb{Z}$, a function Ω_v on $\mathbb{C}$ in the variable z:

$$\Omega_v(z;\Lambda) = \frac{\sigma(vz;\Lambda)}{\sigma(z;\Lambda)^{v^2}} \tag{5.3}$$

and for each pair $(u,v) \in \mathbb{Z} \times \mathbb{Z}$, a function $\Omega_{u,v}$ on $\mathbb{C} \times \mathbb{C}$ in variables z and w:

$$\Omega_{u,v}(z,w;\Lambda) = \frac{\sigma(uz+vw;\Lambda)}{\sigma(z;\Lambda)^{u^2-uv}\sigma(z+w;\Lambda)^{uv}\sigma(w;\Lambda)^{v^2-uv}}. \tag{5.4}$$

Proposition 5.1.1. *The functions $\Omega_{\mathbf{v}}$ are elliptic functions in each variable.*

Proof. Let $\omega \in \Lambda$. We show the function is elliptic in the first variable. Let $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{Z}^n$ and $\mathbf{z} = (z_1, \ldots, z_n), \mathbf{w} = (\omega, 0, \ldots, 0) \in \mathbb{C}^n$. Using (5.1), we calculate

$$F = \frac{\Omega_{\mathbf{v}}(\mathbf{z}+\mathbf{w};\Lambda)}{\Omega_{\mathbf{v}}(\mathbf{z};\Lambda)} = \frac{\lambda(v_1\omega)}{\lambda(\omega)^{v_1^2}} = \lambda(\omega)^{v_1 - v_1^2} = 1$$

Thus $\Omega_{\mathbf{v}}$ is invariant under adding a period to the variable z_1. Similarly $\Omega_{\mathbf{v}}$ is elliptic in each variable on $(\mathbb{C}/\Lambda)^n$. □

Proposition 5.1.2. *Fix a lattice $\Lambda \in \mathbb{C}$. Let $\mathbf{v} \in \mathbb{Z}^m$ and $\mathbf{z} \in \mathbb{C}^n$. Let T be an $n \times m$ matrix with entries in $\mathbb{Z}$ and transpose T^{tr}. Then*

$$\Omega_{\mathbf{v}}(T^{tr}(\mathbf{z});\Lambda) = \frac{\Omega_{T(\mathbf{v})}(\mathbf{z};\Lambda)}{\prod_{i=1}^{n} \Omega_{T(\mathbf{e}_i)}(\mathbf{z};\Lambda)^{2v_i^2 - \sum_{j=1}^n v_i v_j} \prod_{\substack{1 \le i,j \le n \\ i \ne j}} \Omega_{T(\mathbf{e}_i+\mathbf{e}_j)}(\mathbf{z};\Lambda)^{v_i v_j}}.$$

Proof. A straightforward calculation using (5.1). □

Lemma 5.1.3.

$$\wp(z) - \wp(w) = -\frac{\sigma(z+w)\sigma(z-w)}{\sigma(z)^2\sigma(w)^2}, \tag{5.5}$$

$$\wp(nz) - \wp(mz) = -\frac{\Omega_{m+n}(z)\Omega_{m-n}(z)}{\Omega_m(z)^2\Omega_n(z)^2}. \tag{5.6}$$

Proof. The first statement (5.5) is a standard result (for example [10, IV.3] or [34, VI.350]). The second statement (5.6) follows by direct calculation. □

Lemma 5.1.4.

$$\zeta(x+a) - \zeta(a) - \zeta(x+b) + \zeta(b) = \frac{\sigma(x+a+b)\sigma(x)\sigma(a-b)}{\sigma(x+a)\sigma(x+b)\sigma(a)\sigma(b)}, \tag{5.7}$$

$$\zeta(x+a+b) - \zeta(x+a) - \zeta(x+b) + \zeta(x) = \frac{\sigma(2x+a+b)\sigma(a)\sigma(b)}{\sigma(x+a+b)\sigma(x+a)\sigma(x+b)\sigma(x)}. \tag{5.8}$$

Proof. Denote by f and g the left and right side of (5.7) respectively. Considered as functions of any one of x, a or b, these are elliptic functions. Suppose that $a, b \notin \Lambda$. Consider f and g as functions of x. The set of poles of f or g is $\{-a, -b\}$. The zeroes of g are at $-a-b$ and 0. These are also zeroes of f, since ζ is an odd function. Hence we have $f = cg$ for some c not depending on x. Now define instead

$$F = (\zeta(x+a) - \zeta(a) - \zeta(x+b) + \zeta(b))\,\sigma(x+a)\sigma(x+b),$$
$$G = \sigma(x+a+b)\sigma(x).$$

We have $F = c'G$ for some constant c' independent of x. Taking derivatives and evaluating at $x = 0$, we have

$$(\wp(b) - \wp(a))\,\sigma(a)\sigma(b) = c'\sigma(a+b)\sigma'(0)$$

We have $\sigma'(0) = 1$. By Lemma 5.1.3, we then have

$$c' = -\frac{\sigma(a-b)}{\sigma(a)\sigma(b)}$$

which proves the first equation (5.7). The second equation (5.8) is obtained by a change of variables $x \leftarrow a$, $a \leftarrow x+b$, $b \leftarrow x$. □

5.2 Forming the net

Theorem 5.2.1. *Fix an elliptic curve E over $\mathbb{C}$ and points $P_1, \ldots, P_n \in E(\mathbb{C})$. Let $\Lambda \subset \mathbb{C}$ be the lattice associated to E and $z_1, \ldots, z_n$ the points associated to $P_1, \ldots, P_n$ respectively. Then the function $W : \mathbb{Z}^n \to \mathbb{C}$ defined by*

$$W(\mathbf{v}) = \Omega_{\mathbf{v}}(z_1, \ldots, z_n; \Lambda)$$

is an elliptic net.

Proof. For notational simplicity, we drop the arguments z_i, Λ and also write $\sigma(\mathbf{v})$, $\wp(\mathbf{v})$ and $\zeta(\mathbf{v})$ for $\sigma(v_1 z_1 + \ldots + v_n z_n)$, $\wp(v_1 z_1 + \ldots + v_n z_n)$ and $\zeta(v_1 z_1 + \ldots + v_n z_n)$.

We wish to demonstrate the recurrence relation (3.1). We observe that $\mathbf{v} = \mathbf{0}$ if and only if $\Omega_{\mathbf{v}} \equiv 0$. Therefore, we may assume that none of $\mathbf{p}$, $\mathbf{q}$ or $\mathbf{r}$ are zero, for if so, then the recurrence relation (3.1) holds trivially. Hence none of $\Omega_{\mathbf{p}}$, $\Omega_{\mathbf{q}}$, or $\Omega_{\mathbf{r}}$ is identically zero.

For any quadratic form f defined on $\mathbb{Z}^n$, we have the following relation for all $\mathbf{p}, \mathbf{q}, \mathbf{s} \in \mathbb{Z}^n$:

$$f(\mathbf{p}+\mathbf{q}+\mathbf{s}) + f(\mathbf{p}-\mathbf{q}) + f(\mathbf{s}) - f(\mathbf{p}+\mathbf{s}) - f(\mathbf{p}) - f(\mathbf{q}+\mathbf{s}) - f(\mathbf{q}) = 0. \tag{5.9}$$

First we address the case that $\mathbf{s} = \mathbf{0}$. By (5.9) and Lemma 5.1.3,

$$\frac{\Omega_{\mathbf{p}+\mathbf{q}}\Omega_{\mathbf{p}-\mathbf{q}}}{\Omega_{\mathbf{p}}^2\Omega_{\mathbf{q}}^2} = \frac{\sigma(\mathbf{p}+\mathbf{q})\sigma(\mathbf{p}-\mathbf{q})}{\sigma(\mathbf{p})^2\sigma(\mathbf{q})^2} = \wp(\mathbf{q}) - \wp(\mathbf{p}).$$

Therefore, we have

$$\frac{\Omega_{\mathbf{p}+\mathbf{q}}\Omega_{\mathbf{p}-\mathbf{q}}}{\Omega_{\mathbf{p}}^2\Omega_{\mathbf{q}}^2} + \frac{\Omega_{\mathbf{q}+\mathbf{r}}\Omega_{\mathbf{q}-\mathbf{r}}}{\Omega_{\mathbf{q}}^2\Omega_{\mathbf{r}}^2} + \frac{\Omega_{\mathbf{r}+\mathbf{p}}\Omega_{\mathbf{r}-\mathbf{p}}}{\Omega_{\mathbf{r}}^2\Omega_{\mathbf{p}}^2} = 0,$$

which gives the relation (3.1) for $\mathbf{s} = \mathbf{0}$:

$$\Omega_{\mathbf{p}+\mathbf{q}}\Omega_{\mathbf{p}-\mathbf{q}}\Omega_{\mathbf{r}}^2 + \Omega_{\mathbf{q}+\mathbf{r}}\Omega_{\mathbf{q}-\mathbf{r}}\Omega_{\mathbf{p}}^2 + \Omega_{\mathbf{r}+\mathbf{p}}\Omega_{\mathbf{r}-\mathbf{p}}\Omega_{\mathbf{q}}^2 = 0.$$

Now suppose that $\mathbf{s} \neq \mathbf{0}$ and so $\Omega_{\mathbf{s}} \not\equiv 0$. By (5.9) and Lemma 5.1.4,

$$\frac{\Omega_{\mathbf{p}+\mathbf{q}+\mathbf{s}}\Omega_{\mathbf{p}-\mathbf{q}}\Omega_{\mathbf{s}}}{\Omega_{\mathbf{p}+\mathbf{s}}\Omega_{\mathbf{p}}\Omega_{\mathbf{q}+\mathbf{s}}\Omega_{\mathbf{q}}} = \frac{\sigma(\mathbf{p}+\mathbf{q}+\mathbf{s})\sigma(\mathbf{p}-\mathbf{q})\sigma(\mathbf{s})}{\sigma(\mathbf{p}+\mathbf{s})\sigma(\mathbf{p})\sigma(\mathbf{q}+\mathbf{s})\sigma(\mathbf{q})} = \zeta(\mathbf{p}+\mathbf{s}) - \zeta(\mathbf{p}) - \zeta(\mathbf{q}+\mathbf{s}) + \zeta(\mathbf{q}).$$

Therefore, we have

$$\frac{\Omega_{p+q+s}\Omega_{p-q}\Omega_s}{\Omega_{p+s}\Omega_p\Omega_{q+s}\Omega_q}+\frac{\Omega_{q+r+s}\Omega_{q-r}\Omega_s}{\Omega_{q+s}\Omega_q\Omega_{r+s}\Omega_r}+\frac{\Omega_{r+p+s}\Omega_{r-p}\Omega_s}{\Omega_{r+s}\Omega_r\Omega_{p+s}\Omega_p}=0,$$

or, more simply,

$$\Omega_{p+q+s}\Omega_{p-q}\Omega_{r+s}\Omega_r+\Omega_{q+r+s}\Omega_{q-r}\Omega_{p+s}\Omega_p+\Omega_{r+p+s}\Omega_{r-p}\Omega_{q+s}\Omega_q=0,$$

which is what was required to prove. □

This identity is known in various forms in complex function theory. See [34, VI.359] and [77].

Chapter 6

Elliptic net polynomials

In the case of rank one, the functions $\Omega_{\mathbf{v}}$ are well-known to have a polynomial representation in terms of the coefficients of the Weierstrass equation and the coordinates $x = \wp(z), y = \wp'(z)$. These are called *division polynomials* (see Section 1.2 or [63, Ex 3.7]). We wish to create analogous *net polynomials*[1] in the higher rank case. To begin, we calculate some representations explicitly. Then, using the recurrence structure of the net, we define the net polynomials in general. Finally, in Section 6.3, we show that they have properties analogous to those of division polynomials, i.e., have a restricted form with integer coefficients and do not vanish modulo primes p.

6.1 Defining net polynomials

Define $L = \mathbb{Q}(\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_6)$ to be a field of transcendence degree five over the rationals; α_i are the indeterminates. Let

$$f(x,y) = y^2 + \alpha_1 xy + \alpha_3 y - x^3 - \alpha_2 x^2 - \alpha_4 x - \alpha_6. \tag{6.1}$$

Define the ring

$$\mathcal{L}_n = L[x_i, y_i]_{1\le i,j,\le n} \left[(x_i - x_j)^{-1} \right]_{1\le i<j\le n} \Big/ \left\langle f(x_i, y_i) \right\rangle_{1\le i\le n}.$$

In this section, we wish to define an elliptic net $\Psi_{\mathbf{v}}$ of elements of $\mathcal{L}_n$. More specifically,

Theorem 6.1.1. *Let n be a positive integer. There exist $\Psi_{\mathbf{v}} \in \mathcal{L}_n$ indexed by $\mathbf{v} \in \mathbb{Z}^n$ such that the following holds: For each elliptic curve E specified by Weierstrass coefficients $a_1, \dots a_6 \in \mathbb{Q}$, consider the quotient map $\mathcal{L}_n \to \mathbb{Q}(E^n)$ defined by taking $\alpha_i \mapsto a_i$. The functions $\Omega_{\mathbf{v}}$ are elements of $\mathbb{Q}(E^n)$ and this map takes $\Psi_{\mathbf{v}} \mapsto \Omega_{\mathbf{v}}$. Furthermore, the $\Psi_{\mathbf{v}}$ form an elliptic net.*

Note that the theorem claims that the $\Omega_{\mathbf{v}}$ are elements of $\mathbb{Q}(E^n)$ and that they are the images of some $\Psi_{\mathbf{v}}$ having special properties. The demonstrations of these two claims are intimately intertwined.

First we calculate a few of the $\Omega_{\mathbf{v}}$ explicitly, and see that they are elements of $\mathbb{Q}(E^n)$.

[1] One word of warning: net polynomials will not be polynomials in $\wp(z_i), \wp'(z_i)$, but instead may involve a restricted set of polynomial denominators. Thus, the name may be a slight–but justifiable–misnomer.

Proposition 6.1.2. *Consider an elliptic curve defined over the rationals with Weierstrass equation*

$$y^2 + a_1xy + a_3y - x^3 - a_2x^2 - a_4x - a_6 = 0.$$

Define b_2, b_4, b_6 and b_8 in the usual way (see (1.4)*). We have the following expressions for $n = 1$:*

$$\begin{aligned}
&\Omega_1 = 1, \qquad \Omega_2 = 2y + a_1x + a_3,\\
&\Omega_3 = 3x^4 + b_2x^3 + 3b_4x^2 + 3b_6x + b_8,\\
&\Omega_4 = (2y + a_1x + a_3)(2x^6 + b_2x^5 + 5b_4x^4 + 10b_6x^3 + 10b_8x^2 + (b_2b_8 - b_4b_6)x + b_4b_8 - b_6^2);
\end{aligned}$$

and for $n = 2$:

$$\begin{aligned}
\Omega_{(1,-1)} &= x_2 - x_1,\\
\Omega_{(2,1)} &= 2x_1 + x_2 - \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - a_1\left(\frac{y_2 - y_1}{x_2 - x_1}\right) + a_2.\\
\Omega_{(2,-1)} &= (y_1 + y_2)^2 - (2x_1 + x_2)(x_1 - x_2)^2 \ .
\end{aligned}$$

Proof. These are classical results (see for example [10, III.4, IV.3], [22, 4.4.5.a], [34, VI.349, 352] and [63, Ex 3.7]) which can be calculated using the formulæ for addition on the curve. □

Proposition 6.1.3. *The terms $\Omega_{\mathbf{v}}$ indexed by the set*

$$S_3 = \{\mathbf{v} \in \mathbb{Z}^3 : N(\mathbf{v}) = 1\}$$

have representations as rational functions in the $x_i = \wp(z_i)$ and $y_i = \wp'(z_i)$ for $i = 1, 2, 3$. These representations can be chosen to have denominators a product of at most three linear terms of the form $x_i - x_j$ for some $1 \le i < j \le 3$.

Proof. We have $\Omega_{(1,0,0)} = \Omega_{(0,1,0)} = \Omega_{(0,0,1)} = \Omega_{(1,1,0)} = \Omega_{(0,1,1)} = \Omega_{(1,0,1)} = 1$. We also have

$$\Omega_{(1,-1,0)} = x_2 - x_1, \qquad \Omega_{(0,1,-1)} = x_3 - x_2, \qquad \Omega_{(-1,0,1)} = x_1 - x_3,$$

and the corresponding negatives. That leaves only the cases where $v_i \neq 0$ for all $i = 1, 2, 3$.

We have the following recurrence:

$$\begin{matrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & -1 \end{matrix}
\left[\begin{array}{cccc|cccc|cccc}
1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 & 2 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
1 & 0 & -1 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 1
\end{array}\right],$$

which gives

$$\Omega_{(1,1,1)}\Omega_{(1,1,-1)} + \Omega_{(-1,0,1)} - \Omega_{(2,1,0)} = 0.$$

By Proposition 6.1.2,

$$\Omega_{(1,1,1)}\Omega_{(1,1,-1)} = x_1 + x_2 + x_3 - \left(\frac{y_1 - y_2}{x_1 - x_2}\right)^2 - a_1\left(\frac{y_1 - y_2}{x_1 - x_2}\right) + a_2. \tag{6.2}$$

We also have

$$\left[\begin{array}{cccc|cccc|cccc|cccc} 1 & 1 & 0 & -1 & 1 & 0 & -1 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 & 1 & 0 & 0 & -1 & 0 & 1 & 1 & 0 & 0 & -1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 & 2 & 0 & 0 & 0 \end{array}\right],$$

which gives

$$\Omega_{(1,1,1)}\Omega_{(-1,0,1)}\Omega_{(0,-1,1)} + \Omega_{(1,1,-1)} - \Omega_{(0,0,2)} = 0,$$

which becomes, by Proposition 6.1.2,

$$\Omega_{(1,1,1)}(x_1 - x_3)(x_2 - x_3) + \Omega_{(1,1,-1)} = 2y_3 + a_1 x_3 + a_3. \tag{6.3}$$

Multiplying (6.3) by $\Omega_{(1,1,1)}$ and using (6.2), we obtain

$$(2y_3 + a_1 x_3 + a_3)\Omega_{(1,1,1)} - (x_1 - x_3)(x_2 - x_3)\Omega_{(1,1,1)}^2 \\ = x_1 + x_2 + x_3 - \left(\frac{y_1 - y_2}{x_1 - x_2}\right)^2 - a_1\left(\frac{y_1 - y_2}{x_1 - x_2}\right) + a_2.$$

Similarly,

$$(2y_2 + a_1 x_2 + a_3)\Omega_{(1,1,1)} - (x_1 - x_2)(x_3 - x_2)\Omega_{(1,1,1)}^2 \\ = x_1 + x_2 + x_3 - \left(\frac{y_1 - y_3}{x_1 - x_3}\right)^2 - a_1\left(\frac{y_1 - y_3}{x_1 - x_3}\right) + a_2. \tag{6.4}$$

$$(2y_1 + a_1 x_1 + a_3)\Omega_{(1,1,1)} - (x_2 - x_1)(x_3 - x_1)\Omega_{(1,1,1)}^2 \\ = x_1 + x_2 + x_3 - \left(\frac{y_2 - y_3}{x_2 - x_3}\right)^2 - a_1\left(\frac{y_2 - y_3}{x_2 - x_3}\right) + a_2. \tag{6.5}$$

Adding $(x_3 - x_1)$ times (6.4) and $(x_3 - x_2)$ times (6.5), we obtain

$$\Omega_{(1,1,1)} = \frac{(2x_3 - x_1 - x_2)(x_1 + x_2 + x_3 + a_2) + \frac{(y_1 - y_3)^2}{x_1 - x_3} + \frac{(y_2 - y_3)^2}{x_2 - x_3} - a_1(2y_3 - y_1 - y_2)}{(x_3 - x_1)(2y_2 + a_1 x_2 + a_3) + (x_3 - x_2)(2y_1 + a_1 x_1 + a_3)} \tag{6.6}$$

Multiplying top and bottom by $(x_3 - x_1)(2y_2 + a_1 x_2 + a_3) - (x_3 - x_2)(2y_1 + a_1 x_1 + a_3)$, this becomes

$$\Omega_{(1,1,1)} = \frac{y_1(x_2 - x_3) + y_2(x_3 - x_1) + y_3(x_1 - x_2)}{(x_1 - x_2)(x_1 - x_3)(x_2 - x_3)} \tag{6.7}$$

and we also obtain from (6.2),

$$\Omega_{(-1,1,1)} = \frac{y_1(x_2 - x_3) - y_2(x_3 - x_1) - y_3(x_1 - x_2)}{(x_2 - x_3)} + a_1 x_1 + a_3,$$
$$\Omega_{(1,-1,1)} = \frac{-y_1(x_2 - x_3) + y_2(x_3 - x_1) - y_3(x_1 - x_2)}{(x_3 - x_1)} + a_1 x_2 + a_3,$$
$$\Omega_{(1,1,-1)} = \frac{-y_1(x_2 - x_3) - y_2(x_3 - x_1) + y_3(x_1 - x_2)}{(x_1 - x_2)} + a_1 x_3 + a_3.$$

This completes the necessary calculations. □

Proof of Theorem 6.1.1. We use the inductive structure of elliptic nets. For $\mathbf{v}$ of dimension less than 4 satisfying $N(\mathbf{v}) = 1$, define $\Psi_{\mathbf{v}}$ to be the rational functions given in Propositions 6.1.2 and 6.1.3, with each a_i replaced by α_i.

By the collection of inductive results in Chapter 4 (specifically, Theorems 4.2.1, 4.2.3, and 4.3.3), we may define all other $\Psi_{\mathbf{v}}$ using recurrence relations, but we do not yet know that this method gives something well-defined (that is, a different choice of recurrence relations may yield a different result). That is to say, we do not know that the $\Psi_{\mathbf{v}}$ form an elliptic net. Nevertheless, we may take for our definition some arbitrary choice of 'implications' (in the terminology of Chapter 4), and then we know the $\Psi_{\mathbf{v}}$ thus defined must lie within the field of fractions $\mathrm{Frac}(\mathcal{L}_n)$ of $\mathcal{L}_n$.

Define

$$\mathcal{S}_n = \mathbb{Q}[\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_6][x_i, y_i]_{1 \le i,j, \le n} \left[(x_i - x_j)^{-1} \right]_{1 \le i < j \le n} \Big/ \left\langle f(x_i, y_i) \right\rangle_{1 \le i \le n}.$$

The ring injection

$$\mathbb{Q}[\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_6] \longrightarrow \mathcal{S}_n$$

defines $\mathrm{Spec}\, \mathcal{S}_n$ as a variety over $\mathbb{A}^5_{\mathbb{Q}}$. Let C be the curve defined by a polynomial of the form

$$f_C(x, y) = y^2 + a_1 xy + a_3 y - x^3 - a_2 x^2 - a_4 x - a_6,$$

where $a_i \in \mathbb{Q}$. Then define

$$\mathcal{Q}_{n,C} = \mathbb{Q}[x_i, y_i]_{1 \le i,j, \le n} \left[(x_i - x_j)^{-1} \right]_{1 \le i < j \le n} \Big/ \left\langle f_C(x_i, y_i) \right\rangle_{1 \le i \le n}.$$

The fibres of $\mathrm{Spec}\, \mathcal{S}_n$ over $\mathbb{A}^5_{\mathbb{Q}}$ are $\mathrm{Spec}\, \mathcal{Q}_{n,C}$, which is an affine piece of C^n for the curve C over $\mathbb{Q}$. The fibres which are $\mathrm{Spec}\, \mathcal{Q}_{n,E}$ for an elliptic curve E form a Zariski open dense set in $\mathrm{Spec}\, \mathcal{S}_n$. The inclusion of these fibres gives quotient maps defined on $\mathcal{S}_n \to \mathcal{Q}_{n,C}$ or $\mathrm{Frac}(\mathcal{S}_n) \to \mathrm{Frac}(\mathcal{Q}_{n,C})$ by $\alpha_i \mapsto a_i$; in either case, call this quotient map ϕ_C.

Let us return momentarily to the $\Omega_{\mathbf{v}}$ (defined for an elliptic curve over $\mathbb{C}$). We know that these lie in $\mathrm{Frac}(\mathcal{Q}_{n,E}) \subset \mathbb{Q}(E)$ since they lie in this field on a baseset (by Propositions 6.1.2 and 6.1.3), and we can induct using the results of Chapter 4. Further, by consideration of the divisor of $\Omega_{\mathbf{v}}$, it is clear that in fact $\Omega_{\mathbf{v}} \in \mathcal{Q}_{n,E}$. If E is an elliptic curve, the image $\phi_E(\Psi_{\mathbf{v}})$ is exactly $\Omega_{\mathbf{v}}$ by definition.

The function $\Psi_{\mathbf{v}}$ is a rational function on $\mathrm{Spec}\, \mathcal{S}_n$ (note that $\mathrm{Frac}(\mathcal{L}_n) = \mathrm{Frac}(\mathcal{S}_n)$). On each fibre $\mathrm{Spec}\, \mathcal{Q}_{n,E}$, it restricts to $\Omega_{\mathbf{v}}$, which is regular (since $\Omega_{\mathbf{v}} \in \mathcal{Q}_{n,E}$). Therefore, the support of the divisor of poles of $\Psi_{\mathbf{v}}$ must consist only of some number of vertical fibres. That is to say, $\Psi_{\mathbf{v}} \in \mathcal{L}_n$.

Finally, a rational function on $\mathrm{Spec}\, \mathcal{S}_n$ that is zero on a Zariski open dense subset must be zero. Therefore, the ring homomorphism

$$\left(\prod_{E \text{ elliptic}} \phi_E \right) : \mathcal{L}_n \to \prod_{E \text{ elliptic}} \mathcal{Q}_{n,E}$$

is injective. Since the image of the collection $\{\Psi_{\mathbf{v}}\}$ under this map forms an elliptic net, it must be that the $\Psi_{\mathbf{v}}$ form an elliptic net. □

We call these $\Psi_{\mathbf{v}}$ the *net polynomials*. We can now restate the formulæ given in Proposition 6.1.2 for the more general $\Psi_{\mathbf{v}}$.

Proposition 6.1.4. *Consider an elliptic curve defined over the rationals with Weierstrass equation*

$$y^2 + a_1xy + a_3y - x^3 - a_2x^2 - a_4x - a_6 = 0.$$

Define b_2, b_4, b_6 and b_8 as usual (see (1.4)*). We have the following expressions for $n = 1$:*

$$\Psi_1 = 1, \tag{6.8}$$

$$\Psi_2 = 2y + a_1x + a_3, \tag{6.9}$$

$$\Psi_3 = 3x^4 + b_2x^3 + 3b_4x^2 + 3b_6x + b_8, \tag{6.10}$$

$$\Psi_4 = (2y + a_1x + a_3)(2x^6 + b_2x^5 + 5b_4x^4 + 10b_6x^3 + 10b_8x^2 + (b_2b_8 - b_4b_6)x + b_4b_8 - b_6^2); \tag{6.11}$$

for $n = 2$:

$$\Psi_{(1,-1)} = x_2 - x_1, \tag{6.12}$$

$$\Psi_{(2,1)} = 2x_1 + x_2 - \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - a_1\left(\frac{y_2 - y_1}{x_2 - x_1}\right) + a_2, \tag{6.13}$$

$$\Psi_{(2,-1)} = (y_1 + y_2)^2 - (2x_1 + x_2)(x_1 - x_2)^2 \ ; \tag{6.14}$$

and for $n = 3$:

$$\Psi_{(1,1,1)} = \frac{y_1(x_2 - x_3) + y_2(x_3 - x_1) + y_3(x_1 - x_2)}{(x_1 - x_2)(x_1 - x_3)(x_2 - x_3)}, \tag{6.15}$$

$$\Psi_{(-1,1,1)} = \frac{y_1(x_2 - x_3) - y_2(x_3 - x_1) - y_3(x_1 - x_2)}{(x_2 - x_3)} + a_1x_1 + a_3, \tag{6.16}$$

$$\Psi_{(1,-1,1)} = \frac{-y_1(x_2 - x_3) + y_2(x_3 - x_1) - y_3(x_1 - x_2)}{(x_3 - x_1)} + a_1x_2 + a_3, \tag{6.17}$$

$$\Psi_{(1,1,-1)} = \frac{-y_1(x_2 - x_3) - y_2(x_3 - x_1) + y_3(x_1 - x_2)}{(x_1 - x_2)} + a_1x_3 + a_3. \tag{6.18}$$

Proof. These formulæ follow from Proposition 6.1.2, the proof of Proposition 6.1.3 and Theorem 6.1.1. □

6.2 Properties of net polynomials

In this section, we transfer some useful properties of the $\Omega_{\mathbf{v}}$ from the complex context to the context of the $\Psi_{\mathbf{v}}$. In particular, we are interested in the generalisation of Proposition 5.1.2.

Lemma 6.2.1. *Let E be an elliptic curve. With the notation of the proof of Theorem 6.1.1, for each positive integer m there exist $x_{\times m}, y_{\times m} \in \mathrm{Frac}(\mathcal{L}_1)$ such that, considered as complex functions on E,*

$$\left(\phi_E(x_{\times m})\right)(z) = \wp(mz), \quad \text{and} \quad \left(\phi_E(y_{\times m})\right)(z) = \wp'(mz).$$

Furthermore, there are functions x_{add} and y_{add} in $\mathrm{Frac}(\mathcal{L}_2)$ *such that*

$$\left(\phi_E(x_{add})\right)(z,w) = \wp(z+w), \quad \textit{and} \quad \left(\phi_E(y_{add})\right)(z,w) = \wp'(z+w).$$

Proof. The group law of an elliptic curve gives the following equations

$$\wp(mz) = \left(\frac{\wp'((m-1)z)-\wp'(z)}{\wp((m-1)z)-\wp(z)}\right)^2 + a_1\left(\frac{\wp'((m-1)z)-\wp'(z)}{\wp((m-1)z)-\wp(z)}\right) - a_2 - \wp((m-1)z) - \wp(z), \quad (6.19)$$

$$\wp'(mz) = -\left(\frac{\wp'((m-1)z)-\wp'(z)}{\wp((m-1)z)-\wp(z)} + a_1\right)\wp(mz) - \left(\frac{\wp'(z)\wp((m-1)z)-\wp'((m-1)z)\wp(z)}{\wp((m-1)z)-\wp(z)}\right) - a_3. \quad (6.20)$$

The exact form is not important; only that, for $m > 2$, these define $\wp(mz)$ and $\wp'(mz)$ inductively from $\wp(nz)$ and $\wp'(nz)$ for $n < m$. For $m = 1, 2$, let

$$x_{\times 1} = x, \qquad x_{\times 2} = \frac{x^4 - b_4x^2 - 2b_6x - b_8}{4x^3 + b_2x^2 + 2b_4x + b_6}.$$

So $(\phi_E(x_{\times 1}))(z) = \wp(z)$ and $(\phi_E(x_{\times 2}))(z) = \wp(2z)$. Define $x_{\times m}, y_{\times m}$ inductively from these using formulæ of the same form as (6.19) and (6.20):

$$x_{\times m} = \left(\frac{y_{\times m-1}-y_{\times 1}}{x_{\times m-1}-x_{\times 1}}\right)^2 + a_1\left(\frac{y_{\times m-1}-y_{\times 1}}{x_{\times m-1}-x_{\times 1}}\right) - a_2 - x_{\times m-1} - x_{\times 1},$$

$$y_{\times m} = -\left(\frac{y_{\times m-1}-y_{\times 1}}{x_{\times m-1}-x_{\times 1}} + a_1\right)x_{\times m} - \left(\frac{y_{\times 1}x_{\times m-1}-y_{\times m-1}x_{\times 1}}{x_{\times m-1}-x_{\times 1}}\right) - a_3.$$

Then $x_{\times m}, y_{\times m} \in \mathrm{Frac}(\mathcal{L}_1)$ and by definition

$$(\phi_E(x_{\times m}))(z) = \wp(mx) \quad \text{and} \quad (\phi_E(y_{\times m}))(z) = \wp'(mx)$$

for all E and m.

As for x_{add} and y_{add}, these also have explicit rational function representations, and so we can do the same; see any elementary text on elliptic curves. □

Since the same equations define the group law on the nonsingular part of a singular cubic curve, we may also refer to $x_{\times m}$,$y_{\times m}$, x_{add} and y_{add} as giving the coordinates of the multiplication-by-m or addition maps in this context.

Lemma 6.2.2.

$$\Psi_m^2\Psi_n^2(x_{\times m} - x_{\times n}) = -\Psi_{m+n}\Psi_{m-n}. \quad (6.21)$$

Proof. This follows immediately from Lemma 5.1.3 and the fact that the map

$$\left(\prod_{E \text{ elliptic}} \phi_E\right) : \mathcal{L}_n \to \prod_{E \text{ elliptic}} \mathcal{Q}_{n,E}$$

of Theorem 6.1.1 is injective. □

Now we make an important definition. For each $\mathbf{v} \in \mathbb{Z}^m$ and T an $n \times m$ matrix, we let $\Psi_{\mathbf{v}} \circ T$ be the rational function in $\mathrm{Frac}(\mathcal{L}_n)$ obtained by substituting for x_i, y_i the rational function expressions for the x and y coordinates of

$$[T_{i,1}](x_1, y_1) + [T_{i,2}](x_2, y_2) + \ldots + [T_{i,m}](x_m, y_m).$$

In essence, we precompose by a linear transformation on the points. Lemma 6.2.1 ensures that this is well-defined. Now, since the map

$$\left(\prod_{E \text{ elliptic}} \phi_E \right) : \mathcal{L}_n \to \prod_{E \text{ elliptic}} \mathcal{Q}_{n,E}$$

of Theorem 6.1.1 is injective, Proposition 5.1.2 holds in the context of the $\Psi_{\mathbf{v}}$. To be precise,

Theorem 6.2.3. *Let $\mathbf{v} \in \mathbb{Z}^n$. Let T be any $n \times m$ matrix with entries in $\mathbb{Z}$ and transpose T^{tr}. Then*

$$(\Psi_{\mathbf{v}} \circ T) \left(\prod_{i=1}^{n} \Psi_{T^{tr}(\mathbf{e}_i)}^{v_i^2 - \Sigma_{j \neq i}^n v_i v_j} \right) \left(\prod_{1 \leq i < j \leq n} \Psi_{T^{tr}(\mathbf{e}_i + \mathbf{e}_j)}^{v_i v_j} \right) = \Psi_{T^{tr}(\mathbf{v})}.$$

6.3 Net polynomials at primes

We have now defined an elliptic net $\Psi_{\mathbf{v}} \in \mathcal{L}_n$. In this section we determine a little more about the exact nature of this net. Loosely speaking, we wish to show that the coefficients of the net polynomials are contained in a proper subring $R = \mathbb{Z}[\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_6]$ of the field $L = \mathbb{Q}(\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_6)$ defined at the beginning of Section 6.1 (note that R has field of fractions L). This is analogous to the rank one statement that the division polynomials are polynomials in the coefficients of the curve and the variables x and y with *integer* coefficients. Furthermore, we wish to show the analogue of the statement that the division polynomial Ψ_n does not vanish modulo p for every prime p and integer n.

To be more precise, define

$$\mathcal{R}_n = R[x_i, y_i]_{1 \leq i,j, \leq n} \left[(x_i - x_j)^{-1} \right]_{1 \leq i < j \leq n} \Big/ \left\langle f(x_i, y_i) \right\rangle_{1 \leq i \leq n},$$

which injects into $\mathcal{L}_n$.

Define the usual quantities

$$\begin{aligned}
&\beta_2 = \alpha_1^2 + 4\alpha_2, \qquad \beta_4 = 2\alpha_4 + \alpha_1\alpha_3, \qquad \beta_6 = \alpha_3^2 + 4\alpha_6, \\
&\beta_8 = \alpha_1^2\alpha_6 + 4\alpha_2\alpha_6 - \alpha_1\alpha_3\alpha_4 + \alpha_2\alpha_3^2 - \alpha_4^2, \\
&\gamma_4 = \beta_2^2 - 24\beta_4, \qquad \delta = -\beta_2^2\beta_8 - 8\beta_4^3 - 27\beta_6^2 + 9\beta_2\beta_4\beta_6.
\end{aligned}$$

Theorem 6.3.1. *The functions $\Psi_{\mathbf{v}}$ are elements of $\mathcal{R}_n \subset \mathcal{L}_n$. Let $\mathfrak{p}$ be any prime of $\mathcal{R}_n$ that is a lifting of a prime of R, and that does not contain both δ and γ_4. Then $\Psi_{\mathbf{v}} \notin \mathfrak{p}$.*

Proof. For $n = 1, 2$, Theorems 4.2.1 and 4.2.3, and Proposition 6.1.2 imply that $\Psi_{\mathbf{v}} \in \mathcal{R}_n$.

We now consider the second statement in rank one and two. Note that it suffices to consider maximal ideals $\mathfrak{p}$, since all prime ideals are contained in some maximal ideal. In dimension one, the statement is a consequence of Lemma 6.2.2 as follows. Equation (6.21) implies:

$$\Psi_{m-1}^2(x_{\times m-1} - x_{\times 1}) = -\Psi_m \Psi_{m-2}. \tag{6.22}$$

By the explicit form of Ψ_1 and Ψ_2, we know they are not in $\mathfrak{p}$. We claim that $x_{\times m-1} - x_{\times 1} \notin \mathfrak{p}$ for any $m > 1$. If the claim holds, we may use (6.22) and induction on m to show that $\Psi_m \notin \mathfrak{p}$ for all $m > 2$. It remains to prove the claim.

The claim is exactly the statement that for $m > 2$ the multiplication-by-m map $[m]$ is not the identity map on a non-singular fibre or the nonsingular part of a nodal singular fibre over $\mathfrak{p}$ of the elliptic scheme

$$y^2 + \alpha_1 xy + \alpha_3 y = x^3 + \alpha_2 x^2 + \alpha_4 x + \alpha_6$$

over R. If $[m] = [1]$ on some such fibre, then $[m] = [1]$ on some one-dimensional fibre, i.e., on some curve over $\mathbb{F}_p$. If this is the case, then $[m-1] = [0]$ on this curve, a contradiction.

The rank two case may be reduced to the rank one case by the use of Theorem 6.2.3. We know that $\Psi_{\mathbf{v}}$ is integral, so we show that it is not contained in any prime $\mathfrak{p}$ lifted from R. It suffices to show this for minimal such primes (where we can use the language of valuations), since if it is contained in any prime, then it is contained in a minimal such one. In particular, first use

$$T = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix},$$

For any k, we have

$$(\Psi_{(k,k)} \circ T)\Psi_{2,0}^{k^2} = \Psi_{2k,0}.$$

Consider the valuation ν on $\mathrm{Frac}(\mathcal{L}_n)$ associated to the ideal $\mathfrak{p}$. The terms $\Psi_{2,0}^{k^2}$ and $\Psi_{2k,0}$ have zero valuation, and so $\nu(\Psi_{(k,k)} \circ T) = 0$. We also know that $\nu(\Psi_{(k,k)}) \geq 0$. Pre-composition by T means substituting $x_2 := x_1$ and $y_2 := y_1$. So, $\nu(\Psi_{(k,k)}) \leq \nu(\Psi_{(k,k)} \circ T) = 0$. Hence $\nu(\Psi_{(k,k)}) = 0$.

Now suppose that $n = 2$, so $\mathbf{v} = (v_1, v_2)$. Without loss of generality, $v_1 v_2 \neq 0$. For now, assume $v_1 v_2 > 0$. Using

$$T = \begin{pmatrix} v_2 & 0 \\ 0 & v_1 \end{pmatrix},$$

we obtain

$$(\Psi_{(v_1,v_2)} \circ T)\Psi_{(v_2,v_1)}^{v_1 v_2}\Psi_{(v_2,0)}^{v_1^2 - v_1 v_2} = \Psi_{(v_1 v_2, v_1 v_2)}\Psi_{(0,v_1)}^{v_1 v_2 - v_2^2}.$$

By the previous cases, then,

$$\nu((\Psi_{(v_1,v_2)} \circ T)\Psi_{(v_2,v_1)}^{v_1 v_2}) = 0.$$

In this case, pre-composition by T means substituting

$$x_1 := \phi_{v_2}\Psi_{v_2}^{-2}, \qquad y_1 := \omega_{v_2}\Psi_{v_2}^{-3}, \qquad x_2 := \phi_{v_1}\Psi_{v_1}^{-2}, \qquad y_2 := \omega_{v_1}\Psi_{v_1}^{-3},$$

where $\gcd(\phi_{v_1}, \Psi_{v_1}) = \gcd(\phi_{v_2}, \Psi_{v_2}) = \gcd(\omega_{v_1}, \Psi_{v_1}) = \gcd(\omega_{v_2}, \Psi_{v_2}) = 1$. Since Ψ_{v_2} has zero valuation,

$$\nu(\Psi_{(v_1,v_2)} \circ T) \geq \nu(\Psi_{(v_1,v_2)}) \geq 0.$$

By symmetry, $\Psi_{(v_1,v_2)}$ and $\Psi_{(v_2,v_1)}$ have the same valuation, and so it must be that $\nu(\Psi_{(v_1,v_2)}) = 0$.

For the case $v_1 v_2 < 0$ some minor modification is required; in the first part, show $\nu(\Psi_{(k,-k)}) = 0$ and then use

$$T = \begin{pmatrix} v_2 & 0 \\ 0 & -v_1 \end{pmatrix}$$

in the second part.

Now we wish to show the case of general n. We first show by induction on n that terms indexed by S_n have valuation zero. This holds for $n = 2$ by the previous case. We use the second statement of Theorem 4.3.3 for the inductive step. The terms indexed by S'_n have zero valuation by the inductive hypothesis, and therefore terms of S_n have non-negative valuation (i.e., these terms are in $\mathcal{R}_n$). To show that they have zero valuation requires an application of Theorem 6.2.3.

In this case, use the $n \times 2$ matrix

$$T = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ \vdots & \vdots \\ 1 & 0 \\ 0 & 1 \end{pmatrix},$$

which gives

$$(\Psi_{(1,1,\ldots,1)} \circ T)\Psi_{(2,0)}^n = \Psi_{(n,1)}$$

Therefore, $\nu(\Psi_{(1,1,\ldots,1)} \circ T) = 0$, and hence $\nu(\Psi_{(1,1,\ldots,1)}) = 0$.

This completes the induction.

Now, by the first part of Theorem 4.3.3, in the rank one case, all Ψ_n have non-negative valuation (i.e., are in $\mathcal{R}_n$). We need to show that they are also non-positive, and again, this will be an application of Theorem 6.2.3.

Use the same matrix T and this time we obtain

$$(\Psi_{\mathbf{v}} \circ T)\Psi_{(2,0)}^{\sum_{1 \leq i < j < n} v_i v_j} = \Psi_{(\sum_{i=1}^{n-1} v_i, v_n)}.$$

Therefore, $\nu(\Psi_{\mathbf{v}} \circ T) = 0$ and hence $\nu(\Psi_{\mathbf{v}}) = 0$.

This completes the proof. □

Chapter 7

A curve gives a net

Let E be an elliptic curve over any field K. In this chapter we collect the results of the previous chapters to define functions $\Psi_{\mathbf{v}} : E^n \to K$ for all $\mathbf{v} \in \mathbb{Z}^n$. We show that we can do so in such a way that the map $W_{E,\mathbf{P}} : \mathbb{Z}^n \to K$ given by fixing $\mathbf{P} \in E^n$ and defining

$$W_{E,\mathbf{P}}(\mathbf{v}) = \Psi_{\mathbf{v}}(\mathbf{P})$$

is an elliptic net. Thanks to the generality of our previous results, we can do this for elliptic curves over an arbitrary field K. In the first section we state our most general result on the functions $\Psi_{\mathbf{v}}$ and in the second section we show how to make the construction for a given elliptic curve.

7.1 Net polynomials over arbitrary fields

Let $n \geq 1$. For any elliptic curve or scheme C, let $\mathcal{O}$ denote the identity, $[m] : C \to C$ denote multiplication by m, $p_i : C^n \to C$ denote projection onto the i-th component, and $s : C^n \to C$ denote the sum of all components. For $\mathbf{v} \in \mathbb{Z}^n$, define the expression

$$D_{C,\mathbf{v}} = ([v_1] \times \ldots \times [v_n])^* s^*(\mathcal{O}) - \sum_{1 \leq k < j \leq n} v_k v_j (p_k^* \times p_j^*) s^*(\mathcal{O}) - \sum_{k=1}^{n} \left(2v_k^2 - \sum_{j=1}^{n} v_k v_j \right) p_k^*(\mathcal{O}),$$

which is a divisor on the n-fold product C^n. Over the complex numbers, the functions $\Omega_{\mathbf{v}}$ have these divisors and satisfy the elliptic net recurrence (3.1).

We now collect the results of the previous sections.

Theorem 7.1.1. *Let*

$$f(x,y) = y^2 + \alpha_1 xy + \alpha_3 y - x^3 - \alpha_2 x^2 - \alpha_4 x - \alpha_6$$

define an elliptic scheme $E_{\mathbb{Z}}$ over the ring $R = \mathbb{Z}[\alpha_1, \ldots, \alpha_6]$ localised at (δ) and (γ_4). Let $n \geq 1$.

There exists a rational function $\Psi_{\mathbf{v}}$ on $E_{\mathbb{Z}}^n$ for each $\mathbf{v} \in \mathbb{Z}^n$ such that the collection of $\Psi_{\mathbf{v}}$ satisfies the following properties:

1. *The* $\Psi_{\mathbf{v}}$ *satisfy the recurrence* (3.1) *in terms of* $\mathbf{v}$.
2. $\Psi_{\mathbf{v}} = 1$ *whenever* $\mathbf{v} = \mathbf{e}_i$ *for some* $1 \le i \le n$ *or* $\mathbf{v} = \mathbf{e}_i + \mathbf{e}_j$ *for some* $1 \le i < j \le n$. *(Here* $\mathbf{e}_1, \ldots, \mathbf{e}_n$ *represent the standard basis vectors in* $\mathbb{Z}^n$.*)*
3. $\operatorname{div}(\Psi_{\mathbf{v}}) = D_{E_{\mathbb{Z}},\mathbf{v}}$.
4. *The* $\Psi_{\mathbf{v}}$ *can be expressed as elements of the ring*

$$\mathcal{R}_n = \mathbb{Z}[\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_6][x_i, y_i]_{i=1}^n \left[(x_i - x_j)^{-1}\right]_{1 \le i < j \le n} \Big/ \left\langle f(x_i, y_i) \right\rangle_{i=1}^n.$$

Proof. The $\Psi_{\mathbf{v}}$ are exactly those defined in the previous sections, which are elements of the ring $\mathcal{R}_n$. The ring $\mathcal{R}_n$ is the affine coordinate ring of the affine piece of $E_{\mathbb{Z}}^n$ obtained by removing all the axes, diagonals and antidiagonals. Therefore the $\Psi_{\mathbf{v}}$ are rational functions on $E_{\mathbb{Z}}^n$. We have seen that they satisfy Properties 1 and 2 (Theorem 6.1.1). The divisor at the generic point is just $D_{L_n,\mathbf{v}}$ since $\Psi_{\mathbf{v}} \in \mathcal{L}_n$. By Theorem 6.3.1, the divisors of the $\Psi_{\mathbf{v}}$ have no vertical components. Therefore, the divisor at the generic point extends. This gives Property 3. □

7.2 The elliptic net associated to a curve

Now fix any field K. Consider a curve C defined over K by the polynomial

$$f_C(x, y) = y^2 + a_1 xy + a_3 y - x^3 - a_2 x^2 - a_4 x - a_6.$$

The functions $\phi_C(\Psi_{\mathbf{v}})$ satisfy properties 1, 2 and 4 of Theorem 7.1.1 with C in place of $E_{\mathbb{Z}}$ and

$$\mathcal{R}_{n,C} = \mathbb{Z}[a_1, a_2, a_3, a_4, a_6][x_i, y_i]_{i=1}^n \left[(x_i - x_j)^{-1}\right]_{1 \le i < j \le n} \Big/ \left\langle f(x_i, y_i) \right\rangle_{i=1}^n$$

in place of $\mathcal{R}_n$. In particular, for $K = \mathbb{Q}$, the functions $\phi_C(\Psi_{\mathbf{v}})$ are exactly those $\Omega_{\mathbf{v}}$ defined in Chapter 5.

Note that the divisor of $\phi_C(\Psi_{\mathbf{v}})$ will be the pullback of the divisor $D_{E_{\mathbb{Z}},\mathbf{v}}$ under the map from C to $E_{\mathbb{Z}}$, but it may not have the form $D_{C,\mathbf{v}}$ because points may coincide. For example, if C is not supersingular over $\mathbb{F}_p$, then the divisor of $\phi_C(\Psi_p)$ over $\mathbb{F}_p$ has degree p at each of the p points of $E[p]$.

It is now natural to define

Definition 7.2.1. For any curve C with Weierstrass equation

$$f(x, y) = y^2 + a_1 xy + a_3 y - x^3 - a_2 x^2 - a_4 x - a_6$$

defined over a field K and point $\mathbf{P} = (P_1, \ldots, P_n) \in C(K)^n$ such that $P_i \neq \mathcal{O}$ for any i and $P_i \neq \pm P_j$ for $i \neq j$, we associate the elliptic net

$$W_{C,\mathbf{P}} : \mathbb{Z}^n \to K,$$

defined by

$$W_{C,\mathbf{P}}(\mathbf{v}) = \phi_C(\Psi_{\mathbf{v}}).$$

The conditions on the P_i arises from the requirement that $(x_i - x_j)^{-1}$, x_i, and y_i do not blowup in the formulæ for $\Psi_{\mathbf{v}}$.

We have the following additional corollary to Theorem 7.1.1.

Corollary 7.2.1. *For an elliptic net $W_{E,\mathbf{P}} : \mathbb{Z}^n \to K$ associated to an elliptic curve E, $\mathbf{P}$, we have $W(\mathbf{v}) = 0$ if and only if $\mathbf{v} \cdot \mathbf{P} = \mathcal{O}$ on E.*

Proof. Immediate from the divisor of $\Psi_{\mathbf{v}}$. □

Chapter 8

A net gives a curve

In the last chapter, we demonstrated a way to construct an elliptic net from an elliptic curve. In this chapter, we provide the other half of the 'Curve-Net Theorem,' by constructing, from a given elliptic net, a curve which will give rise to it.

8.1 Scale equivalence and normalisation

This section serves to set some useful definitions for the statement of the Curve-Net Theorem and subsequent chapters of the thesis.

Let B and C be abelian groups. Recall that a quadratic function $f : B \to C$ is a function such that for all $x, y, z \in B$,

$$f(x+y+z) - f(x+y) - f(y+z) - f(x+z) + f(x) + f(y) + f(z) = 0.$$

Proposition 8.1.1. *Let $W : A \to K$ be an elliptic net. Let $f : A \to K^*$ be a quadratic function. Define $W^f : A \to K$ by*

$$W^f(\mathbf{v}) = f(\mathbf{v})W(\mathbf{v}).$$

Then W^f is an elliptic net.

Proof. We use multiplicative notation in K^*, so that the quadratic function f satisfies

$$f(x+y+z)f(x)f(y)f(z)f(x+y)^{-1}f(y+z)^{-1}f(x+z)^{-1} = 1. \tag{8.1}$$

The parallelogram law for quadratic functions (written multiplicatively) states that

$$f(x-y) = f(x)^2 f(y)^2 f(x+y)^{-1}. \tag{8.2}$$

Equations (8.1) and (8.2) imply

$$f(p+q+s)f(p-q)f(r+s)f(r) = f(q+s)f(p+s)f(r+s)f(p)f(q)f(r)f(s)^{-1},$$

and so

$$f(p+q+s)f(p-q)f(r+s)f(r) = f(q+r+s)f(q-r)f(p+s)f(p)$$
$$= f(r+p+s)f(r-p)f(q+s)f(q).$$

This is a sort of symmetry property which suffices to show that the recurrence relation (3.1) holds for W^f. □

Definition 8.1.1. If two elliptic nets are related in the manner of W and W^f for some quadratic f, then we call them *scale equivalent*.

This is clearly an equivalence relation.

Definition 8.1.2. Let $W : \mathbb{Z}^n \to K$ be an elliptic net. Let $\mathbf{e}_1, \ldots, \mathbf{e}_n$ be the standard basis vectors in $\mathbb{Z}^n$. We say that W is *normalised* if $W(\mathbf{e}_i) = 1$ for all $1 \le i \le n$ and $W(\mathbf{e}_i + \mathbf{e}_j) = 1$ for all $1 \le i < j \le n$. If any term of the form $W(\mathbf{e}_i)$, $W(\mathbf{e}_i + \mathbf{e}_j)$, $W(\mathbf{e}_i - \mathbf{e}_j)$ is zero, or if $n = 1$ and any term of the form $W(2\mathbf{e}_1)$ or $W(3\mathbf{e}_1)$ is zero, then we say that W is *degenerate*.

The reason for the exact definition of degenerate given here will become clear in the following section. An elliptic net arising from an elliptic curve and points is normalised.

Proposition 8.1.2. *Let $W : \mathbb{Z}^n \to K$ be a non-degenerate elliptic net. Then there is exactly one scaling W^f which is normalised.*

Proof. We will give a function f which normalises W. Specify $A_{ij} \in K^*$ for $1 \le i \le j \le n$ as follows. Set $A_{ii} = W(\mathbf{e}_i)^{-1}$ for each $1 \le i \le n$ and

$$A_{ij} = \frac{W(\mathbf{e}_i)W(\mathbf{e}_j)}{W(\mathbf{e}_i + \mathbf{e}_j)},$$

for $1 \le i < j \le n$. Set

$$f(\mathbf{v}) = \prod_{1 \le i \le j \le n} A_{ij}^{v_i v_j}.$$

Uniqueness is clear. □

In particular, scale equivalence has

$$\binom{n}{2} + n = \binom{n+1}{2}$$

degrees of freedom, in the sense that any equivalence class is an $\binom{n+1}{2}$-dimensional vector space.

We define the *normalisation* of an elliptic net W to be the unique normalised elliptic net which is scale equivalent to W. We denote this by $\widetilde{W}$. Also, a *coordinate sublattice* of $\mathbb{Z}^n$ refers to a sublattice of the form $\{\mathbf{v} \in \mathbb{Z}^n : v_i = 0 \text{ for } i \in I\}$ for some nonzero subset I of $\{1, 2, \ldots, n\}$. (This is in analogy with a *coordinate plane*.)

Proposition 8.1.3. *Let $n > k \ge 2$. Let $W, V : \mathbb{Z}^n \to K$ be elliptic nets. Suppose that for every coordinate sublattice $L \subset \mathbb{Z}^n$ of rank k, $W|L$ and $V|L$ are scale equivalent. Then W and V are scale equivalent.*

Proof. We may assume without loss of generality that W and V are normalised, since normalising them will not change the condition that subnets are scale equivalent. We show that, once normalised, $W = V$.

First, normalising W and V automatically normalises each of the $W|L$ and $V|L$. For each L, since these are scale equivalent and normalised, they agree. That is, $W(\mathbf{v}) = V(\mathbf{v})$ for any vector $\mathbf{v}$ in any coordinate sublattice of rank k.

Theorem 4.3.3 tells us that an elliptic net on $\mathbb{Z}^n$ is uniquely determined by its values on the rank $n-1$ coordinate planes. Therefore W and V agree on all rank three coordinate planes since they agree on all rank two coordinate planes. By repeated application of Theorem 4.3.3 in this fashion, we eventually find that $W = V$ on all of $\mathbb{Z}^n$. □

8.2 Curves from nets of ranks 1 and 2

Recall that a change of variables of a cubic curve in Weierstrass form is said to be *unihomothetic* if it is of the form

$$\begin{aligned} x' &= x + r, \\ y' &= y + sx + t. \end{aligned} \tag{8.3}$$

Given an elliptic net of rank one or two, we can now describe explicitly how to obtain a curve from which it arises. The rank one case is due originally to Ward [74, Thm 12.1], but we provide Swart's version here.

Proposition 8.2.1 ([68, Thm 4.5.3]). *Let $W : \mathbb{Z} \to K$ be a normalised non-degenerate elliptic net. Then the family of curve-point pairs (C, P) such that $W = W_{C,P}$ is three dimensional. These are the curve and point*

$$C : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \qquad P = (0,0)$$

where

$$a_1 = \frac{W(4) + W(2)^5 - 2W(2)W(3)}{W(2)^2W(3)}$$

$$a_2 = \frac{W(2)W(3)^2 + W(4) + W(2)^5 - W(2)W(3)}{W(2)^3W(3)}$$

$$a_3 = W(2), \qquad a_4 = 1, \qquad a_6 = 0$$

or any image of these under a unihomothetic change of coordinates.

Proof. First, note that the division polynomials Ψ_1, Ψ_2, Ψ_3 and Ψ_4 are invariant under a change of coordinates of the form (8.3). Then, it is a simple calculation to check that $W_{C,P}$ agrees with W at the first four terms; hence $W_{C,P} = W$. Conversely, suppose $W = W_{C',P'}$. After applying a transformation of the form (8.3) taking P' to $(0,0)$ and taking a_4 to 1, substitution of the division polynomials into the equations above verifies that $a'_i = a_i$ for all i. □

Proposition 8.2.2. *Let $W : \mathbb{Z}^2 \to K$ be a normalised non-degenerate elliptic net. Then the family of 3-tuples (C, P_1, P_2) such that $W = W_{C,P_1,P_2}$ is three dimensional. These are the curve*

$$C : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

and points

$$P_1 = (0,0), \qquad P_2 = (W(2,1) - W(1,2), 0),$$

with

$$a_1 = \frac{W(2,0) - W(0,2)}{W(2,1) - W(1,2)}, \qquad a_2 = 2W(2,1) - W(1,2), \qquad a_3 = W(2,0)$$
$$a_4 = (W(2,1) - W(1,2))W(2,1), \qquad a_6 = 0$$

or any image of these under a unihomothetic change of coordinates.

Proof. The formulæ for $W(2,0)$, $W(0,2)$, $W(2,1)$ and $W(1,2)$ are invariant under a change of coordinates of the form (8.3). The net W_{C,P_1,P_2} agrees with W at the terms $(2,0)$, $(0,2)$, $(2,1)$ and $(1,2)$; hence $W_{C,P_1,P_2} = W$. Conversely, suppose $W = W_{C',P_1',P_2'}$. After applying a unihomothetic transformation taking P_1' to $(0,0)$ and P_2' to $(W(1,2) - W(2,1), 0)$, substitution of the net polynomials into the equations above verifies that $a_i' = a_i$ for all i. □

A more symmetric set of equations in the case of characteristic not equal to 2 is as follows:

$$P_1 = (v,0), \qquad P_2 = (-v,0), \qquad 2v = W(2,1) - W(1,2),$$
$$a_1 = \frac{W(2,0) - W(0,2)}{W(2,1) - W(1,2)}, \qquad 2a_2 = W(2,1) + W(1,2), \qquad 2a_3 = W(2,0) + W(0,2)$$
$$4a_4 = -(W(2,1) - W(1,2))^2, \qquad 8a_6 = -(W(2,1) - W(1,2))^2(W(2,1) + W(1,2))$$

8.3 Curves from nets of rank $n \geq 3$

Theorem 8.3.1. *Let $W : \mathbb{Z}^n \to K$ be a normalised non-degenerate elliptic net. Then the set of curves C defined over K and $\mathbf{P} \in C^n$ such that $W = W_{C,\mathbf{P}}$ forms a three-dimensional family of tuples $(C, \mathbf{P})$. In particular, the family consists of one such tuple and all its images under a unihomothetic change of coordinates.*

Proof. First we observe a useful consequence of Theorem 6.2.3. Suppose $V_1 : \mathbb{Z}^m \to K$ is an elliptic net of rank m associated to C and $\mathbf{P}$. Also suppose

$$V_2 : \{\mathbf{v} \in \mathbb{Z}^m : v_m = 0\} \to K$$

is the elliptic subnet of V_1 associated to the coordinate lattice of rank $m-1$ consisting of vectors with last coordinate zero. Suppose $V_2' : \mathbb{Z}^{m-1} \to K$ is naturally identified with V_2 by simply deleting the last coordinate of the domain. Then V_2' is associated to C and $\mathbf{P}'$ where $\mathbf{P}'$ is simply $\mathbf{P}$ with the last

coordinate deleted. This result holds equally well for any coordinate plane (not just the one with last coordinate zero).

The theorem holds for elliptic nets of rank 1 and 2 by Propositions 8.2.1 and 8.2.2. We demonstrate the statement for higher ranks by induction. Suppose $n \geq 3$ and the theorem holds for all nets of rank less than n. Let $W_1, W_2, \ldots, W_n$ be the normalised elliptic subnets of W associated to the rank $n-1$ coordinate lattices $L_i = \{\mathbf{v} : v_i = 0\}$. These are defined as nets $W_i : L_i \to K$; they can be identified with nets $W_i' : \mathbb{Z}^{n-1} \to K$ in an obvious way. They are non-degenerate. Then, by the inductive hypothesis, we have $W_i' = W_{C_i,\mathbf{P}_i}$ for some curves C_i and points $\mathbf{P}_i \in C_i^{n-1}$.

Consider two such nets, W_i and W_j (where $i < j$). Let $W_{ij} = W_i \cap W_j$ in W. Define $W_{ij}' : \mathbb{Z}^{n-2} \to K$ by the obvious identification. Then, $W_{ij}' = W_{C_{ij},\mathbf{P}_{ij}}$ for some curve C_{ij} and $\mathbf{P}_{ij} \in C_{ij}^{n-2}$. By the foregoing, $C_i = C_j = C_{ij}$, $\mathbf{P}_{ij}$ is just $\mathbf{P}_j$ with the i-th coordinate deleted, and $\mathbf{P}_{ij}$ is just $\mathbf{P}_i$ with the $(j-1)$-th coordinate deleted.

Considering every such pair, we may define a candidate curve C by $C = C_i$ for all i and $\mathbf{P} \in C^n$ defined as the unique n-tuple which gives each $\mathbf{P}_i$ by deleting the i-th coordinate. By the foregoing, this is well-defined.

Now we see that W agrees with $W_{C,\mathbf{P}}$ on all coordinate sublattices of rank $n-1$ and hence by Proposition 8.1.3, $W = W_{C,\mathbf{P}}$.

If we apply a change of coordinates of the form (8.3) to C, the elliptic net does not change since it is determined by its values on the 2-dimensional coordinate planes (by induction using Theorem 4.3.3; see the last paragraph of the proof of Proposition 8.1.3). Furthermore, if two tuples *not* related by such a change of coordinates generate the same net W, then the same would be true for the rank-two subnets – a contradiction. □

Chapter 9

The curve-net theorem

After a few preliminaries, we will now state the bijection between elliptic nets and curve-point tuples– the famed 'curve-net theorem.'

9.1 Homothety and singular nets

The only changes of coordinates of a Weierstrass equation into another are compositions of unihomothetic changes of coordinates and changes of coordinates of the form $(x,y) \mapsto (\lambda^2 x, \lambda^3 y)$, which we refer to as *homotheties*.

Proposition 9.1.1. *Consider the rank n elliptic net $W_{C,\mathbf{P}}$ associated to*

$$C : y^2 + a_1 xy + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6$$

and $\mathbf{P} \in C^n$. Let λ be a non-zero element of K. Suppose $\phi_\lambda : C \to C_\lambda$ is the isomorphism of curves taking C to

$$C_\lambda : y^2 + \lambda a_1 xy + \lambda^3 a_3 y = x^3 + \lambda^2 a_2 x^2 + \lambda^4 a_4 x + \lambda^6 a_6$$

under the change of coordinates $(x,y) \mapsto (\lambda^2 x, \lambda^3 y)$. Then

$$\widetilde{W}_{C_\lambda, \phi_\lambda(\mathbf{P})} = \lambda \widetilde{W}_{C,\mathbf{P}}$$

In particular, let δ_{ij} be the Kronecker delta, and define

$$g(\mathbf{v}) = -1 - \sum_{1 \le i < j \le n} (-1)^{\delta_{ij}} v_i v_j.$$

Then

$$W_{C_\lambda, \phi_\lambda(\mathbf{P})} = \lambda^{g(\mathbf{v})} W_{C,\mathbf{P}}.$$

Proof. If λ is a non-zero element of K then C_λ is again an elliptic curve. The proposition holds for $\Omega_{\mathbf{v}}$ over $\mathbb{C}$ by Definition 5.1.1. Therefore the rational function representations of Theorem 7.1.1 are weighted homogeneous in the appropriate way; hence it holds over any field. □

Therefore we set the following definition

Definition 9.1.1. If a basis $\mathcal{B} : b_1, \ldots, b_n$ is specified for A, then with the notation of Proposition 9.1.1, we define

$$W^{\lambda}(\mathbf{v}) = \lambda^{g(\mathbf{v})} W(\mathbf{v}).$$

This gives an action of K on elliptic nets $W : \mathbb{Z}^n \to K$ called the *homothety action*. Two elliptic nets are *homothetic* if they are in the same orbit of the action of K.

Proposition 9.1.2. *Let W be an elliptic net. Then for any non-zero $\lambda \in K$, W^{λ} is normalised if and only if W is.*

Let $W : \mathbb{Z}^n \to K$ be an elliptic net. If the curve C associated to its normalisation is a nodal or cuspidal cubic, then W is called *singular*. If, instead, C is an elliptic curve, then W is called *non-singular*. In either case, the discriminant Δ of W is defined to be the discriminant of the associated Weierstrass equation. Similarly, the j-invariant of a non-singular elliptic net is the j-invariant of the associated Weierstrass equation. These are well-defined since the discriminant and j-invariant do not change under unihomothetic changes of variables. The discriminant of an elliptic net changes by a factor of λ^{12} under homothety, while the j-invariant remains unaltered.

Both scale equivalence and multiplication by a constant take an elliptic net to another elliptic net. Therefore we will define the slightly more general notion of *equivalence* as any combination of the two.

Definition 9.1.2. Let W_1 and W_2 be elliptic nets. Suppose $\alpha, \beta \in K^*$, and $f : A \to \mathbb{Z}$ is a quadratic form. If

$$W_1(\mathbf{v}) = \alpha \beta^{f(\mathbf{v})} W_2(\mathbf{v})$$

for all $\mathbf{v}$, then we say W_1 *is equivalent to* W_2 and write $W_1 \sim W_2$.

9.2 The curve-net theorem

For any Weierstrass curve C, we may put a partial ordering on the tuples of points of C by $(P_1, \ldots, P_n) \leq (Q_1, \ldots, Q_m)$ if the groups they generate satisfy a containment $\langle P_1, \ldots, P_n \rangle \subseteq \langle Q_1, \ldots, Q_m \rangle$. The collection of all elliptic nets is ordered by the subnet relation.

Collecting our work up to this point, we have shown:

Theorem 9.2.1. *We call a set of points $\{P_1, \ldots, P_n\}$ on the non-singular part C_0 of a cubic curve* appropriate *if $P_i \neq \pm P_j$ for any $i \neq j$ and if $[2]P_1$ and $[3]P_1$ are nonzero in the case $n = 1$. For each field K, there is an explicit isomorphism of partially ordered sets*

$$\left\{ \begin{array}{l} \textit{scale equivalence classes of} \\ \textit{non-degenerate elliptic nets} \\ W : \mathbb{Z}^n \to K \textit{ for some } n \end{array} \right\} \longleftrightarrow \left\{ \begin{array}{l} \textit{tuples } (C, P_1, \ldots, P_m) \textit{ for some } m, \textit{ where } C \\ \textit{is a cubic curve in Weierstrass form over } K, \\ \textit{considered modulo unihomothetic changes} \\ \textit{of variables, and such that } \{P_i\} \in C_0(K)^m \\ \textit{is appropriate} \end{array} \right\}.$$

Non-singular nets correspond to elliptic curves. The action of K (by homothety) on the sets preserves the order and respects the isomorphism. The bijection takes an elliptic net of rank n to a tuple with n points. The elliptic net W associated to a tuple $(C, P_1, \ldots, P_n)$ *satisfies the property that* $W(v_1, \ldots, v_n) = 0$ *if and only if* $v_1 P_1 + \ldots + v_n P_n = 0$ *on the curve* C.

Proof. See Theorem 7.1.1, Definition 7.2.1, Theorem 8.3.1, Corollary 7.2.1 and Proposition 9.1.1 □

Chapter 10

Bases and periodicity

This centerpiece to this chapter is a notion of basis change for elliptic nets. This is explained in the first section. As a consequence, we derive partial periodicity properties generalising those of Ward, and examine quantities which are invariant under change of basis. A warning: in this section (and henceforth), we will use the terms 'basis' and 'coordinate' in an arbitrary abelian group. In this context, coordinates of a point with respect to a basis are no longer unique. And bases are not required to span, so sometimes coordinates for a point do not exist.

10.1 Freedom from the tyranny of bases

Let E be an elliptic curve and P a point on E. Consider two elliptic nets: $W_{E,P}$ and $W_{E,[2]P}$. Not unreasonably, we would like to consider the second a 'subnet' of the first. However, they do not satisfy the definition of a subnet relationship, since they are both defined on $\mathbb{Z}$ and do not, in general, agree anywhere. Of course, we can say that the elliptic net

$$W' : 2\mathbb{Z} \to K, \qquad W'(2n) = W_{E,P}(2n)$$

is a subnet of $W_{E,P}$ corresponding to the inclusion $2\mathbb{Z} \subset \mathbb{Z}$. But where has $W_{E,[2]P}$ gone in this statement? The sequences

$$W_{E,P}(2n), \quad \text{and} \quad W_{E,[2]P}(n)$$

should in some sense both be related to the sequence of multiples $[2n]P$, so we expect them to relate to one another. In fact, it turns out that

$$W'(2n) = W_{E,P}(2n) = \frac{W_{E,[2]P}(n)}{W_{E,P}(2)^{n^2}}. \tag{10.1}$$

The problem we face here is to develop a language appropriate to 'changing the basis' of an elliptic net. In fact, we already have the answer, in the form of an earlier theorem, which we restate here.

Theorem 10.1.1 (Restatement of Theorem 6.2.3). *Let T be any $n \times m$ matrix. Let $\mathbf{P} \in E^m$, $\mathbf{v} \in \mathbb{Z}^n$. Then*

$$W_{E,\mathbf{P}}(T^{tr}(\mathbf{v})) = W_{E,T(\mathbf{P})}(\mathbf{v}) \prod_{i=1}^{n} W_{E,\mathbf{P}}(T^{tr}(\mathbf{e}_i))^{v_i^2 - v_i(\Sigma_{j \neq i} v_j)} \prod_{1 \leq i < j \leq n} W_{E,\mathbf{P}}(T^{tr}(\mathbf{e}_i + \mathbf{e}_j))^{v_i v_j} \quad (10.2)$$

In particular, the two elliptic nets

$$W_{E,\mathbf{P}} \circ T^{tr} : \mathbb{Z}^n \to K, \quad \text{and} \quad W_{E,T(\mathbf{P})} : \mathbb{Z}^n \to K$$

are scale equivalent.

The equation (10.1) is a corollary.

10.2 Higher rank periodicity properties

Look back to Morgan Ward's 'periodicity property' for elliptic divisibility sequences (Theorem 2.6.2), which we restate here:

Theorem 10.2.1 ([74, Thm 9.2], Restatement of Theorem 2.6.2). *Let W be an integer elliptic divisibility sequence such that $W(1) = 1$ and $W(2) | W(4)$. Let p be an odd prime and suppose $W(2)W(3) \not\equiv 0 \mod p$. Let r be the rank of apparition of W with respect to p. Then there exist integers a, b such that for all non-negative integers k and s, we have*

$$W(sr+k) \equiv a^{ks} b^{s^2} W(k) \mod p.$$

Furthermore, the integers a and b satisfy

$$a \equiv \frac{W(r-2)}{W(r-1)W(2)}, \qquad b \equiv \frac{W(r-1)^2 W(2)}{W(r-2)} \mod p.$$

Similar periodicity properties for prime power moduli, and their properties, have been extensively studied by Ayad [2] and especially Swart [68].

Definition 10.2.1. The zeroes of an elliptic divisibility sequence or elliptic net appear as a sublattice of the lattice of indices. We call this sublattice the *lattice of zero-apparition*. In the case of a sequence, this sublattice is specified by a single positive integer, equal to the smallest positive index of a vanishing term, and this number is called the *rank of zero-apparition*.

The rank of zero-apparition of an elliptic divisibility sequence associated to a point P will equal the order of the point P. In the case of an array associated to points $P_1, \ldots, P_n$, the elements $\mathbf{v} = (v_1, \ldots, v_n)$ of the lattice of zero-apparition correspond to linear combinations $\mathbf{v} \cdot \mathbf{P}$ that vanish. Although the zeroes in an elliptic divisibility sequence appear regularly at a specific interval, that interval is not always a period for the sequence. An elliptic net is not necessarily periodic with respect to its lattice of zero-apparition.

In this section we use Theorem 10.1.1 to prove periodicity properties for general n. First we state and prove Ward's result for rank 1 to illustrate the method.

Theorem 10.2.2 (Generalisation of Theorem 2.6.2). *Suppose that $W_{E,P}(r) = 0$ for a non-degenerate elliptic divisibility sequence. Then for all $s, k \in \mathbb{Z}$, we have*

$$W_{E,P}(sr+k) = W_{E,P}(k)a^{sk}b^{s^2} \tag{10.3}$$

where

$$a = \frac{W_{E,P}(r+2)}{W_{E,P}(r+1)W_{E,P}(2)}, \qquad b = \frac{W_{E,P}(r+1)^2 W_{E,P}(2)}{W_{E,P}(r+2)} \tag{10.4}$$

Furthermore, $a^r = b^2$. Therefore, there exists an $\alpha \in \bar{K}$, the algebraic closure of K, such that $\alpha^2 = a$ and $\alpha^r = b$, and so

$$W_{E,P}(sr+k) = W_{E,P}(k)\alpha^{(sr+k)^2-k^2}.$$

Proof. The first equation was first proven by Morgan Ward in the case of $K = \mathbb{Q}$ [74, Thm. 8.1]. We prove it here from Theorem 10.1.1. We use (10.2) with

$$T = \begin{pmatrix} r+2 \\ 1 \end{pmatrix}.$$

We obtain

$$W_{E,([r+2]P,P)}(s,t)W_{E,P}(r+2)^{s^2-st}W_{E,P}(r+3)^{st}W_{E,P}(1)^{t^2-st} = W_{E,P}(sr+2s+t).$$

Instead, using

$$T = \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

we obtain

$$W_{E,([2]P,P)}(s,t)W_{E,P}(2)^{s^2-st}W_{E,P}(3)^{st}W_{E,P}(1)^{t^2-st} = W_{E,P}(2s+t).$$

Recalling that $W_{E,P}(1) = 1$ and $[r+2]P = [2]P$, and setting $t = k-2s$, these combine to give

$$W_{E,P}(sr+k) = W_{E,P}(k)a^{sk}b^{s^2}$$

for some a, b independent of s and k. This does not require dividing by zero by the non-degeneracy hypothesis.

Finally, the expressions for a and b given in the statement of the theorem may be derived from this equation with $s = 1, k = 1, 2$. Finally, since a and b are nonzero, choose some k so that $W_{E,P}(k) \neq 0$ and we may calculate

$$W_{E,P}(k)a^{2k}b^4 = W_{E,P}(2r+k) = W_{E,P}(r+(r+k)) = W_{E,P}(r+k)a^{r+k}b = W_{E,P}(k)a^{r+2k}b^2$$

from which $a^r = b^2$. □

The proof for the general case works along much the same lines.

Theorem 10.2.3. *Let K be a field. Suppose that $W_{E,\mathbf{P}} : \mathbb{Z}^n \to K$ is a non-degenerate elliptic net, with lattice of zero-apparition Γ. For any $\mathbf{r} \in \Gamma$ and $\mathbf{k} \in \mathbb{Z}^n \backslash \Gamma$, define $g : \Gamma \times (\mathbb{Z}^n \backslash \Gamma) \to K^*$ by*

$$g(\mathbf{r}, \mathbf{k}) = \frac{W_{E,\mathbf{P}}(\mathbf{r}+\mathbf{k})}{W_{E,\mathbf{P}}(\mathbf{k})}. \tag{10.5}$$

Then g is a quadratic function where defined, which is affine linear in the second factor in the sense that

$$g(\mathbf{r}, \mathbf{k}_1 + \mathbf{k}_2) - g(\mathbf{r}, \mathbf{k}_1) - g(\mathbf{r}, \mathbf{k}_2) + g(\mathbf{r}, \mathbf{0}) = 0.$$

Proof. Since this proof is rather complicated, we lay out the steps here as a more detailed statement of the above:

1. Suppose that $W_{E,\mathbf{P}}$ is a non-degenerate elliptic net of rank n. Suppose that $W_{E,\mathbf{P}}(\mathbf{r}) = 0$. Then for any $\mathbf{k} \in \mathbb{Z}^n \backslash \Gamma$, we have
$$W_{E,\mathbf{P}}(l\mathbf{r} + \mathbf{k}) = W_{E,\mathbf{P}}(\mathbf{k}) f_{\mathbf{r}}(l, \mathbf{k}) \tag{10.6}$$
where $f_{\mathbf{r}} : \mathbb{Z}^{n+1} \to K^*$ is a quadratic form.

2. Furthermore, f satisfies
$$f_{\mathbf{r}}(l, \mathbf{k}) = \prod_{j=0}^{l-1} f_{\mathbf{r}}(1, \mathbf{k} + j\mathbf{r}). \tag{10.7}$$

3. There exists a quadratic function $q_{\mathbf{r}} : \mathbb{Z}^n \to K^*$ such that
$$f_{\mathbf{r}}(l, \mathbf{k}) = q_{\mathbf{r}}(l\mathbf{r} + \mathbf{k}) q_{\mathbf{r}}(\mathbf{k})^{-1}. \tag{10.8}$$

4. In particular, $f_{\mathbf{r}}$ is affine linear in the second factor, in the sense that
$$f_{\mathbf{r}}(l, \mathbf{k}_1 + \mathbf{k}_2) - f_{\mathbf{r}}(l, \mathbf{k}_1) - f_{\mathbf{r}}(l, \mathbf{k}_2) + f_{\mathbf{r}}(l, \mathbf{0}) = 0.$$

5. Finally, for different $\mathbf{r}_1$ and $\mathbf{r}_2$ we have
$$f_{\mathbf{r}_1 + \mathbf{r}_2}(l, \mathbf{k}) = f_{\mathbf{r}_1}(l, \mathbf{k}) f_{\mathbf{r}_2}(l, \mathbf{k} + \mathbf{r}_1). \tag{10.9}$$

6. Therefore we may define $g : \Gamma \times \mathbb{Z}^n \to K^*$ which is quadratic, and affine linear in the second coordinate by
$$g(\mathbf{r}, \mathbf{k}) = f_{\mathbf{r}}(1, \mathbf{k}).$$

We begin with the first statement, (10.6). Suppose that $\mathbf{r} = (r_1, \ldots, r_n)$. Let $\mathbf{s} = (s_1, \ldots, s_n) \in \mathbb{Z}^n$ such that $\mathbf{s} \cdot \mathbf{P}$ is not of the form P_i, or $P_i \pm P_j$ for any i and j. Then, we apply Theorem 10.1.1 with the $(n+1) \times n$ matrices

$$T_1 = \begin{pmatrix} s_1 & s_2 & \cdots & s_n \\ 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix}, \quad \text{and} \quad T_2 = \begin{pmatrix} s_1 + r_1 & s_2 + r_2 & \cdots & s_n + r_n \\ 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix}.$$

The condition on $\mathbf{s}$ ensures that $W_{E,T_1(\mathbf{P})}$ and $W_{E,T_2(\mathbf{P})}$ are non-degenerate elliptic nets. We obtain

$$W_{E,\mathbf{P}}(k\mathbf{s}+l_1\mathbf{e}_1+\cdots+l_n\mathbf{e}_n) = W_{E,T_1(\mathbf{P})}(k,l_1,\ldots,l_n)W_{E,\mathbf{P}}(\mathbf{s})^{k^2-\Sigma_{i=1}^n kl_i} \times \prod_{i=1}^n W_{E,\mathbf{P}}(\mathbf{e}_i)^{l_i^2-kl_i-\Sigma_{i\neq j}l_il_j} \prod_{i=1}^n W_{E,\mathbf{P}}(\mathbf{s}+\mathbf{e}_i)^{kl_i} \prod_{1\leq i<j\leq n} W_{E,\mathbf{P}}(\mathbf{e}_i+\mathbf{e}_j)^{l_il_j}$$

and

$$W_{E,\mathbf{P}}(k\mathbf{r}+k\mathbf{s}+l_1\mathbf{e}_1+\cdots+l_n\mathbf{e}_n) = W_{E,T_2(\mathbf{P})}(k,l_1,\ldots,l_n)W_{E,\mathbf{P}}(\mathbf{r}+\mathbf{s})^{k^2-\Sigma_{i=1}^n kl_i} \times \prod_{i=1}^n W_{E,\mathbf{P}}(\mathbf{e}_i)^{l_i^2-kl_i-\Sigma_{i\neq j}l_il_j} \prod_{i=1}^n W_{E,\mathbf{P}}(\mathbf{r}+\mathbf{s}+\mathbf{e}_i)^{kl_i} \prod_{1\leq i<j\leq n} W_{E,\mathbf{P}}(\mathbf{e}_i+\mathbf{e}_j)^{l_il_j}.$$

Notice that $T_1(\mathbf{P}) = T_2(\mathbf{P})$ so we can combine the above to obtain

$$W_{E,\mathbf{P}}(k\mathbf{r}+k\mathbf{s}+l_1\mathbf{e}_1+\cdots+l_n\mathbf{e}_n) = W_{E,\mathbf{P}}(k\mathbf{s}+l_1\mathbf{e}_1+\cdots+l_n\mathbf{e}_n) \times W_{E,\mathbf{P}}(\mathbf{r}+\mathbf{s})^{k^2-\Sigma_{i=1}^n kl_i} \prod_{i=1}^n W_{E,\mathbf{P}}(\mathbf{r}+\mathbf{s}+\mathbf{e}_i)^{kl_i} \left(W_{E,\mathbf{P}}(\mathbf{s})^{k^2-\Sigma_{i=1}^n kl_i} \prod_{i=1}^n W_{E,\mathbf{P}}(\mathbf{s}+\mathbf{e}_i)^{kl_i} \right)^{-1}$$

For any k and $\mathbf{k}$, the values $l_1,\ldots,l_n$ can be chosen so that

$$\mathbf{k} = k\mathbf{s}+l_1\mathbf{e}_1+\ldots l_n\mathbf{e}_n.$$

Hence this proves the first statement, Item 1. We show the second statement, Item 2, by induction. Suppose it holds for $l < N$. Then we have

$$\begin{aligned} f_r(N,\mathbf{k}) &= \frac{W_{E,\mathbf{P}}(N\mathbf{r}+\mathbf{k})}{W_{E,\mathbf{P}}(\mathbf{k})} \\ &= \left(\frac{W_{E,\mathbf{P}}(N\mathbf{r}+\mathbf{k})W_{E,\mathbf{P}}}{W_{E,\mathbf{P}}((N-1)\mathbf{r}+\mathbf{k})}\right)\left(\frac{W_{E,\mathbf{P}}((N-1)\mathbf{r}+\mathbf{k})}{W_{E,\mathbf{P}}(\mathbf{k})}\right) \\ &= f_r(1,(N-1)\mathbf{r}+\mathbf{k})f_r(N-1,\mathbf{k}) \\ &= \prod_{j=1}^{N-1} f_r(1,\mathbf{k}+j\mathbf{r}). \end{aligned}$$

For Item 3, chose $\mathbf{u}\in\mathbb{Z}^n$ and set $c = \mathbf{u}\cdot\mathbf{r}$. Then, let $U:\mathbb{Z}^n\to\mathbb{Z}^n$ be the linear transformation $U(\mathbf{v}) = c\mathbf{v}-(\mathbf{u}\cdot\mathbf{v})\mathbf{r}$. This transformation is such that $U(\mathbf{r}) = 0$. Define the quadratic function

$$q_0(\mathbf{v}) = f_{\mathbf{r}}(\mathbf{u}\cdot\mathbf{v}, U(\mathbf{v})).$$

Then,

$$\frac{q_0(l\mathbf{r}+\mathbf{k})}{q_0(\mathbf{k})} = \frac{f_{\mathbf{r}}(\mathbf{u}\cdot\mathbf{k}+cl, c\mathbf{k}-(\mathbf{u}\cdot\mathbf{k})\mathbf{r})}{f_{\mathbf{r}}(\mathbf{u}\cdot\mathbf{k}, c\mathbf{k}-(\mathbf{u}\cdot\mathbf{k})\mathbf{r})} = \prod_{j=\mathbf{u}\cdot\mathbf{k}}^{cl+\mathbf{u}\cdot\mathbf{k}-1} f_{\mathbf{r}}(1, c\mathbf{k}-(\mathbf{u}\cdot\mathbf{k})\mathbf{r}+j\mathbf{r}) = f_{\mathbf{r}}(cl, c\mathbf{k}).$$

Now define another function on elements of $c\mathbb{Z}^n$ by

$$q_{\mathbf{r}}(c\mathbf{v}) = q_0(\mathbf{v}).$$

This is a quadratic function and by interpolation we extend the domain of definition to $\mathbb{Z}^n$. Since $f_{\mathbf{r}}(cl, c\mathbf{k})$ is also a quadratic function, it can be interpolated also, and we still have

$$f_{\mathbf{r}}(l,\mathbf{k}) = q_{\mathbf{r}}(l\mathbf{r}+\mathbf{k})q_{\mathbf{r}}(\mathbf{k})^{-1},$$

from which we deduce affine linearity, Item 4.

The formula (10.9) in Item 5 is immediate, and from this and affine linearity, the expression

$$\frac{g(\mathbf{r}_1+\mathbf{r}_2+\mathbf{r}_3,\mathbf{k}_1+\mathbf{k}_2+\mathbf{k}_3)g(\mathbf{r}_1,\mathbf{k}_1)g(\mathbf{r}_2,\mathbf{k}_2)g(\mathbf{r}_3,\mathbf{k}_3)}{g(\mathbf{r}_1+\mathbf{r}_2,\mathbf{k}_1+\mathbf{k}_2)g(\mathbf{r}_2+\mathbf{r}_3,\mathbf{k}_2+\mathbf{k}_3)g(\mathbf{r}_1+\mathbf{r}_3,\mathbf{k}_1+\mathbf{k}_3)}$$

becomes unity, which completes the proof. □

Note that the interpolation of $q_{\mathbf{r}}$ will in general require enlarging the field of definition of the coefficients of the quadratic form (here 'coefficients' is interpreted in the multiplicative sense). This is analogous to the one-dimensional case where it was necessary to move to a quadratic extension of K to define α in the statement of Theorem 10.2.2.

Corollary 10.2.4. *For every elliptic net $W : \mathbb{Z}^n \to K$, and $\mathbf{r} \in \mathbb{Z}^n$ such that $W(\mathbf{r}) = 0$, there exists an equivalent elliptic net $W_{\mathbf{r}} : \mathbb{Z}^n \to \bar{K}$ which is periodic with respect to $\mathbf{r}$.*

Proof. Combining (10.6) with (10.8) shows that the elliptic net

$$\frac{W_{E,\mathbf{P}}(\mathbf{v})}{q_{\mathbf{r}}(\mathbf{v})}$$

is periodic with respect to $\mathbf{r}$. □

We state a lemma for the rank 2 case, both as an example, and since it will be useful later.

Lemma 10.2.5. *Let $P, Q \in E$ and $W_{E,P,Q}$ be the associated elliptic net. Let $W_{E,P,Q}(\mathbf{r}) = 0$ for some $\mathbf{r} = (r_1, r_2)$. We have the form*

$$g(l\mathbf{r}, k_1, k_2) = a_{\mathbf{r}}^{lk_1} b_{\mathbf{r}}^{lk_2} c_{\mathbf{r}}^{l^2} \tag{10.10}$$

where

$$a_{\mathbf{r}} = \frac{W_{E,P,Q}(r_1+2, r_2)}{W_{E,P,Q}(r_1+1, r_2)W_{E,P,Q}(2,0)}, \quad b_{\mathbf{r}} = \frac{W_{E,P,Q}(r_1, r_2+2)}{W_{E,P,Q}(r_1, r_2+1)W_{E,P,Q}(0,2)}, \tag{10.11}$$

$$c_{\mathbf{r}} = \frac{W_{E,P,Q}(r_1+1, r_2+1)}{a_{\mathbf{r}} b_{\mathbf{r}} W_{E,P,Q}(1,1)}. \tag{10.12}$$

Proof. The function g is quadratic, but affine linear in the second argument. Thus, it has the form

$$g(l\mathbf{r}, k_1, k_2) = a_{\mathbf{r}}^{lk_1} b_{\mathbf{r}}^{lk_2} c_{\mathbf{r}}^{l^2}$$

for some $a_{\mathbf{r}}$, $b_{\mathbf{r}}$ and $c_{\mathbf{r}}$. Collect the equations (10.5) for $\mathbf{r}$ and each of the vectors

$$\mathbf{k} = (2,0), (1,0), (0,2), (0,1), (1,1).$$

By linear algebra,

$$a_{\mathbf{r}} = \frac{g(\mathbf{r},2,0)}{g(\mathbf{r},1,0)} = \frac{W_{E,P,Q}(r_1+2,r_2)}{W_{E,P,Q}(r_1+1,r_2)W_{E,P,Q}(2,0)},$$
$$b_{\mathbf{r}} = \frac{g(\mathbf{r},0,2)}{g(\mathbf{r},0,1)} = \frac{W_{E,P,Q}(r_1,r_2+2)}{W_{E,P,Q}(r_1,r_2+1)W_{E,P,Q}(0,2)},$$
$$c_{\mathbf{r}} = \frac{g(\mathbf{r},1,1)}{a_{\mathbf{r}}b_{\mathbf{r}}} = \frac{W_{E,P,Q}(r_1+1,r_2+1)}{a_{\mathbf{r}}b_{\mathbf{r}}W_{E,P,Q}(1,1)}.$$

□

10.3 Quantities which do not depend on basis

In light of Theorem 10.1.1, we would like to define a notation for expressions in terms of elliptic nets which can be evaluated under any choice of basis. To see what is meant, let Q be the point $[2]P$ on E. Consider the (as yet undefined) notation

$$\frac{\mathcal{W}(3q)}{\mathcal{W}(q)^9}. \tag{10.13}$$

By this notation we shall mean that the reader should perform the following steps: choose any elliptic net W associated to E and a basis $\mathbf{T} = P_1, \ldots, P_n \in E$; find some 'coordinates' of Q in this basis, say $\mathbf{v}_Q$ (that is to say, such that $\mathbf{v}_Q \cdot \mathbf{T} = Q$); and finally take the quotient of the evaluations of W at $3\mathbf{v}_Q$ and $\mathbf{v}_Q$ respectively to the powers shown. For example, if we choose the elliptic net $W_{E,P}$ associated to the basis P, then a coordinate $\mathbf{v}_Q$ of Q is 2 (since $Q = [2](P)$), and we obtain

$$\frac{W_{E,P}(6)}{W_{E,P}(2)^9},$$

whereas if we choose the elliptic net $W_{E,[2]P}$ associated to basis $[2]P$, a coordinate of Q is 1 (since $Q = [1]([2]P)$, and we obtain

$$\frac{W_{E,[2]P}(3)}{W_{E,[2]P}(1)^9}.$$

The punchline is that *these expressions are equal.* This is a result of Theorem 10.1.1 (or equation (10.1)) and fortuitous cancellation:

$$\frac{W_{E,P}(6)}{W_{E,P}(2)^9} = \left(\frac{W_{E,[2]P}(3)}{W_{E,P}(2)^9}\right) \div \left(\frac{W_{E,[2]P}(1)}{W_{E,P}(2)^1}\right)^9 = \frac{W_{E,[2]P}(3)}{W_{E,[2]P}(1)^9}.$$

In general, given any choices of bases in which Q has a coordinate, the resulting values will always be equal. This is only true because the expression (10.13) has a special quadratic shape (it would not work if we replaced the '9' with an '8'). When the value of such an expression is independent of the choice of elliptic net used for its evaluation, we will say that such an expression is *well-defined*, and sometimes say that it is a *quadratic quantity.*

There is a small but very important catch. If we wish to evaluate $\mathcal{W}(p)$, we need to make *two* choices: the basis $\mathbf{T}$ for the elliptic net, and the coordinate $\mathbf{v}_P$ of P with respect to this net. Suppose for the sake of argument that we choose two bases $\mathbf{T}_1$ and $\mathbf{T}_2$ such that $M(\mathbf{T}_1) = \mathbf{T}_2$ for some linear transformation M. Then, Theorem 10.1.1 tells us that the the evaluation of $\mathcal{W}(p)$ performed with these two choices will agree up to equivalence *when using two coordinates* $\mathbf{v}_1$ *and* $\mathbf{v}_2$ *for* P *which satisfy* $M(\mathbf{v}_1) = \mathbf{v}_2$. There may be, in general, other ways to choose coordinates. It is Theorem 10.1.1 which we will use to demonstrate the invariance of quadratic quantities, so we must pay special attention to this matter. The partial periodicity results of the last section will be very important.

Following this informal discussion, we now formalise the notion with a few definitions and a theorem.

Definition 10.3.1. Let the free group on $\mathrm{Div}(E)$ be denoted $\mathrm{Div}^2(E)$. Elements will be denoted

$$\sum_{\substack{\text{divisors} \\ D=\sum_P n_{D,P}(P)}} m_D[D] \tag{10.14}$$

Example elements and calculations are

$$[(P)-(Q)]+3[2(P)-5([4]Q)], \qquad [(P)], \qquad [(P)+(Q)]-2[(P)+(Q)] = -[(P)+(Q)].$$

Definition 10.3.2. We define a subgroup $\mathrm{Quad}(E) \subset \mathrm{Div}^2(E)$ given by elements (called *quadratic*) of the form (10.14) satisfying

$$\sum_{\substack{\text{divisors} \\ D=\sum_P n_{D,P}(P)}} m_D \left(\sum_P n_{D,P} x_P \right)^2 = 0$$

as a polynomial identity in all the independent variables x_P which appear.

Just as the group of principal divisors is generated by divisors of the form

$$(P)+(Q)-(P+Q)-(\mathcal{O}),$$

so the group $\mathrm{Quad}(E)$ is generated by elements of the form

$$[(P)+(Q)+(R)+(S)]-[(P)+(Q)+(S)]-[(P)+(R)+(S)]-[(Q)+(R)+(S)] \\ +[(P)+(S)]+[(Q)+(S)]+[(R)+(S)]-[(S)].$$

So, in particular, an element of $\mathrm{Quad}(E)$ also satisfies

$$\sum_{\substack{\text{divisors} \\ D=\sum_P n_{D,P}(P)}} m_D \left(\sum_P n_{D,P} x_P \right) = 0.$$

The following depends upon Theorem 10.1.1.

Theorem 10.3.1. *Let*

$$\Theta = \sum_{\substack{divisors \\ D=\sum_P n_{D,P}(P)}} m_D[D]$$

be an element of $\mathrm{Quad}(E)$*, and such that none of the divisors D has sum $\mathcal{O}$, i.e., $\sum_P [n_{D,P}]P \neq \mathcal{O}$. Let $\mathbf{T} \in E(K)^n$ such that every P appearing in Θ is in the group $\Gamma_{\mathbf{T}} = \langle T_1, \ldots, T_n \rangle$ generated by the collection of T_i. For each such P, let $\mathbf{v}_{\mathbf{T},P} \in \mathbb{Z}^n$ such that $\mathbf{v}_{\mathbf{T},P} \cdot \mathbf{T} = P$ (i.e., $\mathbf{v}_{\mathbf{T},P}$ are the 'coordinates' of P in terms of $\mathbf{T}$). Then, the value*

$$\aleph = \prod_{\substack{divisors \\ D=\sum_P n_{D,P}(P)}} W_{E,\mathbf{T}}\left(\sum_P n_{D,P}\mathbf{v}_{\mathbf{T},P}\right)^{m_D} \tag{10.15}$$

in K^* is independent of the choice of basis $\mathbf{T}$ and the choice of coordinates in that basis.

Proof. The fact that the divisors D do not have sum zero guarantees that the values

$$W_{E,\mathbf{T}}\left(\sum_P n_{D,P}\mathbf{v}_P\right)$$

are each in K^*.

First, we will show independence of the choice of coordinates given a single choice of basis $\mathbf{T}$.

By Theorem 10.2.3, $g(\mathbf{r},\mathbf{k})$ is affine linear in the second coordinate where defined. Suppose we wish to compare the calculation of $\aleph$ using the same basis, but two different sets of coordinates: $\mathbf{v}_{\mathbf{T},P}$ and $\mathbf{u}_{\mathbf{T},P}$. These coordinates differ by $\mathbf{m}_{\mathbf{T},P} = \mathbf{v}_{\mathbf{T},P} - \mathbf{u}_{\mathbf{T},P}$ such that $W_{E,\mathbf{T}}(\mathbf{m}_{\mathbf{T},P}) = 0$. Then, the quotient of the two calculations will be of the form

$$\begin{aligned}\prod_{\substack{\text{divisors} \\ D=\sum_P n_{D,P}(P)}} \left(\frac{W_{E,\mathbf{T}}\left(\sum_P n_{D,P}\mathbf{v}_{\mathbf{T},P}\right)}{W_{E,\mathbf{T}}\left(\sum_P n_{D,P}\mathbf{u}_{\mathbf{T},P}\right)}\right)^{m_D} &= \prod_{\substack{\text{divisors} \\ D=\sum_P n_{D,P}(P)}} \left(g\left(\sum_P n_{D,P}\mathbf{m}_{\mathbf{T},P}, \sum_P n_{D,P}\mathbf{u}_{\mathbf{T},P}\right)\right)^{m_D} \\ &= \prod_{\substack{\text{divisors} \\ D=\sum_P n_{D,P}(P)}} \left(\prod_P g\left(\sum_P n_{D,P}\mathbf{m}_{\mathbf{T},P}, \mathbf{u}_{\mathbf{T},P}\right)^{n_D}\right)^{m_D} = 1\end{aligned}$$

by the affine linearity of g in the second factor. The final product is trivial because

$$\sum_{\substack{\text{divisors} \\ D=\sum_P n_{D,P}(P)}} m_D\left(\sum_P n_{D,P}x_P\right) = 0.$$

This gives freedom from the choice of coordinates. Now we turn to the question of different bases. Choose two vectors $\mathbf{T}$ and $\mathbf{R}$ in $E(K)^n$. Let $\mathbf{S} \in E(K)^n$ be chosen so that all $T_i, R_i \in \Gamma_{\mathbf{S}}$. Then, there exist matrices $M_{\mathbf{T}}$ and $M_{\mathbf{R}}$ such that

$$M_{\mathbf{T}}(\mathbf{S}) = \mathbf{T}, \qquad M_{\mathbf{R}}(\mathbf{S}) = \mathbf{R}$$

on E. We propose to show that $\aleph_{\mathbf{T}} = \aleph_{\mathbf{S}}$, which, repeated for $\mathbf{R}$, will give $\aleph_{\mathbf{T}} = \aleph_{\mathbf{R}}$ as desired.

Now, we use Theorem 10.1.1 with the matrix $M_{\mathbf{T}}$. We obtain that $W_{E,\mathbf{T}}$ and $W_{E,\mathbf{S}} \circ M_{\mathbf{T}}^{tr}$ are equivalent.

Suppose we also have that $\mathbf{v}_{\mathbf{S},P} = M_{\mathbf{T}}^{tr}(\mathbf{v}_{\mathbf{T},P})$. Then, by the condition that Θ is quadratic, the 'equivalence factor' vanishes and we have $\aleph_{\mathbf{T}} = \aleph_{\mathbf{S}}$ as required.

Suppose instead that we do not have $\mathbf{v}_{\mathbf{S},P} = M_{\mathbf{T}}^{tr}(\mathbf{v}_{\mathbf{T},P})$. Then we can alter our choice of coordinates according to the above, to reduce to the first case. □

The notation we have introduced is indeed clumsy, but it was necessary for the careful proof of the theorem. Henceforth we will adopt our softer notation:

Definition 10.3.3. Whenever we have a product of the form (10.15) which is defined independently of $\mathbf{T}$ as described in Theorem 10.3.1, we will say that it is a *quadratic quantity in elliptic nets for E* and write it

$$\prod_D \mathcal{W}\left(\sum_P n_{D,P} p\right)$$

where the $\mathcal{W}$ stands for any suitable choice of $W_{E,\mathbf{T}}$ and the lowercase p, q, r etc. stand for the $\mathbf{v}_{\mathbf{T},P}, \mathbf{v}_{\mathbf{T},Q}, \mathbf{v}_{\mathbf{T},R}$ etc. associated with the points P, Q, R appearing in the divisors.

Example 10.3.1. The expression

$$\frac{\mathcal{W}(p+q+r)\mathcal{W}(p)\mathcal{W}(q)\mathcal{W}(r)}{\mathcal{W}(p+q)\mathcal{W}(q+r)\mathcal{W}(r+p)}$$

is a quadratic quantity in elliptic nets for E, for any P, Q, R in E such that none of $P+Q+R$, P, Q, R, $P+Q$, $Q+R$, or $R+P$ vanish. To see this, calculate

$$(p+q+r)^2 + p^2 + q^2 + r^2 - (p+q)^2 - (q+r)^2 - (r+p)^2 = 0.$$

To evaluate such a quantity we may choose any suitable $\mathbf{T}$ (suitable in the sense that $P, Q, R \in \Gamma_{\mathbf{T}}$).

Finally, we wish to remove the nonvanishing conditions. The function $g(\mathbf{r}, \mathbf{k})$ of Theorem 10.2.3 is not well-defined when $W(\mathbf{k}) = 0$. However, for each $\mathbf{r}$, it extends uniquely as a function of $\mathbf{k}$ to preserve affine linearity.

Now we wish to extend all elliptic nets in the same fashion. That is, whenever $W(\mathbf{r}) = 0$, we would like to give a value in K^*, called the *residue* at $\mathbf{r}$. Then we can replace the values 0 with the respective residues to obtain an *extended elliptic net* $\overline{W}$ defined everywhere and taking all its values in K^*. First, set $\overline{W}(0) = 1$. Then, the extended function g determine values for all $\overline{W}(\mathbf{r})$ where $\mathbf{r}$ is in the lattice of zero-apparition via the relationship:

$$\overline{W}_{E,\mathbf{P}}(\mathbf{r}+\mathbf{k}) = \overline{W}_{E,\mathbf{P}}(\mathbf{k}) g(\mathbf{r}, \mathbf{k}).$$

Now repeat the proof of Theorem 10.3.1 with the extended functions $f_{\mathbf{r}}(\mathbf{k}) = g(\mathbf{r}, \mathbf{k})$ and the extended elliptic nets $\overline{W}$. We discover that the quantity

$$\prod_D \mathcal{W}\left(\sum_P n_{D,P} p\right),$$

defined using the extended elliptic nets, is still independent of basis and coordinates, and so is well-defined everywhere. Furthermore it is equal to the original definition wherever they are both defined.

To evaluate a quadratic quantity which includes a residue, it is most convenient simply to change coordinates so that it requires evaluating the easy residue $\overline{W}(\mathbf{0}) = 1$. If this is not possible, it becomes necessary to compute a residue.

Example 10.3.2. As an example of such a calculation, consider an elliptic divisibility sequence W associated to a curve

$$E : y^2 + xy + y = x^3 - x^2 - 3x + 3$$

and point $P = (0,1)$. This point has order 7 and so $W(7) = 0$. We can calculate its residue $\overline{W}(7)$ using Theorems 10.2.3 and 10.2.2:

$$\overline{W}(7) = \frac{\overline{W}(7)}{\overline{W}(0)} = g(7,0) = a^0 b^1 = \frac{W(8)^2 W(2)}{W(9)} = \frac{(134217728)^2(2)}{17179869184} = 2097152.$$

Chapter 11

Catching an elliptic curve

This chapter consists of several detailed examples verifying the results so far encountered.

11.1 An extended example

Example 11.1.1. We expand upon Example 3.2.6. Consider the elliptic curve

$$E : y^2 + y = x^3 + x^2 - 2x$$

and the points $P = (0,0)$, $Q = (1,0)$ on that curve. Some of the smaller terms of the net $W_{E,P,Q}$ can be calculated using Proposition 6.1.4.

$$W(0,0 = 0, \quad W(1,0) = W(0,1) = W(1,1) = 1$$

$$W(2,0) = 2y_1 + a_1x_1 + a_3 = 1, \qquad W(0,2) = 2y_2 + a_1x_2 + a_3 = 1$$

$$W(1,-1) = x_2 - x_1 = 1, \qquad W(2,1) = 2x_1 + x_2 - \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - a_1\left(\frac{y_2 - y_1}{x_2 - x_1}\right) + a_2 = 2$$

$$W(2,-1) = (y_1 + y_2)^2 - (2x_1 + x_2)(x_1 - x_2)^2 = -1$$

This example has been chosen to give small manageable numbers. More terms can be calculated using the recurrence relation (3.1) (for example using the algorithms in the scripts in Appendix B). The array in Figure 11.1 shows a portion of the elliptic net centred on $W(0,0) = 0$. Notice the symmetry property $W(-a,-b) = -W(a,b)$. There are no other zeroes visible: in fact, P and Q are independent non-torsion points. The centre row is the elliptic divisibility sequence associated to E and P, which begins

$$1, 1, -3, 11, 38, 249, -2357, 8767, 496035, -3769372, -299154043, -12064147359, 632926474117,$$
$$-65604679199921, -6662962874355342, -720710377683595651, 285131375126739646739,$$
$$5206174703484724719135, -360421577662469237888837209, 1414637218637532261361000 2376, \ldots$$

The centre column is the elliptic divisibility sequence associated to Q.

Figure 11.1: Elliptic net associated to $y^2+y=x^3+x^2-2x$, $P=(0,0)$, $Q=(1,0)$ over $\mathbb{Q}$

	−87039	−47987	3079	6522	3269	−2869	4335	5959	12016	−55287	23921	1587077	−7159461
	−19061	−3376	−1187	535	−127	−299	94	479	919	−2591	13751	68428	424345
	−55	−709	−129	67	−44	−27	−31	53	−33	−350	493	6627	48191
	841	−151	26	19	−1	−7	−5	8	−19	−41	−151	989	−1466
	184	−63	17	5	3	−2	1	3	−1	−13	−36	181	−1535
	−185	−47	−1	4	1	−1	1	1	2	−5	7	89	−149
	−249	−38	−11	3	−1	−1	0	1	1	−3	11	38	249
	149	−89	−7	5	−2	−1	−1	1	−1	−4	1	47	185
	1535	−181	36	13	1	−3	−1	2	−3	−5	−17	63	−184
	1466	−989	151	41	19	−8	5	7	1	−19	−26	151	−841
	−48191	−6627	−493	350	33	−53	31	27	44	−67	129	709	55
↑	−424345	−68428	−13751	2591	−919	−479	−94	299	127	−535	1187	3376	19061
Q	7159461	−1587077	−23921	55287	−12016	−5959	−4335	2869	−3269	−6522	−3079	47987	87039
	$P\rightarrow$												

Let's check an instance of the elliptic net recurrence relation (3.1) in this array. For example, let's check $p=(1,1), q=(-1,1), r=(2,0), s=(2,-1)$. The recurrence relation is then

$$W(2,1)W(2,0)W(4,-1)W(2,0)+W(3,0)W(-3,1)W(3,0)W(1,1) \\ +W(5,0)W(1,-1)W(1,0)W(-1,1)$$

which evaluates to

$$(2)(1)(1)(1)+(-3)(4)(-3)(1)+(38)(1)(1)(-1)=2+36-38=0.$$

Now let us examine this same elliptic curve and points over $\mathbb{F}_{17}$ in Figure 11.2. In this figure, a spade (♠) marks the centre $W(0,0)$; the other zeroes are marked by clubs (♣) to show the lattice of zero-apparition.

The reader is encouraged to check Lemma 10.2.5 for this elliptic net. For example, let $\mathbf{r}=(4,4)$. Then

$$a_{\mathbf{r}}=\frac{W_{E,P,Q}(r_1+2,r_2)}{W_{E,P,Q}(r_1+1,r_2)W_{E,P,Q}(2,0)}=\frac{W_{E,P,Q}(6,4)}{W_{E,P,Q}(5,4)W_{E,P,Q}(2,0)}=\frac{(13)}{(14)(1)}=7,$$
$$b_{\mathbf{r}}=\frac{W_{E,P,Q}(r_1,r_2+2)}{W_{E,P,Q}(r_1,r_2+1)W_{E,P,Q}(0,2)}=\frac{W_{E,P,Q}(4,6)}{W_{E,P,Q}(4,5)W_{E,P,Q}(0,2)}=\frac{(2)}{(15)(1)}=16,$$
$$c_{\mathbf{r}}=\frac{W_{E,P,Q}(r_1+1,r_2+1)}{a_{\mathbf{r}}b_{\mathbf{r}}W_{E,P,Q}(1,1)}=\frac{W_{E,P,Q}(5,5)}{a_{\mathbf{r}}b_{\mathbf{r}}W_{E,P,Q}(1,1)}=\frac{(3)}{(7)(16)(1)}=2.$$

So, for the example $\mathbf{k}=(-7,5)$, equation (10.10) on the left hand side is

$$\frac{W_{E,P,Q}(-3,9)}{W_{E,P,Q}(-7,5)}=\frac{11}{2}=14.$$

and on the right,

$$a_{\mathbf{r}}^{-7}b_{\mathbf{r}}^{-5}c_{\mathbf{r}}^{1}=(7)^{-7}(16)^{-5}(2)=14.$$

Now let's change basis. Select a small integer matrix

$$T=\begin{pmatrix}3 & 2\\ -1 & 1\end{pmatrix}$$

Then the new basis is $T(P,Q)=(P',Q')=([3]P+[2]Q,[-1]P+Q)=((\frac{-36}{169},\frac{755}{2197}),(-1,1))$. The associated elliptic net is shown in Figure 11.3. Notice that this net is not integral, since the initial values have denominators. These intial values are

$$W(0,0=0,\quad W(1,0)=W(0,1)=W(1,1)=1$$
$$W(2,0)=2y_1+a_1x_1+a_3=\frac{3707}{2197},\qquad W(0,2)=2y_2+a_1x_2+a_3=3$$
$$W(1,-1)=W(0,1)^3W(2,1)-W(1,0)^3W(1,2)=x_2-x_1=\frac{-133}{169}$$
$$W(2,1)=2x_1+x_2-\left(\frac{y_2-y_1}{x_2-x_1}\right)^2-a_1\left(\frac{y_2-y_1}{x_2-x_1}\right)+a_2=\frac{-68428}{61009},$$
$$W(1,2)=2x_2+x_1-\left(\frac{y_2-y_1}{x_2-x_1}\right)^2-a_1\left(\frac{y_2-y_1}{x_2-x_1}\right)+a_2=\frac{-689}{361}.$$

Figure 11.2: Elliptic net associated to $y^2 + y = x^3 + x^2 - 2x$, $P = (0,0)$, $Q = (1,0)$ over $\mathbb{F}_{17}$

6	13	8	6	11	2	15	4	8	16	8	4	15	2	11	6	8	13	6	16	14	8	6	13	3	4	16	10	3	2	4
8	3	3	15	9	13	10	16	13	5	6	♣	7	5	16	1	6	13	15	2	12	3	2	♣	1	12	10	2	1	16	12
5	13	11	8	5	2	16	3	12	16	13	2	1	9	16	2	4	16	5	3	1	2	12	8	6	13	12	15	11	15	15
2	1	7	♣	14	13	16	3	3	11	8	10	10	16	15	♣	8	16	6	7	2	11	12	14	4	13	5	♣	6	1	8
15	9	6	8	12	1	11	2	12	8	16	5	5	13	13	9	16	2	16	9	13	13	5	5	16	8	12	2	11	1	12
14	1	13	2	6	5	13	♣	8	14	3	2	9	4	10	1	13	12	6	♣	7	12	16	16	6	4	15	15	12	14	2
13	9	3	6	1	1	3	1	11	2	14	13	11	16	8	7	6	9	15	16	9	4	8	16	2	9	11	7	9	16	6
1	1	14	11	4	11	6	12	9	4	6	♣	12	1	4	12	5	10	2	6	11	13	8	♣	2	13	10	11	9	10	3
4	13	1	4	1	9	5	11	4	1	5	16	10	2	12	4	10	16	2	10	7	9	8	1	2	4	2	1	8	9	7
15	3	5	♣	5	14	15	16	3	1	4	2	11	5	4	♣	9	14	14	2	8	4	7	1	4	12	11	♣	10	12	1
6	13	8	2	8	15	15	15	2	13	7	3	8	9	7	9	3	1	10	15	3	8	13	12	14	13	1	8	4	2	13
10	10	1	10	12	4	13	♣	4	13	5	7	16	7	7	3	2	1	7	♣	14	13	16	3	3	11	8	10	10	16	15
6	15	2	5	6	16	9	8	8	8	2	9	2	16	10	12	8	15	10	2	3	13	7	9	3	2	4	14	14	16	16
4	5	11	2	4	1	3	16	15	14	5	♣	5	3	15	1	3	16	4	15	11	12	4	♣	9	3	14	15	8	13	7
14	8	6	13	3	4	16	10	3	2	4	16	4	1	16	1	1	2	12	7	4	4	12	4	10	8	5	1	10	13	15
13	16	5	♣	11	13	8	5	11	6	13	6	3	16	16	♠	1	1	14	11	4	11	6	12	9	4	6	♣	12	1	4
2	4	7	16	12	9	7	13	5	13	13	10	5	15	16	16	1	16	13	1	13	15	14	7	1	13	14	4	11	9	3
10	4	9	2	3	14	8	♣	13	5	6	2	13	1	14	16	2	14	12	♣	12	3	2	1	14	16	13	15	6	12	13
1	1	3	3	13	15	14	8	10	4	14	15	7	2	9	5	7	1	15	8	15	9	9	9	8	1	11	12	15	2	11
2	1	7	7	9	6	14	14	1	4	3	♣	10	16	15	14	10	10	1	10	12	4	13	♣	4	13	5	7	16	7	7
4	15	13	9	16	4	3	5	4	9	14	2	7	16	14	8	10	8	9	14	10	4	15	2	2	2	9	15	9	4	11
16	5	7	♣	6	5	13	16	10	13	9	15	3	3	8	♣	13	12	6	15	13	16	14	1	2	3	12	♣	12	14	2
10	8	9	16	15	13	15	16	9	8	10	7	15	1	7	13	5	15	7	1	12	16	13	6	12	8	16	13	16	4	13
14	7	8	6	7	4	15	♣	9	4	6	11	15	7	12	5	13	16	5	♣	11	13	8	5	11	6	13	6	3	16	16
11	1	8	10	6	8	15	1	9	13	8	1	2	8	11	10	9	1	6	4	3	15	6	16	14	16	16	11	14	8	4
15	3	5	2	2	13	11	1	1	5	10	♣	11	5	4	16	7	13	8	15	14	3	9	♣	4	12	11	15	4	16	3
5	16	6	15	5	9	1	12	12	4	4	8	1	15	1	8	4	4	12	12	1	9	5	15	6	16	5	9	11	8	2
9	16	11	♣	12	4	13	3	5	6	15	10	11	1	9	♣	2	1	7	7	9	6	14	14	1	4	3	♣	10	16	15
2	2	6	2	5	4	11	9	5	15	16	14	12	1	13	15	1	8	16	15	4	1	5	14	1	15	12	9	6	4	12
5	1	16	15	7	5	16	♣	15	14	5	15	2	4	11	16	1	12	10	♣	11	12	4	1	7	4	8	2	14	14	9
13	15	14	7	1	13	14	4	11	9	3	1	11	4	9	11	6	15	2	13	9	1	9	13	2	15	6	11	9	4	11

Figure 11.3: Elliptic net associated to $y^2+y=x^3+x^2-2x$, $P=(\frac{-36}{169},\frac{755}{2197})$, $Q=(-1,1)$ over $\mathbb{Q}$

$\frac{64017366986980252}{23298085122481}$	$\frac{-391875247}{371293}$	-129	$\frac{331747}{6859}$	$\frac{-464635803151}{16983563041}$	$\frac{-41753192521927146}{2213314919066161}$
$\frac{36544816947871}{137858491849}$	$\frac{1268915}{28561}$	-19	$\frac{-15886}{6859}$	$\frac{218120695}{47045881}$	$\frac{-87224819531}{16983563041}$
$\frac{17849937049}{815730721}$	$\frac{13718}{2197}$	3	$\frac{-689}{361}$	$\frac{1259}{6859}$	$\frac{280178460819}{103359800557}$
$\frac{-27785809}{4826809}$	$\frac{133}{169}$	1	1	$\frac{-68428}{61009}$	$\frac{43030385549}{33107082931}$
$\frac{-3707}{2197}$	-1	0	1	$\frac{3707}{2197}$	$\frac{-3439168815}{815730721}$
$\frac{68428}{61009}$	-1	-1	$\frac{-133}{169}$	$\frac{27785809}{4826809}$	$\frac{320992934452306}{23298085122481}$
$\frac{-1259}{6859}$	$\frac{689}{361}$	-3	$\frac{-13718}{2197}$	$\frac{-17849937049}{815730721}$	$\frac{9420521994063176331}{51185893014090757}$

From Theorem 4.4.1, the denominators appearing in this elliptic net should come from these denominators and the numerator of $W(1,-1)$. Factoring these, we obtain the primes 7, 13, and 19. All the denominators in Figure 11.3 have prime factorisations containing only these primes. For example,

$$W_{E,P',Q'}(-2,4)=\frac{64017366986980252}{23298085122481}$$

in the upper left has denominator

$$23298085122481=13^{12}.$$

As an example of Theorem 10.1.1, consider the point $[5]P+[5]Q=[2]P'+[1]Q'$. The equation (10.2) has left hand side

$$W_{E,P,Q}\left(\begin{pmatrix}3&-1\\2&1\end{pmatrix}\begin{pmatrix}2\\1\end{pmatrix}\right)=W_{E,P,Q}(5,5)=68428$$

and right hand side

$$\begin{aligned}&W_{E,P',Q'}(2,1)W_{E,P,Q}(3,2)^2W_{E,P,Q}(-1,1)^{-1}W_{E,P,Q}(2,3)^2\\&=\left(\frac{-68428}{61009}\right)(-13)^2(-1)^{-1}(-19)^2=68428,\end{aligned}$$

verifying the statement.

11.2 A closer look at the $\mathbb{G}_m$ case

Example 11.2.1. Consider the sequence of even-indexed Fibonacci numbers,

1, 3, 8, 21, 55, 144, 377, 987, 2584, 6765, 17711, 46368, 121393, 317811, 832040, 2178309, 5702887, 14930352, 39088169, 102334155, 267914296, 701408733, 1836311903, 4807526976, 12586269025, 32951280099, 86267571272, 225851433717, 591286729879, 1548008755920, 4052739537881, 10610209857723, 27777890035288, 72723460248141, 190392490709135, 498454011879264, 1304969544928657, 3416454622906707, 8944394323791464, 23416728348467685, ...

which satisfy the recurrence relation

$$W(n+2) = 3W(n+1) - W(n).$$

This Lucas sequence must be associated to a singular cubic curve and point. Using Swart's Proposition 8.2.1, the curve and point can be chosen to be

$$C : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \qquad P = (0,0)$$

where

$$a_1 = \frac{W(4) + W(2)^5 - 2W(2)W(3)}{W(2)^2W(3)} = \frac{21 + 3^5 - (2)(3)(8)}{3^2(8)} = 3$$
$$a_2 = \frac{W(2)W(3)^2 + W(4) + W(2)^5 - W(2)W(3)}{W(2)^3W(3)} = \frac{(3)8^2 + 21 + 3^5 - (3)(8)}{3^3(8)} = 2$$
$$a_3 = W(2) = 3, \qquad a_4 = 1, \qquad a_6 = 0$$

which gives the singular cubic

$$C : y^2 + 3xy + 3y = x^3 + 2x^2 + x.$$

Take another point $Q = (1, \sqrt{13} - 3)$ on this curve. The elliptic divisibility sequence associated to C and Q begins

$$1,\ 2\sqrt{13},\ 88,\ 576\sqrt{13},\ 97280,\ 2523136\sqrt{13},\ 1700790272,\ 176362094592\sqrt{13},\ 475470059536384,$$
$$197208405557903360\sqrt{13},\ 21266718016386139136,\ 3528271845490278518489088\sqrt{13},$$
$$152193787051469992404816232448,\ 100999318809160606336084888841379 84\sqrt{13},$$
$$174266260490479115765850543576936611840,$$
$$462588489574296349185285316088637528893030 4\sqrt{13},$$
$$319263801325351639864156552348727777441526853632,\ \ldots$$

This is another elliptic divisibility sequence on the same singular cubic, so we expect it to be a singular sequence. Morgan Ward showed that such sequences are scale equivalent either to the integers or a Lucas sequence [74, Thm. 22.1]. In our case, it must be the latter, since our singularity is a node. In fact, consider the equivalent sequence

$$A_n = \sqrt{2}^{n^2-1} W_{E,P}(n),$$

which begins

$$1,\ \frac{\sqrt{13}}{\sqrt{2}},\ \frac{11}{2},\ \frac{9\sqrt{13}}{2\sqrt{2}},\ \frac{95}{4},\ \frac{77\sqrt{13}}{4\sqrt{2}},\ \frac{811}{8},\ \frac{657\sqrt{13}}{8\sqrt{2}},\ \frac{6919}{16},\ \frac{5605\sqrt{13}}{16\sqrt{2}},\ \frac{59027}{32},$$
$$\frac{47817\sqrt{13}}{32\sqrt{2}},\ \frac{503567}{64},\ \frac{407933\sqrt{13}}{64\sqrt{2}},\ \frac{4295995}{128},\ \ldots$$

This sequence satisfies the linear recurrence relation

$$A_{n+2} = \left(\frac{\sqrt{13}}{\sqrt{2}}\right) A_{n+1} - A_n.$$

For example,

$$A_3 = \left(\frac{\sqrt{13}}{\sqrt{2}}\right) A_2 - A_1 = \left(\frac{\sqrt{13}}{\sqrt{2}}\right)\left(\frac{\sqrt{13}}{\sqrt{2}}\right) - 1 = \frac{11}{2}.$$

The rank two elliptic net associated to C, P and Q is shown in Figure 11.4 (a multi-dimensional Fibonacci sequence!). In this figure, the vectors

$$\begin{pmatrix} a \\ b \end{pmatrix}$$

correspond to the numbers $a\sqrt{13}+b$. The origin is in the second column. Notice the conjugate symmetry in this array:

$$W(a,b) = \overline{W(a,-b)}.$$

The curve C has a node at $(-1,0)$. The tangent lines are $y = \left(-\frac{3}{2} \pm \frac{\sqrt{5}}{2}\right)(x+1)$. Let us denote the non-singular part of C by C_{ns}. Then C_{ns} is isomorphic with K^* under the isomorphism

$$(x,y) \mapsto \frac{2y + (3+\sqrt{5})(x+1)}{2y + (3-\sqrt{5})(x+1)}. \tag{11.1}$$

That is to say, C_{ns} is a twisted form of $\mathbb{G}_m$.

The point $P = (0,0)$ is associated to the unit

$$\left(\frac{3+\sqrt{5}}{3-\sqrt{5}}\right)$$

in the multiplicative group. If p is a prime of $\mathbb{Q}$ and we reduce the curve C and point P modulo p, the associated elliptic divisibility sequence is reduced modulo p. But this corresponds to reduction modulo $\mathfrak{p}$ on $\mathbb{G}_m$, for some prime $\mathfrak{p}$ of $\mathbb{Q}(\sqrt{5})$ lying over p. Therefore the order of the reduced point P must divide p^2-1 or $p-1$ depending on whether p splits in $\mathbb{Q}(\sqrt{5})$. For example, the order of the point modulo 7 should divide 48, while the order of the point modulo 11 should divide 10. Accordingly, 7 divides the 4-th term of $W_{E,P}$ and 11 divides the 10-th.

Another consequence of the form of the isomorphism (11.1) is that all rational points on C_{ns} map to elements of the unit group in $\mathbb{Q}(\sqrt{5})$. This unit group is rank 1, and so all rational points on C_{ns} are dependent. This means that any entirely rational elliptic net of rank two associated to C must have non-trivial zeroes, and its terms are derived from those appearing in some single elliptic divisibility sequence (see Example 11.3.1 for an elliptic net with dependent points). This explains the choice of Q for the example in Figure 11.4: Q should be independent of P since otherwise it can't be considered a true 'two-dimensional Fibonacci sequence'!

Figure 11.4: Elliptic net associated to $y^2+3xy+3y=x^3+2x^2+x$ and points $P=(0,0)$ and $Q=(1,\sqrt{13}-3)$

$$\begin{pmatrix} 624869376 \\ 2252980224 \end{pmatrix} \quad \begin{pmatrix} 2523136 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 624869376 \\ -2252980224 \end{pmatrix} \quad \begin{pmatrix} 307737706496 \\ -1109564080128 \end{pmatrix} \quad \begin{pmatrix} 145553031069696 \\ -524798916820992 \end{pmatrix} \quad \begin{pmatrix} 48306063204990976 \\ -174169987801399296 \end{pmatrix}$$
$$\begin{pmatrix} 2016768 \\ 7270912 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 97280 \end{pmatrix} \quad \begin{pmatrix} 2016768 \\ -7270912 \end{pmatrix} \quad \begin{pmatrix} -294087168 \\ 1060346368 \end{pmatrix} \quad \begin{pmatrix} 35253070848 \\ -127106754560 \end{pmatrix} \quad \begin{pmatrix} 3858417639936 \\ -13911722642944 \end{pmatrix}$$
$$\begin{pmatrix} 12864 \\ 46336 \end{pmatrix} \quad \begin{pmatrix} 576 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 12864 \\ -46336 \end{pmatrix} \quad \begin{pmatrix} 512064 \\ -1846272 \end{pmatrix} \quad \begin{pmatrix} 742464 \\ -2676992 \end{pmatrix} \quad \begin{pmatrix} -6985271232 \\ 25185753600 \end{pmatrix}$$
$$\begin{pmatrix} 156 \\ 556 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 88 \end{pmatrix} \quad \begin{pmatrix} 156 \\ -556 \end{pmatrix} \quad \begin{pmatrix} -924 \\ 3332 \end{pmatrix} \quad \begin{pmatrix} -101448 \\ 365776 \end{pmatrix} \quad \begin{pmatrix} 12164268 \\ -43858892 \end{pmatrix}$$
$$\begin{pmatrix} 3 \\ 9 \end{pmatrix} \quad \begin{pmatrix} 2 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 3 \\ -9 \end{pmatrix} \quad \begin{pmatrix} -29 \\ 105 \end{pmatrix} \quad \begin{pmatrix} -1278 \\ 4608 \end{pmatrix} \quad \begin{pmatrix} -38365 \\ 138327 \end{pmatrix}$$
$$\begin{pmatrix} 0 \\ -1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 3 \\ -10 \end{pmatrix} \quad \begin{pmatrix} -27 \\ 98 \end{pmatrix} \quad \begin{pmatrix} 237 \\ -854 \end{pmatrix}$$
$$\begin{pmatrix} 0 \\ -1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 3 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 8 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 21 \end{pmatrix}$$
$$\begin{pmatrix} 0 \\ -1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ -1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 3 \\ 10 \end{pmatrix} \quad \begin{pmatrix} 27 \\ 98 \end{pmatrix} \quad \begin{pmatrix} 237 \\ 854 \end{pmatrix}$$
$$\begin{pmatrix} -3 \\ 9 \end{pmatrix} \quad \begin{pmatrix} -2 \\ 0 \end{pmatrix} \quad \begin{pmatrix} -3 \\ -9 \end{pmatrix} \quad \begin{pmatrix} 29 \\ 105 \end{pmatrix} \quad \begin{pmatrix} 1278 \\ 4608 \end{pmatrix} \quad \begin{pmatrix} 38365 \\ 138327 \end{pmatrix}$$
$$\begin{pmatrix} -156 \\ 556 \end{pmatrix} \quad \begin{pmatrix} 0 \\ -88 \end{pmatrix} \quad \begin{pmatrix} -156 \\ -556 \end{pmatrix} \quad \begin{pmatrix} -924 \\ -3332 \end{pmatrix} \quad \begin{pmatrix} 101448 \\ 365776 \end{pmatrix} \quad \begin{pmatrix} 12164268 \\ 43858892 \end{pmatrix}$$
$$\begin{pmatrix} -12864 \\ 46336 \end{pmatrix} \quad \begin{pmatrix} -576 \\ 0 \end{pmatrix} \quad \begin{pmatrix} -12864 \\ -46336 \end{pmatrix} \quad \begin{pmatrix} -512064 \\ -1846272 \end{pmatrix} \quad \begin{pmatrix} -742464 \\ -2676992 \end{pmatrix} \quad \begin{pmatrix} 6985271232 \\ 25185753600 \end{pmatrix}$$
$$\begin{pmatrix} -2016768 \\ 7270912 \end{pmatrix} \quad \begin{pmatrix} 0 \\ -97280 \end{pmatrix} \quad \begin{pmatrix} -2016768 \\ -7270912 \end{pmatrix} \quad \begin{pmatrix} -294087168 \\ -1060346368 \end{pmatrix} \quad \begin{pmatrix} -35253070848 \\ -127106754560 \end{pmatrix} \quad \begin{pmatrix} 3858417639936 \\ 13911722642944 \end{pmatrix}$$
$$\begin{pmatrix} -624869376 \\ 2252980224 \end{pmatrix} \quad \begin{pmatrix} -2523136 \\ 0 \end{pmatrix} \quad \begin{pmatrix} -624869376 \\ -2252980224 \end{pmatrix} \quad \begin{pmatrix} -307737706496 \\ -1109564080128 \end{pmatrix} \quad \begin{pmatrix} -145553031069696 \\ -524798916820992 \end{pmatrix} \quad \begin{pmatrix} -48306063204990976 \\ -174169987801399296 \end{pmatrix}$$

Figure 11.5: Elliptic net associated to $y^2+2xy+2y=x^3+2x^2+x$ and $P=(0,0)$

$2^8 3^{-20}(11)$	$3^{-16}(12)$	$2^{-8}3^{-12}(13)$	$2^{-16}3^{-8}(14)$	$2^{-24}3^{-4}(15)$	$2^{-32}(16)$
$2^6 3^{-12}(8)$	$3^{-9}(9)$	$2^{-6}3^{-6}(10)$	$2^{-12}3^{-3}(11)$	$2^{-18}(12)$	$2^{-24}3^3(13)$
$2^4 3^{-6}(5)$	$3^{-4}(6)$	$2^{-4}3^{-2}(7)$	$2^{-8}(8)$	$2^{-12}3^2(9)$	$2^{-16}3^4(10)$
$2^2 3^{-2}(2)$	$3^{-1}(3)$	$2^{-2}(4)$	$2^{-4}3^1(5)$	$2^{-6}3^2(6)$	$2^{-8}3^3(7)$
(-1)	(0)	(1)	(2)	(3)	(4)
$2^{-2}(-4)$	$3^{-1}(-3)$	$2^2 3^{-2}(-2)$	$2^4 3^{-3}(-1)$	$2^6 3^{-4}(0)$	$2^8 3^{-5}(1)$

11.3 What about $\mathbb{G}_a$?

Example 11.3.1. Suppose we wish to produce the famous sequence

$$1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,\ldots?$$

Using Proposition 8.2.1, we can calculate the associated curve and point:

$$y^2+2xy+2y=x^3+2x^2+x, \qquad P=(0,0).$$

Suppose we choose another point, $Q=[3]P=(\frac{-8}{9},\frac{-2}{27})$. Then the associated elliptic net is formed from two dependent points. It is shown in Figure 11.5, where the values are suggestively factored (the origin is at the intersection of the second row from bottom and second column). The elliptic net is of the form

$$W(n,m)=2^{-2nm}3^{nm-m^2}(n+3m).$$

All elliptic nets satisfy $W(1,0)=W(0,1)=W(1,1)=1$ from which fact one can deduce the powers of 2 and 3.

Part III

Deeper connections

Chapter 12

Three perspectives on group extensions

This chapter contains background relating to group extensions in general and central extensions in particular. We look at extensions from three perspectives: homological algebra and the Baer sum; cohomology and factor sets; and multiplicative torsors. The formalism of factor sets and torsors, in particular, will be used in later chapters.

12.1 Group extensions and Baer sum

In this section we see that the set of group extensions of a group G by an G-module M form a group. Weibel [75, §3.4] provides a good reference for this section.

Consider an extension of groups

$$0 \longrightarrow M \xrightarrow{j} X \xrightarrow{\pi} G \longrightarrow 0.$$

in which M is abelian. To each such extension is associated an action of G on M, for, we may choose any section σ of π and let G act on M by conjugation as follows:

$$m^g = j^{-1}(\sigma(g)\, j(m) \sigma(g)^{-1}).$$

This is well defined since $j(M)$ is a normal abelian subgroup of X (this guarantees that conjugation by a different choice of section σ' has the same effect, for $\sigma'(g) = \sigma(g)\, j(m_0)$ for some m_0). This action gives a G-module structure to M.

Therefore, given a group G and group M already endowed with a G-module structure, we may consider the set of all group extensions

$$0 \longrightarrow M \xrightarrow{j} X \xrightarrow{\pi} G \longrightarrow 0.$$

giving rise in the sense above to the G-module structure already specified on M. We will see that this set, modulo an appropriate equivalence relation, forms a group. In the next section we will show that it is isomorphic to the group $H^2(G, M)$ in group cohomology.

If there are two extensions X and X', such that the following diagram commutes, where α is a group isomorphism then we say that X and X' are equivalent.

$$\begin{array}{ccccccccc} 0 & \longrightarrow & M & \longrightarrow & X & \longrightarrow & G & \longrightarrow & 0 \\ & & \downarrow{\scriptstyle id} & & \downarrow{\scriptstyle \alpha} & & \downarrow{\scriptstyle id} & & \\ 0 & \longrightarrow & M & \longrightarrow & X' & \longrightarrow & G & \longrightarrow & 0 \end{array} \tag{12.1}$$

This is evidently an equivalence relation.

Furthermore, we may define a composition law on extensions. Suppose we wish to add X and X' defined by

$$0 \longrightarrow M \xrightarrow{j_1} X \xrightarrow{\pi_1} G \longrightarrow 0,$$

$$0 \longrightarrow M \xrightarrow{j_2} X' \xrightarrow{\pi_2} G \longrightarrow 0.$$

Their *Baer sum* is defined as follows. Let Y be the pullback

$$\begin{array}{ccc} Y & \longrightarrow & X' \\ \downarrow & & \downarrow{\scriptstyle \pi_2} \\ X & \xrightarrow{\pi_1} & G \end{array} \tag{12.2}$$

That is, $Y = \{(x_1, x_2) \in X \times X' : \pi_1(x_1) = \pi_2(x_2)\}$. In particular, $Y \supset j_1(M) \times j_2(M)$. Define X'' to be the quotient of Y by the antidiagonal (or *skew diagonal*) $\widetilde{\Delta} = \{(j_1(m), j_2(m)^{-1}) : m \in M\}$. Then, X'' forms an extension

$$0 \longrightarrow M \xrightarrow{j} X'' \xrightarrow{\pi} G \longrightarrow 0$$

where j is the identification of M with $M \times 0$ (which is identified with $0 \times M$ in X''), and π is given by either of the two commuting maps $Y \to G$ in (12.2) (which is well-defined since $\widetilde{\Delta}$ is in the kernel). The exactness of this sequence follows from the exactness for X and X'.

Proposition 12.1.1. *Fix a group G and a G-module M. The set of equivalence classes of extensions of G by M forms a group under Baer sum.*

Proof. To check that this is a well-defined operation on the collection of equivalence classes, assume that the pairs $\{X, Y\}$, $\{X', Y'\}$ are pairs of equivalent extensions, X'' is the Baer sum of X and X', and Y'' is the Baer sum of Y and Y'. We must show that X'' and Y'' are equivalent. The equivalences of the first two pairs gives group homomorphisms $\alpha_1 : X \to Y$ and $\alpha_2 : X' \to Y'$. Define $\alpha : X'' \to Y''$ by $\alpha((\overline{x}_1, \overline{x}_2)) = (\overline{\alpha_1(x_1)}, \overline{\alpha_2(x_2)})$, where the overline represents quotient by the anti-diagonal as in the definition of Baer sum. This is clearly a group homomorphism. Also, the required commutative diagram of extensions follows from the respective diagrams for the equivalences of the pairs $\{X, Y\}$ and $\{X', Y'\}$.

The identity for this group is the equivalence class containing the extension

$$0 \longrightarrow M \xrightarrow{j} M \oplus G \xrightarrow{\pi} G \longrightarrow 0.$$

For, suppose that X in the notation of the Baer sum definition is this extension. Then, continuing with that notation, $Y \cong M \oplus X'$, and $X'' \cong (M \oplus X')/\widetilde{\Delta} \cong X'$. (This last isomorphism has the form $(m,x) \mapsto x\,j(m)^{-1}, x \mapsto (0,x)$).

The inverse of an extension X is the X' such that X'' is equivalent to the extension $M \oplus G$. We define X' by $X' \cong X$, $\pi_2 = \pi_1$ but $j_2 = j_1 \circ [-1]$ where $[-1]$ represents the map $m \mapsto m^{-1}$ on M. Then $\widetilde{\Delta} = \{(j_1(m), j_2(m)^{-1} : m \in M\} = \{(j_1(m), j_1(m)) : m \in M\}$ Then $X'' = \{(x_1,x_2) : \pi_1(x_1) = \pi_2(x_2)\}/\widetilde{\Delta} \cong X'$. This last isomorphism is given by $(\overline{x_1}, \overline{x_2}) \mapsto x_2 x_1^{-1}, x \mapsto (\overline{x}, \overline{0})$. □

If we assume that the G-action on M is trivial, then $\sigma(g)\,j(m)\sigma(g)^{-1} = j(m)$ for all $\sigma(g)$, and M is an abelian group, so j takes M into the centre of X. Converseley, if j takes M into the centre of X then the G-action on M is trivial. These are called *central extensions* of the group G by the abelian group M. This is the case we will be concerned with later.

12.2 Factor sets and $H^2(G,M)$

Now we will associate to each extension a function $f : G \times G \to M$ called a *factor set.* Dummit and Foote describe much of what is covered in this section [15, §17.4]. Suppose we choose a section $\sigma : G \to X$ of π. Then for any $g,h \in G$, $\sigma(g)\sigma(h)$ is in the same coset as $\sigma(gh)$ of X/M. So there is some $f_\sigma(g,h) \in M$ such that

$$\sigma(g)\sigma(h) = f_\sigma(g,h)\sigma(gh). \tag{12.3}$$

This defines a function $f_\sigma : G \times G \to M$. Every element of X may be written uniquely in the form $j(m)\sigma(g)$ for some $m \in M, g \in G$. Thus, we may identify X with $M \times G$ with the modified group law '$\oplus$' given by

$$\begin{aligned}
(m,g) \oplus (n,h) &= j(m)\sigma(g)\,j(n)\sigma(h) \\
&= j(m)\sigma(g)\,j(n)\sigma(g)^{-1}\sigma(g)\sigma(h) \\
&= j(m)\,j(n^g)\sigma(g)\sigma(h) \\
&= j(mn^g f_\sigma(g,h))\sigma(gh) \\
&= (mn^g f_\sigma(g,h), gh).
\end{aligned} \tag{12.4}$$

(Recall that G acts on M by conjugation, and this action is denoted m^g.)

This group law must satisfy the associativity law, from which we derive the following identity for f_σ:

$$f_\sigma(g,h)\,f_\sigma(gh,k) = f_\sigma(h,k)^g f_\sigma(g,hk). \tag{12.5}$$

We will use this property as defining and say that any function $f : G \times G \to M$ satisfying (12.5) is called a *factor set.* Recall that $H^2(G,M)$ consists of 2-cocycles $f : G \times G \to M$ modulo 2-coboundaries. The condition (12.5) is exactly the 2-cocycle condition for $H^2(G,M)$, i.e., f_σ is a 2-cocycle.

Now suppose that σ' is another section for the same extension, and $f_{\sigma'}$ the associated factor set. How do f_σ and $f_{\sigma'}$ relate? Observe that since $\sigma(g)$ and $\sigma'(g)$ always lie in the same coset, we have

$$\sigma(g) = f_1(g)\sigma'(g)$$

for some $f_1(g) \in M$. Hence we may define a function $f_1 : G \to M$. Then, we check that

$$\begin{aligned}
\sigma'(g)\sigma'(h) &= f_1(g)\sigma(g)f_1(h)\sigma(h) \\
&= f_1(g)\sigma(g)f_1(h)\sigma(g)^{-1}\sigma(g)\sigma(h) \\
&= f_1(g)f_1(h)^g f_\sigma(g,h)\sigma(gh) \\
&= f_1(g)f_1(h)^g f_\sigma(g,h)f_1(gh)^{-1}\sigma'(gh)
\end{aligned}$$

from which we deduce that

$$f_{\sigma'}(g,h)f_\sigma(g,h)^{-1} = f_1(g)f_1(h)^g f_1(gh)^{-1}$$

for some $f_1 : G \to M$. This says exactly that f_σ and $f_{\sigma'}$ differ by a 2-coboundary in $H^2(G,M)$. Therefore, to each extension of G by M, we associate an element of $H^2(G,M)$.

Now we verify that two equivalent extensions have the same factor set. Consider the diagram (12.1). Then if σ is a section for the top row, $\alpha \circ \sigma$ is a section for the bottom row. Applying α to equation (12.3), and recalling that α is the identity on $j(M)$, we have

$$\alpha \circ \sigma(g)\alpha \circ \sigma(h) = f_\sigma(g,h)\alpha \circ \sigma(gh)$$

which is to say, $f_\sigma = f_{\alpha\circ\sigma}$.

Now, we demonstrate the map in the other direction: how a factor set determines an extension. First, note that given any factor set f, we may choose a factor set $\hat{f}$ from the same cohomology class satisfying $\hat{f}(1,h) = \hat{f}(g,1) = 1$ by subtracting a coboundary δf_1 where $f_1(g) = f(1,1)$ is a constant element of $H^1(G,M)$. First, note that (12.5) implies that $f(1,1)f(1,k) = f(1,k)f(1,k)$ and so $f(1,k) = f(1,1)$. Also, it implies that $f(1,1)^g = f(g,1)$. Thus we may calculate $\hat{f}(1,h) = f(1,h)f(1,1)^{-1} = 1$ and $\hat{f}(g,1) = f(g,1)(f(1,1)^g)^{-1} = f(g,1)f(g,1)^{-1} = 1$.

Therefore we may assume without loss of generality that the factor set with which we intend to build our extension satisfies $f(1,h) = f(g,1) = 1$. The group law $\oplus$ described in (12.4) gives the clue to defining the extension: let X be $M \times G$ as a set together with the operation

$$(m,g) \oplus (n,h) = (mn^g f(g,h), gh).$$

The factor set property gives associativity for this group law, and the property that $f(1,h) = f(g,1) = 1$ shows that $(1,1)$ is the identity. Taking $k = 1, h = g^{-1}$ in (12.5) gives $f(g,g^{-1}) = 1$. Thus the inverse of (m,g) is $((m^{g^{-1}})^{-1}, g^{-1})$.

Thus we have a bijection between the group of equivalence classes of extensions and $H^2(G,M)$. Finally, we must verify that this is a group homomorphism. Suppose that we have the setup as described in the definition of the Baer sum, σ_1 is a section of π_1 and σ_2 is a section of π_2. Then define σ

a section for π by $\sigma(g) = (\overline{\sigma_1(g)}, \overline{\sigma_2(g)})$. Then we may calculate

$$\begin{aligned}
\sigma(g)\sigma(h) &= (\overline{\sigma_1(g)}, \overline{\sigma_2(g)})(\overline{\sigma_1(h)}, \overline{\sigma_2(h)}) \\
&= (\overline{\sigma_1(g)\sigma_1(h)}, \overline{\sigma_2(g)\sigma_2(h)}) \\
&= (\overline{f_{\sigma_1}(g,h)\sigma_1(gh)}, \overline{f_{\sigma_2}(g,h)\sigma_2(gh)}) \\
&= (\overline{f_{\sigma_1}(gh)}, \overline{f_{\sigma_2}(gh)})(\overline{\sigma_1(gh)}, \overline{\sigma_2(gh)}) \\
&= (\overline{f_{\sigma_1}(gh)}, \overline{f_{\sigma_2}(gh)})\sigma(gh).
\end{aligned}$$

Recall that $M \times 0$ and $0 \times M$ are identified and are the image of M in the extension X''. Therefore $f_\sigma = f_{\sigma_1} f_{\sigma_2}$ as needed.

12.3 Multiplicative torsors

Yet another perspective on central extensions is explained by Grothendieck in SGA7 [1, exp. VII]; see also [54, §10.2] and [9, §1]. Recall that for a group G and an abelian group B, a B-torsor X over G is a set with a free action of B such that the quotient of X by the action is G.

Any two B-torsors X_1 and X_2 over a base G can be added. First take $X_1 \times_G X_2$, the fibre product of the torsors over G. This has two actions of B, one for each coordinate. Form the quotient with respect to the B action $b \cdot (x_1, x_2) = (b \cdot x_1, b^{-1} \cdot x_2)$, and we are left with a single action of B acting by $b \cdot (x_1, x_2) = (b \cdot x_1, b \cdot x_2)$. This is the sum of B-torsors over G, denoted $X_1 + X_2$. Notice the similarity to Baer sum.

12.3.1 An extension gives a multiplicative torsor

Suppose we have a central extension of groups

$$0 \longrightarrow B \xrightarrow{j} X \xrightarrow{\pi} G \longrightarrow 0. \tag{12.6}$$

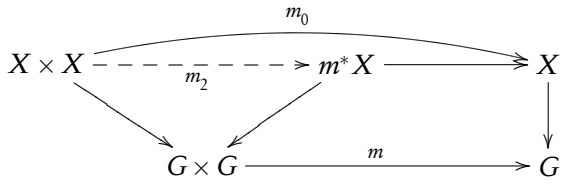
Then, in particular, there is a free action of B on X (denote this by $b \cdot x$) such that G is the quotient of that action under π, i.e., X is a B-torsor over G. Both G and X have group laws, and these are compatible in the sense that the following diagram commutes:

$$\begin{array}{ccc} X \times X & \xrightarrow{m_0} & X \\ \downarrow & & \downarrow \\ G \times G & \xrightarrow{m} & G \end{array} \tag{12.7}$$

Let $p_1, p_2 : G \times G \to G$ be the left and right projection maps, respectively. We claim that the multiplication m_0 determines a map

$$p_1^* X \times_{G \times G} p_2^* X \xrightarrow{m_1} m^* X \tag{12.8}$$

between sets over $G \times G$, and vice versa. We see this as follows. First, the dotted map making the following diagram commute is unique by the universal property of the fibre product:



Hence, m_2 and m_0 determine each other. Similar statements hold for p_1^*X and p_2^*X. Thus, the two projection maps on X give rise to maps $X \times X \to p_1^*X$ and $X \times X \to p_2^*X$. We claim that these two maps make $X \times X$ the fibre product of p_1^*X and p_2^*X over $G \times G$. If that claim holds, then m_2 and the map m_1 determine each other.

The claim will follow when we now look a little more closely at the map m_1. We have

$$\begin{aligned} p_1^*X \times_{G\times G} p_2^*X &\cong \{(e,g_1,g_2,f,h_1,h_2) : e,f \in X, g_i, f_i \in G, \pi(e)=g_1, \pi(f)=h_2, g_1=g_2, h_1=h_2\} \\ &\cong \{(e,f,g,h) : e,f \in X, g,h \in G, \pi(e)=g, \pi(f)=h\} \\ &\cong \{(e,f) : e,f \in X\} \cong X \times X \end{aligned} \tag{12.9}$$

which has two natural actions of B (one for each coordinate). We also have

$$m^*X \cong \{(e,g,h) : e \in X, g,h \in G, \pi(e)=gh\},$$

which has a single action of B. Using (12.9), the map m_1 is such that

$$m_1((e,f,g,h)) = (m_0(e,f),g,h),$$

The map m_1 inherits a respect for the actions of B from m_0, which satisfies

$$m_0(b_1 \cdot e_1, b_2 \cdot e_2) = b_1 b_2 \cdot m_0(e_1,e_2).$$

We have the equality $m_1((e,f,g,h)) = m_1((e',f',g',h'))$ if and only if $m_0(e,f) = m_0(e',f'), g=g', h=h'$. Therefore, m_1 is the quotient by the action $b \cdot (e,f,g,h) = (b \cdot e, b^{-1} \cdot f, g, h)$. In other words, m_1 maps onto the B-torsor $p_1^*X + p_2^*X$ over $G \times G$. (Recall that the sum of two B-torsors over a base is by definition exactly the fibre product of the two over that base, modulo the 'anti-diagonal' action of B described.) The map m_1 inherits surjectivity from m_0. That means we have an isomorphism

$$\mu : p_1^*X + p_2^*X \to m^*X. \tag{12.10}$$

The important point is this: to give such an isomorphism is to give a map m_0 in diagram (12.7).

For any subset I of $\{1,2,3\}$, we define a map $m_I : G \times G \times G \to G$ to be the multiplication of the coordinates I, i.e., $m_{12} = m \times id$ and $m_{123} = m \circ m_{12} = m \circ m_{23}$. Similarly, we define projections p_I on $G \times G \times G$. Then the associativity of multiplication m_0 produces an associativity of μ, which is to say we have a commutative diagram of B-torsors over $G \times G$:

$$\begin{array}{ccc} p_1^*X + p_2^*X + p_3^*X & \longrightarrow & p_{12}^*m^*X + p_3^*X \\ \downarrow & & \downarrow \\ p_1^*X + p_{23}^*m^*X & \longrightarrow & m_{123}^*X \end{array} \tag{12.11}$$

We define a *multiplicative B-torsor over a group G* to be a B-torsor X over a group G together with an isomorphism (12.10) satisfying the commutative diagram (12.11). We have thus described how a central extension gives a multiplicative torsor. We now wish to demonstrate the other direction.

12.3.2 A multiplicative torsor gives an extension

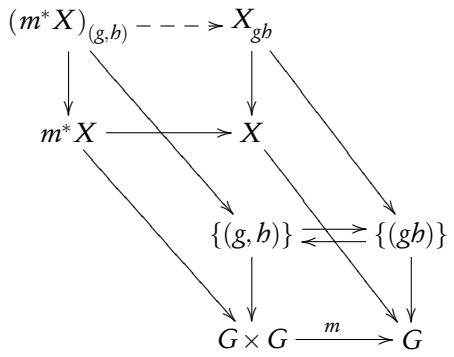
Suppose we have a multiplicative B-torsor X over a group G provided with a multiplication (12.10) satisfying (12.11). As sets, we have an extension (12.6). We wish to endow X with the structure of a group and show that π and j are homomorphisms. We can reverse the construction above, obtaining a map m_1 from μ and from that a map $m_0 : X \times X \to X$ satisfying (12.7) (so, if m_0 is a group law, the map π is a homomorphism). We already have associativity of the multiplication μ by the description above, and this translates to associativity of m_0.

Now, we wish to describe the identity and inverses of this composition law, making it a group law. Finally we show that j is a homomorphism.

The first step is to see what happens when we restrict to fibres: we have

$$(p_1^*X + p_2^*X)_{(g,h)} \xrightarrow{\cong} (m^*X)_{(g,h)}. \tag{12.12}$$

Let $g, h \in G$. The fibres X_{gh} and $(m^*X)_{(g,h)}$ over the points $gh \in G$ and $(g,h) \in G \times G$ respectively are defined by fibre products. These form two vertical faces of the following cubical diagram.



In this diagram, there are commuting maps from $(m^*X)_{(g,h)}$ to G which factor through the point gh and through X, which, by the universal property of the fibre product X_{gh} gives a unique map $(m^*X)_{(g,h)} \to X_{gh}$. Now, $(m^*X)_{(g,h)}$ is actually the fibre product of X_{gh} and $G \times G$. For, suppose some H has maps to X_{gh} and $G \times G$. Then it has a unique map to $\{gh\}$ and by fibre product of the front face, a unique map to $\{(g,h)\}$. It has unique maps to X and by fibre product of the floor of the cube, to m^*X. But then by fibre product of the left face, it has a unique map to $(m^*X)_{(g,h)}$, demonstrating the universal property of fibre products. Since X_{gh} has maps to itself and to $G \times G$, it has a unique map to $(m^*X)_{(g,h)}$ and so the dotted line is an isomorphism. This is what they mean when they say 'by abstract nonsense'. Anyway, by exactly similar abstract nonsense, there are unique isomorphisms

$$(m^*X)_{(g,h)} \cong X_{gh}, \qquad \text{and} \qquad (p_1^*X + p_2^*X)_{(g,h)} \cong X_g + X_h.$$

Thus, the isomorphism (12.10) determines a unique isomorphism

$$X_g + X_h \xrightarrow{\cong} X_{gh} \tag{12.13}$$

for each choice of $g, h \in G$.

Now we wish to find an identity element for X. We have an isomorphism,

$$X_1 + X_h \to X_h.$$

In other words, this gives a free, transitive action of X_1 on X, for any element of X_1 gives an automorphism of X_h. Therefore it gives an identification between X_1 and B. Let us denote by 1_X the element corresponding to the identity in B. Then its action on X_h is trivial for all $h \in G$ and it is an identity element. We also have an isomorphism

$$X_1 + X_1 \to X_1.$$

which indicates that m_0 restricts to a group law on the fibre X_1. The fibre over X_1 is the kernel of the map π, and this identification is exactly the map j, which is injective. So once we show that m_0 is a group law, we will know that j is a homomorphism.

All that remains is the existence of inverses. We have an isomorphism

$$X_g + X_{g^{-1}} \to X_1$$

So, in particular, for any element $x \in X$, let $g = \pi(x)$ and consider $x \in X_g$. Then choose $y \in X_{g^{-1}}$ such that $(x, y) \mapsto 1_X$. This is the inverse of x.

This gives an equivalence of categories, and so multiplicative B-torsors over G form a group. The group law, on the torsor side, is just torsor addition.

Chapter 13

Generalised Jacobians

We will now introduce generalised Jacobians, as described by Rosenlicht [57, 58] (and exposited by Serre [60]). The first section introduces some basic notation for divisors. Then, the generalised Jacobian is introduced. The case of an elliptic curve and a modulus of two points will be our particular concern, and in this case the factor set is made explicit. In the last two sections, this is shown to be equivalent to a group extension arising from a divisor in the Picard group of the curve: the first section shows that the generalised Jacobian is an algebraic group, and that a rational factor set suffices to determine an algebraic extension; in the second we show that the extension formed from a line bundle has a factor set agreeing with that of Rosenlicht's generalised Jacobian. Our interest all along is in understanding the explicit computations, which we will use later.

13.1 Divisors and Weil reciprocity

Suppose D is a divisor on a product $U \times V$. If $\mathfrak{a} = \sum_P n_P(P)$ is a cycle on U (i.e., a formal sum of points of U), then we can write

$$D|_{U\times\mathfrak{a}} = \sum_P n_P D|_{U\times\{P\}},$$

where $D|_{U\times\{P\}}$ denotes the divisor on $U \times \{P\}$ which is the restriction of D to that fibre.

If $D \sim 0$, then we write f_D for a rational function on $U \times V$ whose divisor is D (this notation has an ambiguity, but we shall never use it in a context where the particular choice of such a function matters).

Suppose D is a principal divisor on U, and $\mathfrak{a}$ a cycle of degree zero. A divisor on a zero-dimensional variety doesn't mean much, so we appropriate the notation $D|_{\mathfrak{a}}$ to mean $f_D(\mathfrak{a})$. In other words

$$D|_{\mathfrak{a}} = \prod_P f_D(P)^{n_P}.$$

Using this notation, we have the following generalisation of Weil reciprocity:

Theorem 13.1.1 ([38, VI §4 Thm. 9]). *Let A and B be two abelian varieties, and let D be a divisor on $A \times B$. Let $\mathfrak{a}$ and $\mathfrak{b}$ be two cycles of degree zero on A and B respectively, and whose sums are zero. Suppose no point of $\mathfrak{a} \times \mathfrak{b}$ is contained in D. Then the values*

$$f_{D|_{A\times\mathfrak{b}}}(\mathfrak{a}), \text{ and } \quad f_{D|_{\mathfrak{a}\times B}}(\mathfrak{b})$$

are defined and equal. We denote both by $D|_{\mathfrak{a}\times\mathfrak{b}}$. If D is principal, we may relax the condition that $\mathfrak{a}$ and $\mathfrak{b}$ have sum zero, in which case both of the above are still defined and are equal to $f_D(\mathfrak{a} \times \mathfrak{b})$ (justifying our choice of notation).

The more usual statement of Weil Reciprocity is a corollary.

Corollary 13.1.2. *Suppose that f and g are rational functions on a curve, whose divisors have disjoint support. Then $f((g)) = g((f))$.*

For proofs, see Lang [38, VI §4].

13.2 Generalised Jacobians

Let C be a projective algebraic non-singular irreducible curve defined over a field k. Let $\mathrm{Pic}(C)$ be the group of divisors of C modulo linear equivalence, and let $\mathrm{Pic}^0(C)$ be the subgroup of classes of divisors of degree zero. Recall that there is an abelian variety, the Jacobian $\mathcal{J}(C)$, associated to the curve C, and this variety is isomorphic to $\mathrm{Pic}^0(C)$ as a group. In this section we wish to generalise the notion of Jacobian to certain singular curves in a particular way. First, we describe a group extension of $\mathrm{Pic}^0(C)$ which depends on a choice of *modulus* $\mathfrak{m}$ (a divisor on C). Then, we state that this group is isomorphic to an algebraic group $\mathcal{J}_\mathfrak{m}(C)$. For details, see Serre's *Algebraic Groups and Class Fields* [60] or the original papers of Rosenlicht [57, 58].

Let $\mathfrak{m}$ be an effective divisor $\sum n_P(P)$ on C, which we call the *modulus*. For a rational function f on C, we write

$$f \equiv 1 \mod \mathfrak{m}$$

if $v_P(1-f) \geq n_P$ for every P in the support of $\mathfrak{m}$.

Then we say that two divisors D and D' on A with support disjoint from that of $\mathfrak{m}$ are $\mathfrak{m}$-equivalent if there exists a non-zero rational function f with divisor $D - D'$ and such that $f \equiv 1$ modulo $\mathfrak{m}$. We denote this by $D \sim_\mathfrak{m} D'$. The notion of $\mathfrak{m}$-equivalence is a refinement of linear equivalence. We define $\mathrm{Pic}_\mathfrak{m}(C)$ to be the group of $\mathfrak{m}$-equivalence classes of divisors with support disjoint from $\mathfrak{m}$, and $\mathrm{Pic}^0_\mathfrak{m}(C)$ to be the subgroup consisting of classes of degree zero. For the case of the trivial divisor $\mathfrak{m}$, $\mathfrak{m}$-equivalence is taken to be the usual notion of rational equivalence, so that $\mathrm{Pic}_\mathfrak{m}(C) = \mathrm{Pic}(C)$ and $\mathrm{Pic}^0_\mathfrak{m}(C) = \mathrm{Pic}^0(C)$.

Let $\pi : \mathrm{Pic}^0_\mathfrak{m}(C) \to \mathrm{Pic}(C)$ be the obvious map. Let $L_\mathfrak{m}$ be the kernel. In this way $\mathrm{Pic}^0_\mathfrak{m}(C)$ is an extension of $\mathrm{Pic}^0(C)$ by an algebraic group:

$$0 \longrightarrow L_\mathfrak{m} \xrightarrow{j} \mathrm{Pic}^0_\mathfrak{m}(C) \xrightarrow{\pi} \mathrm{Pic}^0(C) \longrightarrow 0 \tag{13.1}$$

13.3 The case of an elliptic curve with modulus $\mathfrak{m} = (S) + (T)$

Serre and Rosenlicht go on to describe the structure of these extensions in detail, but our concern for the moment will be with elliptic curves, and more specifically, with the case for which it turns out that $L_{\mathfrak{m}} \cong \mathbb{G}_m$. This is the case where $C = E$ is an elliptic curve and $\mathfrak{m} = (S) + (T)$ for two distinct points $S, T \in E$. In this section we will calculate the factor set for the generalised Jacobian $\mathrm{Pic}^0_{\mathfrak{m}}(E)$ explicitly. For further details on the explicit approach we take here, see [14].

In order to calculate the factor set, we must choose a section σ to π in (13.1). Denote by $\mathrm{Div}^0_{\mathfrak{m}}(E)$ divisors on E of degree zero and support disjoint from $\mathfrak{m}$ and denote by $\mathrm{Div}^0(E)$ divisors on E of degree zero. For any divisor $D = \sum_P n_P(P)$, let $s(D) = \sum_P [n_P]P$. We will choose a section σ to π in (13.1) by first defining

$$\sigma : \mathrm{Div}^0(E) \to \mathrm{Div}^0_{\mathfrak{m}}(E).$$

Associate to every $P \in E$ some point R_P such that neither R_P nor $s(D) + R_P$ is in the support of $\mathfrak{m}$. Then we can define

$$\sigma(D) = \left(s(D) + R_{s(D)}\right) - \left(R_{s(D)}\right) \tag{13.2}$$

and be assured that the supports of $\sigma(D)$ and $\mathfrak{m}$ are disjoint. Since $R_{s(D)}$ depends only on the sum of D, it is invariant across any equivalence class in $\mathrm{Pic}^0(E)$. For convenience, we define $R_D := R_{s(D)}$. Therefore σ induces a map which we will call by the same name

$$\sigma : \mathrm{Pic}^0(E) \to \mathrm{Pic}^0_{\mathfrak{m}}(E)$$

taking $[D]$ to $[\sigma(D)]_{\mathfrak{m}}$. This σ is a section to π in (13.1) and we intend to describe the factor set it gives.

We can define the factor set $F_\sigma : \mathrm{Pic}^0(E) \times \mathrm{Pic}^0(E) \to L_{\mathfrak{m}}$ by

$$F_\sigma(D_1, D_2) = \sigma(D_1) + \sigma(D_2) - \sigma(D_1 + D_2).$$

This divisor is linearly equivalent to zero and can be written

$$\left(s(D_1) + R_{D_1}\right) - \left(R_{D_1}\right) + \left(s(D_2) + R_{D_2}\right) - \left(R_{D_2}\right) - \left(s(D_1 + D_2) + R_{D_1+D_2}\right) + \left(R_{D_1+D_2}\right). \tag{13.3}$$

We define a little useful notation: denote by $\mathcal{D}(P, Q)$ the divisor $(P) + (Q) - (P + Q) - (\mathcal{O})$ and denote by $h_{P,Q}$ the rational function with this divisor. Then, the divisor becomes

$$\mathcal{D}\left(s(D_1), s(D_2)\right) + \mathcal{D}\left(s(D_1) + s(D_2), R_{D_1+D_2}\right) - \mathcal{D}\left(s(D_1), R_{D_1}\right) - \mathcal{D}\left(s(D_2), R_{D_2}\right),$$

and we have

$$F_\sigma(D_1, D_2) = \left(\frac{h_{s(D_1), s(D_2)} h_{s(D_1)+s(D_2), R_{D_1+D_2}}}{h_{s(D_1), R_{D_1}} h_{s(D_2), R_{D_2}}} \right). \tag{13.4}$$

Note that this divisor has disjoint support from $\mathfrak{m}$.

Proposition 13.3.1. *The map*

$$F_\sigma : E \times E \to L_{\mathfrak{m}}$$

is a factor set. Hence, we have an extension of groups

$$0 \longrightarrow L_{\mathfrak{m}} \xrightarrow{j} \mathrm{Pic}^0_{\mathfrak{m}}(E) \xrightarrow{\pi} \mathrm{Pic}^0(E) \longrightarrow 0 \tag{13.5}$$

Proof. The verification of (12.5) is straightforward. □

The kernel $L_{\mathfrak{m}}$ consists of classes $[D]_{\mathfrak{m}}$ such that $D = (f)$ and the supports of D and $\mathfrak{m}$ are disjoint. There is a map

$$L_{\mathfrak{m}} \xrightarrow{\phi} \mathbb{G}_m$$

given by $(f) \mapsto f(S)/f(T)$. The kernel of this map is zero. For, suppose $\phi([(f)]_{\mathfrak{m}}) = \phi([(g)]_{\mathfrak{m}})$. Without loss of generality we may choose f and g such that $f(T) = g(T) = 1$, so this implies that

$$\begin{aligned} f(S)/g(S) = 1 \quad &\Longleftrightarrow \quad (f/g)(S) = (f/g)(T) = 1 \\ &\Longleftrightarrow \quad (f/g) \equiv 1 \mod \mathfrak{m} \\ &\Longleftrightarrow \quad [(f)]_{\mathfrak{m}} = [(g)]_{\mathfrak{m}} \end{aligned}$$

This map is also surjective. A quick way to see this is to take the rational function

$$f_a : E \to \mathbb{P}^1, \qquad f_a(B) = \frac{x(B) - a}{x_T - a}$$

where x_T is the x-coordinate of the fixed point T and $x(B)$ denotes taking the x coordinate of the variable B. Then $f_a(T) = 1$ for any a but $f_a(S)$ can be forced to become any desired value of $\mathbb{G}_m$ by taking an appropriate value for a.

Finally, recall that $\mathrm{Pic}^0(E) \cong E$ by the pair of maps $P \mapsto [(P) - (\mathcal{O})]$ and $[\sum_P n_P(P)] \mapsto \sum_P [n_P]P$.

Thus we may choose to consider $\mathrm{Pic}^0_{\mathfrak{m}}(E)$ as the extension

$$0 \longrightarrow \mathbb{G}_m \xrightarrow{j} \mathrm{Pic}^0_{\mathfrak{m}}(E) \xrightarrow{\pi} E \longrightarrow 0$$

given by the factor set

$$\hat{F}_\sigma(P, Q) = \frac{F_\sigma(P, Q)(S)}{F_\sigma(P, Q)(T)}. \tag{13.6}$$

In particular, we can consider $\mathrm{Pic}^0_{\mathfrak{m}}(E)$ to be the group $\mathbb{G}_m \times E$ with the operation '$\oplus$' given by

$$(a, P) \oplus (b, Q) \quad = \quad (ab\hat{F}_\sigma(P, Q), P + Q)$$

Importantly, we have not yet shown that $\mathrm{Pic}^0_{\mathfrak{m}}(E)$ is an *algebraic* group.

Let us momentarily return to considering general C and $\mathfrak{m}$. In that case, we may also describe $\mathrm{Pic}^0_{\mathfrak{m}}(C)$ as a group extension by an algebraic group $L_{\mathfrak{m}}$, and it is isomorphic to a variety which we denote $X_{\mathfrak{m}}$. The dimension of this variety is $g + \deg(\mathfrak{m}) - 1$ for nontrivial $\mathfrak{m}$ and $L_{\mathfrak{m}}$ is an algebraic group isomorphic to a product of a torus and a unipotent group of a certain form. For details, consult Serre [60].

13.4 Rational sections and algebraic groups

In this section, we follow Serre [60, VII §1.4]. Suppose that A and B are commutative connected algebraic groups. A rational map $f : A \times A \to B$ satisfying the cocycle condition (12.5) (recall that the action of A is trivial) is called a *rational factor set.* A rational factor set is trivial if it is the image of a rational map $g : A \to B$ under the coboundary map. We similarly define regular factor sets. Rational and regular functions are closed under addition, hence we have groups $H^2_{rat}(A,B)$ and $H^2_{reg}(A,B)$. If we further assume that the factor sets are symmetric, we define $H^2_{rat}(A,B)_s$ and $H^2_{reg}(A,B)_s$. Finally, define $Ext(A,B)_{rat}$ and $Ext(A,B)_{reg}$ to be the subgroups of $Ext(A,B)$ given by extensions admitting a rational (respectively regular) section.

Theorem 13.4.1 ([60, VII §1.4 Prop. 4])**.**

1. *The group $H^2_{reg}(A,B)_s$ is isomorphic to $Ext(A,B)_{reg}$.*
2. *The group $H^2_{rat}(A,B)_s$ is isomorphic to $Ext(A,B)_{rat}$.*

Proof. Part 1: Restricting the isomorphism $Ext(A,B) \cong H^2(A,B)_s$ explained in Section 12.2, we have an injective homomorphism $\theta : Ext(A,B)_* \to H^2_{reg}(A,B)_s$ given by taking any regular section $g : A \to X$ and defining a factor set $f : A \times A \to B$ by $f(x,y) = g(x) + g(y) - g(xy)$ (clearly the latter is also regular). If f is a regular factor set, then we can define the associated extension as $B \times A$ with the group law

$$(b_1, a_1) \oplus (b_2, a_2) = (b_1 + b_2 + f(a_1, a_2), a_1 + a_2).$$

The map $a \mapsto (1, a)$ is a regular section. So θ is a bijection.

In Part 2 we will several times use the following fact. Claim: *A rational homomorphism between algebraic groups is a regular homomorphism* [60, V. §1 no. 5 Lemma 6]. To show this, suppose $f : G_1 \to G_2$ is such a map, so it is a regular homomorphism on a non-empty open subset $U \subset G_1$. That is, $f(x+y) = f(x) + f(y)$ whenever $x, y, x+y \in U$. Fixing any $x \in U$ and varying y, we see that f is regular on $x + U$. But the $x + U$ cover G_1, so f is regular.

Part 2: We again have a homomorphism $\theta : Ext(A,B)_{rat} \to H^2_{rat}(A,B)_s$. This time, to show injectivity, we must show that having a rational section which is a homomorphism implies having a section (everywhere defined) which is a homomorphism. The Claim provides this step.

It remains to demonstrate surjectivity. As before, we can define a law of composition on $A \times B$ which is rational, making $A \times B$ a birational group. By the results of Weil (see [60, V. §1] or [76]), $A \times B$ is birationally equivalent to a connected commutative algebraic group X by a pair of maps

$$F : A \times B \to X, \qquad G : X \to A \times B,$$

which commute with the defined composition laws (i.e., is a homomorphism where defined).

Then $p_1 \circ G : X \to A$ is a surjective homomorphism (by the Claim). We now define a homomorphism from $\phi : B \to X$ as follows. Let $b \in B$, and choose $a \in A, b' \in B$ such that F is defined at $(a, b+b')$ and (a, b'). Then, set $\phi(b) = F(a, b+b') - F(a, b')$. Then ϕ is independent of the choice of a and b'

by the condition that F respects the composition laws, and is a homomorphism where defined (hence regular by the Claim). We have obtained maps

$$1 \longrightarrow B \xrightarrow{\phi} X \xrightarrow{p_1 \circ G} A \longrightarrow 1.$$

This is exact by the construction. A rational factor set for this extension is f. As in Part 1, a section is given by $x \mapsto F(1,x)$, which is rational. □

In particular, giving a rational section of A by B is enough to determine an extension of algebraic groups. As a corollary we have the following:

Proposition 13.4.2. *If A and B are algebraic groups, then a rational factor set extends uniquely to a factor set defined everywhere.*

Proof. By the bijection above, a rational factor set gives an extension of algebraic groups admitting a rational section. Furthermore, this rational section gives the rational factor set we began with. Therefore, extend this rational section to a section defined on the entire group A, and the associated factor set must agree with the rational factor set where they are both defined. It remains to show uniqueness. But this comes of the factor set condition. For any a, b, c, d, we have (by two applications of the factor set condition):

$$f(a,b) = \frac{f(a-d,b-c)f(d,b+a-d-c)f(a+b-c,c)}{f(d,a-d)f(b-c,c)}.$$

Since f is rational, it is defined on an open subset of $A \times A$. Therefore there are some points c and d such that the right hand side of the above is defined. This gives uniqueness. □

Now, let us return to the notation of the last section, where $\mathfrak{m} = (S) + (T)$ on an elliptic curve E. Both $L_{\mathfrak{m}}$ and $\mathrm{Pic}^0(E)$ of the last section are algebraic groups (the first is $\mathbb{G}_m$ as described, and the second is the Jacobian of the curve). Fix a value $R_0 \neq S, T$ and suppose we choose the section giving the extension $\mathcal{J}_{\mathfrak{m}}$ in such a way that the $R_D = R_0$ almost everywhere (i.e., unless one of $P+R_0$, $Q+R_0$, $P+Q+R_0$, or R_0 is in $\{S,T\}$). We set the notation $h_{P,Q,R}$ for a rational function with divisor

$$(P+R)+(Q+R)-(P+Q+R)-(R)$$

(slightly generalising the notation $h_{P,Q}$ used in Section 13.3). Then, for all but finitely many pairs P, Q, the factor set (13.4) takes value $F(P,Q) = (h_{P,Q,R_0})$. That is, as an extension by $\mathbb{G}_m$, the factor set is

$$F(P,Q) = \frac{h_{P,Q,R_0}(S)}{h_{P,Q,R_0}(T)}. \tag{13.7}$$

This gives a rational factor set for $\mathcal{J}_{\mathfrak{m}}$. Therefore Theorem 13.4.1 tells us that

Theorem 13.4.3. *Let $\mathfrak{m} = (S) + (T)$ on an elliptic curve E. Then $\mathrm{Pic}^0_{\mathfrak{m}}(E)$ is an algebraic group.*

Furthermore, we can show the following.

Proposition 13.4.4. *Let E be an elliptic curve, and let $\mathfrak{m} = (S) + (T)$ and $\mathfrak{m}' = (S') + (T')$. The two generalised Jacobians $\mathcal{J}_{\mathfrak{m}}(E)$ and $\mathcal{J}_{\mathfrak{m}'}(E)$ are equivalent if and only if $\mathfrak{m}' = \tau_M^* \mathfrak{m}$ for some point $M \in E$.*

Proof. As discussed above, choose $R_0 \neq S, T, S', T'$ and use it to make the rational factor sets F and F' respectively of $\mathcal{J}_{\mathfrak{m}}$ and $\mathcal{J}_{\mathfrak{m}'}$. (By the preceeding, throughout what follows we may deal with only with rational 2-cocycles and 2-coboundaries to demonstrate the result.) These satisfy

$$\frac{F(P,Q)}{F'(P,Q)} = \frac{h_{P,Q,R_0}(S) h_{P,Q,R_0}(T')}{h_{P,Q,R_0}(T) h_{P,Q,R_0}(S')}. \tag{13.8}$$

Now suppose that $\mathfrak{m}' = \tau_M^* \mathfrak{m}$. We have

$$\frac{F(P,Q)}{F'(P,Q)} = \frac{h_{P,Q,R_0}(S) h_{P,Q,R_0}(T+M)}{h_{P,Q,R_0}(T) h_{P,Q,R_0}(S+M)}.$$

By Weil reciprocity, this is

$$\frac{F(P,Q)}{F'(P,Q)} = \frac{h_{S-T,M,T}(P+R_0) h_{S-T,M,T}(Q+R_0)}{h_{S-T,M,T}(P+Q+R_0) h_{S-T,M,T}(R_0)},$$

which is a 2-coboundary in variables P and Q. Therefore the extensions are equivalent.

For the converse, suppose that the quotient of the factor sets is a 2-coboundary, so there's some rational function f such that

$$\frac{F(P,Q)}{F'(P,Q)} = \frac{f(P) f(Q)}{f(P+Q)},$$

So we can write for some constant c, that

$$\frac{F(P,Q)}{F'(P,Q)} = c \frac{f(P) f(Q)}{f(P+Q) f(\mathcal{O})},$$

Then f is a product of some functions h_{R_1,R_2,R_3}. Without loss of generality, we may assume it is exactly one such function: $f = h_{R_1,R_2,R_3}$. Then, we have

$$\frac{F(P,Q)}{F'(P,Q)} = c \frac{h_{R_1,R_2,R_3}(P) h_{R_1,R_2,R_3}(Q)}{h_{R_1,R_2,R_3}(P+Q) h_{R_1,R_2,R_3}(\mathcal{O})},$$

and again by Weil reciprocity,

$$\frac{F(P,Q)}{F'(P,Q)} = c \frac{h_{P,Q}(R_1+R_3) h_{P,Q}(R_2+R_3)}{h_{P,Q}(R_1+R_2+R_3) h_{P,Q}(R_3)}. \tag{13.9}$$

Now, by the first direction, we can without loss of generality translate $\mathfrak{m}$ and $\mathfrak{m}'$. So let us assume $T = R_3$ and $T' = R_2 + R_3$. Then, setting (13.8) equal to (13.9), we have

$$c \frac{h_{P,Q}(R_1+R_2+R_3)}{h_{P,Q}(R_1+R_3)} = \frac{h_{P,Q}(S)}{h_{P,Q}(S')}$$

for all P, Q. Let $D = (R_1 + R_2 + R_3) + (S') - (R_1 + R_3) - (S)$. Then we have shown that $f(D) = c^{-1}$ for all but finitely many rational functions f on E with disjoint support from D.

I claim that any divisor D such that $f(D)$ is constant for all but finitely many f must be $D = 0$. Suppose $D = \sum n_p(P)$ with support $\mathcal{S}$, then consider the functions

$$f_{(a_P)} = \sum_{P \in \mathcal{S}} a_P \prod_{Q \in \mathcal{S}, Q \neq P} (x - x_Q)$$

Then $f_{(a_P)}(D) = \prod_P a_P^{n_P}$. By varying the a_P we can obtain any value we like, unless the product is empty (i.e., $D = 0$).

Therefore $(S') - (S) = (R_1 + R_2 + R_3) - (R_1 + R_3)$, so $S = R_1 + R_3$ and $S' = R_1 + R_2 + R_3$. Thus $\mathfrak{m} = \tau_{R_2}^* \mathfrak{m}'$. □

13.5 Line bundles and extensions

Recall that $\mathrm{Pic}^0(A)$ for an abelian variety A is the collection of divisor classes which are translation invariant, i.e., $\tau_P^* D - D \sim 0$.

In Section 13.3, we constructed an extension of an elliptic curve E by $\mathbb{G}_m$ for every $\mathfrak{m} \in \mathrm{Div}(E)$ of the form $(S) + (T)$. There is another way to construct extensions of E by $\mathbb{G}_m$ or more generally extensions of any abelian variety by $\mathbb{G}_m$: any line bundle in $\mathrm{Pic}^0(A)$ 'is' such an extension.

Consider a line bundle L over an abelian variety A. Then the line bundle with the zero section removed (call it $\bar{L}$) forms a $\mathbb{G}_m$-torsor (the zero section must be removed so that $\mathbb{G}_m$ acts freely). Furthermore, the tensor product of line bundles corresponds exactly to the sum of $\mathbb{G}_m$-torsors (see Section 12.3).

If, furthermore, we assume that $L \in \mathrm{Pic}^0(A)$, then we have an isomorphism

$$p_1^* L \otimes p_2^* L \cong m^* L. \tag{13.10}$$

Considered as torsors, this is just an isomorphism

$$p_1^* \bar{L} + p_2^* \bar{L} \cong m^* \bar{L}.$$

When referring to torsors coming from line bundles, we will often use the line bundle notations. In what follows, the line bundle L, considered as a torsor, will become a multiplicative torsor, and therefore give an extension of A by $\mathbb{G}_m$.

We wish to fix a particular isomorphism of (13.10). We shall do this by giving a trivialisation of L at 0, which is to say, choosing an isomorphism between $\mathbb{A}^1$ and the fibre L_0 above 0:

$$L_0 \xrightarrow{\cong} \mathbb{A}^1 , \qquad x_1 \mapsto 1.$$

Tensoring with L_0, we obtain an isomorphism

$$L_0 \otimes L_0 \xrightarrow{\cong} L_0 , \qquad a x_1 \otimes b x_1 \mapsto ab x_1. \tag{13.11}$$

Any isomorphism of (13.10) restricts to an isomophism $L_0 \otimes L_0 \cong L_0$ (see Section 12.3); we choose to fix the isomorphism of (13.10) which restricts to the one given in (13.11).

We wish to show that we have a commutative diagram of isomorphisms

$$\begin{array}{ccc} p_1^*L \otimes p_2^*L \otimes p_3^*L & \longrightarrow & p_{12}^*m^*L \otimes p_3^*L \\ \downarrow & & \downarrow \\ p_1^*L \otimes p_{23}^*m^*L & \longrightarrow & m_{123}^*L \end{array}$$

This is an equality of two isomorphisms. Restrict to $L_0 \otimes L_0 \otimes L_0$ and the two isomorphisms become the maps $ax_1 \otimes bx_1 \otimes cx_1 \mapsto abcx_1$. Since they restrict to the same map $L_0 \otimes L_0 \otimes L_0 \to L_0$, they are the equal.

Therefore, by Section 12.3, for every $L \in \mathrm{Pic}^0(A)$ equipped with a trivialisation at 0, we obtain a central extension.

Let's make the factor set explicit. Since any line bundle admits rational sections, the extension admits rational sections. Suppose we choose a section $g : A \to L$ defined on any open subset where the bundle is trivial. Associated with this rational section is a divisor D such that g is defined on $U = A \setminus \mathrm{supp}(D)$.

Let us set the notation $\lambda(D) = p_1^*D + p_2^*D - m^*D$. We have $\lambda(D) \sim 0$. Hence, there is a rational function f on $A \times A$ which has divisor $\lambda(D)$. This is a map $f_{\lambda(D)} : A \times A \to \mathbb{P}^1$. Thus, we define a rational factor set $F : A \times A \to \mathbb{G}_m$ which is just

$$F(P,Q) = f_{\lambda(D)}(P,Q). \tag{13.12}$$

Of course, we have a choice in $f_{\lambda(D)}$ to scale by a constant, but this gives an equivalent factor set. By definition, (13.12) is exactly the rational factor set associated to the rational section g.

Thus, we have constructed extensions of E by $\mathbb{G}_m$ in two different ways. It turns out these are the same, which we show by comparing their factor sets.

Proposition 13.5.1. *The equivalence class of extensions of E by $\mathbb{G}_m$ associated to the divisor $D = (S) - (T)$ is exactly the equivalence class of generalised Jacobians $\mathcal{J}_{\mathfrak{m}}$ for $\mathfrak{m} = (S) + (T)$.*

Proof. Let

$$C = -m_{123}^*(\mathcal{O}) + m_{12}^*(\mathcal{O}) + m_{23}^*(\mathcal{O}) + m_{13}^*(\mathcal{O}) - m_1^*(\mathcal{O}) - m_2^*(\mathcal{O}) - m_3^*(\mathcal{O}).$$

(Recall that m_I corresponds to the multiplication of the factors indexed by I, e.g. $m_{12} : E \times E \times E \to E$ is $(P_1, P_2, P_3) \mapsto P_1 + P_2$.) By the Theorem of the Cube, $C \sim 0$ on $A \times A \times A$. We can evaluate

$$C|_{\{(X)\} \times A \times A} = p_1^*(-X) + p_2^*(-X) - m_{12}^*(-X).$$

Let us set the convenient notation $\lambda(X) = p_1^*(X) + p_2^*(X) - m_{12}^*(X)$. We also have

$$C|_{A \times \{(X)\} \times \{(Y)\}} = (-X) + (-Y) - (-X - Y) - (\mathcal{O}).$$

Let $\mathfrak{a} = (-P, -Q)$ be a cycle on $A \times A$, and $\mathfrak{b} = (S) - (T)$ be a cycle on A. By Theorem 13.1.1, we have

$$f_{\lambda((S)-(T))}(P,Q) = f_{C|_{\mathfrak{b}\times A\times A}}(\mathfrak{a}) = f_{C|_{A\times\mathfrak{a}}}(\mathfrak{b}) = f_{P,Q}((S)-(T)).$$

This shows that both factor sets (13.7) and (13.12) form the same extension of E by $\mathbb{G}_m$. The equality above only holds on an open subset where it is defined, but by Proposition 13.4.2, this suffices to prove the theorem. □

Chapter 14

Biextensions

14.1 Definitions

We follow Mumford [50] in defining the notion of a biextension X of $B \times C$ by A (here, A, B and C are abelian groups, but X will have a more complicated structure, as we shall see). See also [8], [1, exp. VII], [54, §10.2]. We say X is a biextension of $B \times C$ by A if the following hold. First, A acts freely on X and there is a map

$$X \xrightarrow{\pi} B \times C$$

making $B \times C$ into the quotient of X by the action of A (as a set). Define the fibre product $X \times_B X$ by

$$\begin{array}{ccc} X \times_B X & \xrightarrow{\pi \circ p_2} & B \times C \\ \downarrow{\scriptstyle \pi \circ p_1} & & \downarrow{\scriptstyle p_1} \\ B \times C & \xrightarrow{p_1} & B \end{array}$$

where the p_i are projection maps. Define $X \times_C X$ similarly. There are two laws of composition

$$\begin{aligned} +_1 &: X \times_B X \to X, \\ +_2 &: X \times_C X \to X. \end{aligned}$$

These satisfy the requirements that for each $b \in B$, $X_b = \pi^{-1}(b \times C)$ is an abelian group under $+_1$ and π is a surjective group homomorphism of X_b onto C and has kernel isomorphic to A via the action of A on X_b. A parallel requirement holds for each $c \in C$. Furthermore, for elements $x, y, u, v \in X$ which π maps

$$x \mapsto (b_1, c_1), \quad y \mapsto (b_1, c_2), \quad u \mapsto (b_2, c_1), \quad v \mapsto (b_2, c_2),$$

we have

$$(x +_1 y) +_2 (u +_1 v) = (x +_2 u) +_1 (y +_2 v). \tag{14.1}$$

In brief, X is a pair of parametrised collections (X_b and X_c) of extensions of C and B respectively by A which satisfy some compatibility properties. From the definition above, it follows that for any fixed

$b \in B$, X_b is a group extension of C by A via the map π:

$$0 \longrightarrow A \longrightarrow X_b \xrightarrow{\pi} C \longrightarrow 0. \tag{14.2}$$

Finally, there is a natural notion of equivalence of biextensions, i.e., two biextensions X and X' are equivalent if they are isomorphic as sets under the action of A. We will denote the set of equivalence classes of biextensions by $\mathrm{Biext}(B \times C, A)$. In the next section we will show that this has the structure of a group.

14.2 Cohomology of biextensions

As for extensions, biextensions can be described by cocycles and coboundaries. Choose a section σ to $\pi : X \to B \times C$. Any element $x \in X$ can be written as $\sigma(b,c)^a$ for some $a \in A$, $b \in B$, and $c \in C$ (here, x^a denotes x acted on by a). Since A acts freely, this representation is unique, and so we have an isomorphism $X \cong A \times B \times C$ as sets under the action of A (on the right, a acts by multiplication on A and trivially on the other factors).

Restricting to a fixed $b \in B$, σ is a section to π in the extension (14.2), and has a factor set $f_{b,\sigma} : C \times C \to A$. As always, a parallel statement holds for C in place of B, and we can define

$$\begin{aligned} \phi &: B \times C \times C \to A, \\ \psi &: B \times B \times C \to A. \end{aligned}$$

by $\phi(b;c,c') = f_{b,\sigma}(c,c')$, and $\psi(b,b';c) = f_{c,\sigma}(b,b')$. In other words,

$$\begin{aligned} \phi(b;c,c') &= \sigma(b,c+c') - \sigma(b,c) - \sigma(b,c'), \\ \psi(b,b';c) &= \sigma(b+b',c) - \sigma(b,c) - \sigma(b',c). \end{aligned} \tag{14.3}$$

Here, we mean that the values on the right lie in the fibre over $(b,0)$ and $(0,c)$ respectively, each of which is identified with A since X_b and X_c are extensions of C and B by A respectively. Note that these extensions have trivial action of B and C on A, so the factor sets satisfy

$$\begin{aligned} \phi(b;c+c',c'') + \phi(b;c,c') &= \phi(b;c,c'+c'') + \phi(b;c',c''), \\ \psi(b+b',b'';c) + \psi(b,b';c) &= \psi(b,b'+b'';c) + \psi(b',b'';c). \end{aligned} \tag{14.4}$$

These factor sets give operations $+_1$, $+_2$ on $A \times B \times C$ by

$$(a,b,c) +_1 (a',b,c') = (a + a' + \phi(b;c,c'), b, c+c'), \tag{14.5}$$

$$(a,b,c) +_2 (a',b',c) = (a + a' + \phi(b,b';c), b+b', c) \tag{14.6}$$

Thus, we must make explicit the conditions on ϕ and ψ which correspond to the conditions on $+_1$ and $+_2$ in the definition of a biextension. First, we have the factor set conditions (14.4), but also $+_1$ and $+_2$ must be abelian group laws, which gives

$$\begin{aligned} \phi(b;c,c') &= \phi(b;c',c), \\ \psi(b,b';c') &= \psi(b',b;c). \end{aligned} \tag{14.7}$$

and the compatibility condition says that

$$\phi(b+b';c,c')-\phi(b;c,c')-\phi(b';c,c')=\psi(b,b';c+c')-\psi(b,b';c)-\psi(b,b';c'). \tag{14.8}$$

Thus, any section σ to π gives an associated *factor system* (ϕ,ψ). Furthermore, any factor system satisfying the conditions (14.4), (14.7), and (14.8) gives a biextension. Suppose σ and σ' are two sections for a given biextension, and have factor systems (ϕ,ψ) and (ϕ',ψ') respectively. Defining $\rho : B\times C\to A$ by

$$\rho(b,c)=a \text{ such that } \sigma'(b,c)=\sigma(b,c)^a,$$

one then obtains from the theory of group extensions and factor sets that

$$\begin{aligned}\phi'(b;c,c')-\phi(b;c,c')&=\rho(b,c+c')-\rho(b,c)-\rho(b,c'),\\ \psi'(b,b';c)-\psi(b,b';c)&=\rho(b+b',c)-\rho(b,c)-\rho(b',c).\end{aligned}$$

Therefore, we can use (14.3) to define a coboundary map from the group of functions $\rho : B\times C\to A$ to the group of factor systems (both under addition). Coboundaries are called *trivial factor systems*, and taking a quotient by this subgroup, we obtain a group $H^2_{bi}(B\times C,A)$ of factor systems.

Finally, in the case that two biextensions are equivalent, it is clear that they produce the same factor system. Hence, there is a bijection

$$\mathrm{Biext}(B\times C,A)\leftrightarrow H^2_{bi}(B\times C,A)$$

The collection of biextensions inherits its group structure from this isomorphism (we will not, as we did in the case of extensions, make the group law explicit on the side of biextensions).

14.3 Poincaré line bundle

Suppose A is an abelian variety and $\hat{A}$ is its dual. Mumford's motivating example for the definition of a biextension is the Poincaré line bundle on $A\times\hat{A}$ [50]. This is explained in greater detail in Milne [49, Ex C.1]. We will now describe this biextension in general, before moving on to the special case of an elliptic curve, where this biextension 'patches together' the generalised Jacobians we met in Section 13.2.

Let $i_a:\hat{A}\to A\times\hat{A}$ be the map $i_a(\hat{a})=(a,\hat{a})$ and $i_{\hat{a}}:A\to A\times\hat{A}$ be the map $i_{\hat{a}}(a)=(a,\hat{a})$. An abelian variety $\hat{A}$ is called the dual of the abelian variety A if there exists a divisor class $\mathcal{P}$ on $A\times\hat{A}$ such that the maps

$$\begin{aligned}&\hat{A}\to \mathrm{Pic}^0(A), && \hat{a}\mapsto i^*_{\hat{a}}(\mathcal{P}), && \text{and}\\ &A\to \mathrm{Pic}^0(\hat{A}), && a\mapsto i^*_a(\mathcal{P}),\end{aligned}$$

are both bijections. The divisor class $\mathcal{P}$ is called the *Poincaré divisor class*.

Every abelian variety has a unique dual and Poincaré divisor class (up to isomorphism). Consider the Poincaré line bundle which we will denote $\mathcal{P}$ (i.e., the line bundle associated to the Poincaré divisor

class), with its zero section removed. This is a $\mathbb{G}_m$-torsor over $A \times \hat{A}$, i.e., $\mathbb{G}_m$ acts freely on X and the quotient of this action is $A \times \hat{A}$. We will also denote this by $\mathcal{P}$ without confusion.

Proposition 14.3.1. *The Poincaré line bundle forms a biextension of $A \times \hat{A}$ by $\mathbb{G}_m$.*

Proof. Suppose we restrict $\mathcal{P}$ to $\mathcal{P}_{\hat{a}} = i_{\hat{a}}^* \mathcal{P}$ for $\hat{a} \in \hat{A}$. This is a line bundle over $A \times \{\hat{a}\} \cong A$, and in fact, is an element of $\mathrm{Pic}^0(A)$ by definition. Hence has the structure of a central extension with group law $+_{\hat{a}}$ (see Section 13.5). Similarly for $\mathcal{P}_a$ for $a \in A$, with group law $+_a$. These group laws are abelian. The only thing that requires checking is the compatibility property (14.1), which we leave as an exercise to the reader. □

We would like to write down a factor system for the Poincaré biextension. Our interest will be in the case that A is principally polarised and hence isomorphic to its dual $\hat{A}$. Recall [31, §A.7.3] that if A is principally polarised, then there exists a divisor class c such that $K(c) = 0$ where

$$K(c) = \{a \in A | \tau_a^* c = c\}.$$

Then a Poincaré divisor class is

$$D = p_1^* c + p_2^* c - m^* c.$$

Theorem 14.3.2. *Let A be a principally polarised abelian variety. Let c be a divisor class on A such that $K(c) = 0$. Let g be a rational function on $A \times A \times A$ with divisor*

$$C = m_{123}^* c - m_{12}^* c - m_{23}^* c - m_{13}^* c + m_1^* c + m_2^* c + m_3^* c.$$

Then, $\phi = \psi = g : A \times A \times A \to \mathbb{G}_m$ forms a rational factor system for the Poincaré biextension (where $\hat{A}$ is identified with A).

Proof. Let $\sigma : A \times A$ be a rational section to the Poincaré line bundle associated to this divisor. Then the factor system can be calculated from this section. As usual let $m_I : A \times A \times A \to A$ be multiplication of the indicated factors, and p_I on $A \times A \times A$ be the indicated projection maps onto one or several factors (note that $m_1 = p_1$ etc.). Then $\phi : A \times A \times A \to \mathbb{G}_m$ is given by a rational function with divisor

$$p_{12}^* D + p_{13}^* D - m_{23}^* D = p_1^* c + p_2^* c - m_{12}^* c + p_1^* c + p_3^* c - m_{13}^* c - p_1^* c - m_{23}^* c + m_{123}^* c = C$$

Note that ϕ and ψ must be equal by the symmetry of the biextension (the line bundle is symmetrical). □

14.4 Poincaré biextension for elliptic curves

For an elliptic curve E, a Poincaré divisor is

$$\mathcal{P} = m_{12}^*(\mathcal{O}) - p_1^*(\mathcal{O}) - p_2^*(\mathcal{O}).$$

The central extensions of E by $\mathbb{G}_m$ that arise from this biextension are those obtained from restricting the divisor $\mathcal{P}$ to $E \times \{D\}$ for some $D \in \mathrm{Pic}^0(E)$ (we identify the latter with $\hat{E}$ in the usual way). In particular, $D = (-P_0) - (\mathcal{O})$ for some $P_0 \in E$. Then, the restriction of $\mathcal{P}$ is

$$(P_0) - (\mathcal{O}).$$

As a result, the extension is exactly the extension of E by $\mathbb{G}_m$ which is given by the line bundle associated to $(P_0) - (\mathcal{O})$ or the modulus $\mathfrak{m} = (P_0) + (\mathcal{O})$.

Chapter 15

The elliptic net biextension is the Poincaré biextension

The first part of this chapter addresses some issues of basis, and lays the groundwork for the definition of the elliptic net biextension, which we then show to be equal to the Poincaré biextension. As a consequence, we deduce an extra additive structure on the Poincaré biextension.

15.1 The elliptic net biextension

We will now define a biextension of $E \times E \to \mathbb{G}_m$ using elliptic nets.

In the notation of Section 10.3, define

$$\Lambda(P,Q,R) = \frac{\mathcal{W}(p+q+r)\mathcal{W}(p)\mathcal{W}(q)\mathcal{W}(r)}{\mathcal{W}(p+q)\mathcal{W}(q+r)\mathcal{W}(r+p)} \in K^* \cong \mathbb{G}_m \tag{15.1}$$

By the results of Section 10.3, this is everywhere well-defined.

Theorem 15.1.1. *The function $\Lambda : E \times E \times E \to \mathbb{G}_m$ forms a biextension, and this biextension is the Poincaré biextension.*

Proof. We claim that $\Lambda(P,Q,R)$ gives a biextension in the sense that

$$\phi(P,Q;R) = \Lambda(P,Q,R) = \psi(P;Q,R)$$

is a factor system for a biextension.

To show that this is a factor system is very easy. On account of the symmetry of the definition in P,Q,R, the abelian property (14.7) is immediate, and the compatibility property (14.8) follows from

the factor set property (14.4). Thus it remains to verify the factor set property. We calculate

$$\begin{aligned}
&\Lambda(P+Q,R,S)\Lambda(P,Q,S)\\
&= \frac{\mathcal{W}(p+q+r+s)\mathcal{W}(p+q)\mathcal{W}(r)\mathcal{W}(s)\mathcal{W}(p+q+s)\mathcal{W}(p)\mathcal{W}(q)\mathcal{W}(s)}{\mathcal{W}(p+q+s)\mathcal{W}(r+s)\mathcal{W}(p+q+r)\mathcal{W}(p+s)\mathcal{W}(q+s)\mathcal{W}(p+q)}\\
&= \frac{\mathcal{W}(p+q+r+s)\mathcal{W}(p)\mathcal{W}(q)\mathcal{W}(r)\mathcal{W}(s)^2}{\mathcal{W}(p+q+r)\mathcal{W}(p+s)\mathcal{W}(q+s)\mathcal{W}(r+s)}
\end{aligned}$$

which symmetric in P,Q, and R. The factor set is also everywhere defined. Thus, it gives a biextension.

Wherever $P+Q+R, P, Q, R, P+Q, Q+R$, and $R+P$ do not vanish, the function

$$\frac{\mathcal{W}(p+q+r)\mathcal{W}(p)\mathcal{W}(q)\mathcal{W}(r)}{\mathcal{W}(p+q)\mathcal{W}(q+r)\mathcal{W}(r+p)}$$

is a rational function with divisor

$$m_{123}^*(\mathcal{O}) - m_{12}^*(\mathcal{O}) - m_{13}^*(\mathcal{O}) - m_{23}^*(\mathcal{O}) + m_1^*(\mathcal{O}) + m_2^*(\mathcal{O}) + m_3^*(\mathcal{O}),$$

hence it gives a rational factor system for the Poincaré biextension, by Theorem 14.3.2.

Now, we show that any factor system for an *algebraic* biextension (i.e., a biextension of algebraic groups $B\times C$ by an algebraic group A) which is defined on an open subset must extend uniquely to give a factor system defined everywhere. (The resulting factor system is not regular in general.) We will use Proposition 13.4.2. Call the factor system (ϕ,ψ). First look at all the left-slices $\{P\}\times E$ on which the factor system is rational; on all these it extends uniqely to some (ϕ',ψ'). However, we may miss some slices which are omitted entirely from the domain of (ϕ',ψ'). So we consider the right-slices $E\times\{P\}$ and the partially extended factor system (ϕ',ψ') will be rational on all these slices and so extends completely. □

A brief note on evaluation. Whenever P, Q and R form an appropriate triple (i.e., no two are equal or inverses, and no one is zero), then

$$\Lambda(p,q,r) = W_{P,Q,R}(1,1,1).$$

There are other special cases: for example, if $P=Q$, then we have

$$\Lambda(p,p,r) = \frac{W_{P,R}(2,1)}{W_{P,R}(2,0)}.$$

15.2 The Poincaré biextension has extra structure

As a consequence of Theorem 15.1.1, the Poincaré biextension has an extra additive structure, when considered as a K^*-torsor. The following theorem has not, as far as the author knows, been recorded elsewhere.

Theorem 15.2.1. *Let E be an elliptic curve defined over a field K. The Poincaré biextension for E admits a factor system (ϕ,ψ) of maps*

$$E(K)\times E(K)\times E(K) \to K\backslash\{0\}$$

such that

$$\phi(X_1, X_2; X_3) = \psi(X_1; X_2, X_3)$$

and such that

$$\phi(X_1, X_4 + X_2, -X_2) + \phi(X_2, X_4 + X_3, -X_3) + \phi(X_3, X_4 + X_1, -X_1) = 0$$

for all non-zero points $X_1, X_2, X_3, X_4 \in E(K)$ satisfying the condition that none of the expressions

$$X_4 + X_i \ (i = 1, 2, 3), \quad X_i - X_j \ (i, j = 1, 2, 3, i \neq j), \quad X_4 + X_i + X_j \ (i, j = 1, 2, 3, i \neq j)$$

vanishes.

Proof. Rewrite the elliptic net recurrence relation (3.1) in the following form (by multiplying by a factor):

$$\frac{W(p+q+s)\,W(p-q)\,W(s)}{W(p+s)\,W(q+s)\,W(p)\,W(q)} + \frac{W(q+r+s)\,W(q-r)\,W(s)}{W(q+s)\,W(r+s)\,W(q)\,W(r)} + \frac{W(r+p+s)\,W(r-p)\,W(s)}{W(r+s)\,W(p+s)\,W(r)\,W(p)} = 0.$$

Let $\phi = \psi = \Lambda^{-1}$ where Λ is defined by (15.1). This is a factor system for the Poincaré biextension by Theorem 15.1.1 since whenever (ϕ, ψ) is a factor system for a biextension, so is $(-\phi, -\psi)$ (in additive notation). Then, the recurrence relation above becomes (after multiplication by -1):

$$\phi(p, s+q, -q) + \phi(q, s+r, -r) + \phi(r, s+p, -p) = 0.$$

The statement follows. □

Question 15.2.2. *Does the Poincaré biextension on other principally polarized abelian varieties come equipped with an extra additive structure of this type? This is related to the question of generalising elliptic nets to other abelian varieties or Jacobians of curves.*

Chapter 16

Pairings

This chapter consists of background on pairings, specifically the Weil and Tate-Lichtenbaum pairings. We offer much more background than is strictly necessary to set notation for what follows.

16.1 The Weil pairing for elliptic curves

Consider the m-torsion points on an elliptic curve over $\mathbb{C}$ given by a lattice generated by 1 and τ. These are the points of the form $\frac{a}{m} + \frac{b}{m}\tau$ for $a, b \in \mathbb{Z}$. We define a pairing

$$e_m\left(\frac{a}{m} + \frac{b}{m}\tau, \frac{c}{m} + \frac{d}{m}\tau\right) = e^{\frac{i\pi(ad-bc)}{m}}$$

from $E[m] \times E[m]$ into the m-th roots of unity μ_m. Since the determinant is not dependent on the basis chosen (here, 1 and τ), this pairing is defined independent of that choice. This is the *Weil pairing*.

It can also be viewed as the intersection pairing on the homology group $H^1(E, \mathbb{Z})$. To do this, identify an m-torsion point with the path from the origin to that point, modulo $H^1(E, \mathbb{Z})$. Then, we have

$$E[m] \cong \left(\frac{1}{m}H^1(E, \mathbb{Z})\right) / H^1(E, \mathbb{Z}) \cong H^1(E, \mathbb{Z})/(mH^1(E, \mathbb{Z}))$$

But this latter is actually the group $H^1(E, \mathbb{Z}/m\mathbb{Z})$. If α and β generate $H^1(E, \mathbb{Z})$, then the intersection pairing is determined by the conditions

$$\alpha \cdot \beta = 1, \qquad \beta \cdot \alpha = -1, \qquad \alpha \cdot \alpha = \beta \cdot \beta = 0.$$

In particular,

$$(a\alpha + b\beta) \cdot (c\alpha + d\beta) = ad - bc.$$

So the intersection pairing on $H^1(E, \mathbb{Z}/m\mathbb{Z})$ takes values in $\mathbb{Z}/m\mathbb{Z}$, but under the exponential map, it takes values in μ_m and agrees with the pairing above.

These are meant to be informal definitions of the Weil pairing over $\mathbb{C}$, to be used as motivation (following [25]). Over a more general field, one usually uses the following definition. This section

follows Miller [47] and Silverman [63, Chap III, §8]. The Tate-Lichtenbaum and Weil pairings are defined in a more general setting for abelian varieties, but for the moment we restrict ourselves to the case of elliptic curves.

Definition 16.1.1 (Weil pairing: first definition)**.** Let $m > 1$ be an integer. Let E be an elliptic curve defined over a field K which contains the field of definition of $E[m]$, and with characteristic coprime to m. Suppose that $P, Q \in E[m]$. Choose divisors D_P and D_Q of disjoint support such that

$$D_P \sim (P) - (\mathcal{O}), \qquad D_Q \sim (Q) - (\mathcal{O}).$$

Then $mD_P \sim mD_Q \sim 0$, hence there are functions f_P and f_Q such that

$$(f_P) = mD_P, \qquad (f_Q) = mD_Q.$$

The Weil pairing

$$e_m : E[m] \times E[m] \to \mu_m$$

is defined by

$$e_m(P, Q) = \frac{f_P(D_Q)}{f_Q(D_P)}.$$

As an example, we can choose D_P and D_Q disjoint as follows: first choose some T such that $T \notin \{\mathcal{O}, -P, Q, Q-P\}$. Then set $D_P = (P+T) - (T)$ and $D_Q = (Q) - (\mathcal{O})$. Set the notation $f_{m,X}$ for the rational function with divisor $m(X) - m(\mathcal{O})$. Then,

$$e_m(P, Q) = \frac{f_P(D_Q)}{f_Q(D_P)} = \frac{f_P(Q) f_Q(T)}{f_P(\mathcal{O}) f_Q(P+T)} = \frac{f_{m,P}(Q-T) f_{m,Q}(T)}{f_{m,P}(-T) f_{m,Q}(P+T)}. \tag{16.1}$$

The last equality holds since $f_Q = f_{m,Q} \circ \tau_{-T}$.

Two definitions must surely be better than one.

Definition 16.1.2 (Weil pairing: second definition)**.** Let $m > 1$ be an integer. Let E be an elliptic curve defined over a field K which contains the field of definition of $E[m]$, and with characteristic coprime to m. Suppose that $P, Q \in E[m]$. Let g_P be a rational function such that

$$g_P^m = f_{m,P} \circ [m].$$

Choose P' such that $P = [m]P'$. We know such a g_P exists since it has divisor

$$[m]^*(P) - [m]^*(\mathcal{O}) = \sum_{R \in E[m]} (P' + R) - (R) \sim 0$$

The Weil pairing

$$e_m : E[m] \times E[m] \to \mu_m$$

is defined by

$$e_m(P, Q) = \frac{g_P(X+Q)}{g_Q(X)}.$$

Obviously we have some work to do.

Proposition 16.1.1. *Denote the algebraic closure of K by $\bar{K}$. Definitions 16.1.1 and 16.1.2 are well-defined, agree, and have the following properties:*

1. *Bilinearity: for $P, Q, R \in E[m]$,*

$$e_m(P+R,Q) = e_m(P,Q)e_m(R,Q),$$
$$e_m(P,Q+R) = e_m(P,Q)e_m(P,R).$$

2. *Alternating: for $P \in E[m]$,*

$$e_m(P,P) = 1.$$

3. *Skew-symmetry: for $P, Q \in E[m]$,*

$$e_m(P,Q) = e_m(Q,P)^{-1}.$$

4. *Non-degeneracy: for nonzero $P \in E[m](\bar{K})$, there exists $Q \in E[m](\bar{K})$ such that*

$$e_m(P,Q) \neq 1.$$

5. *Compatibility: for $P \in E[mn], Q \in E[m]$,*

$$e_{mn}(P,Q) = e_m(nP,Q).$$

6. *Galois invariance: for $P, Q \in E[m]$, and $\sigma \in \mathrm{Gal}(\bar{K}/K)$,*

$$e_m(P,Q)^\sigma = e_m(P^\sigma, Q^\sigma).$$

Proof. **Well-definition of first definition:** For now, consider the pairing as taking values in K^*. To see that the pairing is well-defined requires an application of Weil reciprocity. For, suppose we chose $D'_P \sim D_P$ and $D'_Q \sim D_Q$, and obtain functions f'_P and f'_Q such that $(f'_P) = mD'_P$ and $(f'_Q) = mD'_Q$. Then, $D'_P - D_P = (g_P)$ and $D'_Q - D_Q = (g_Q)$ for some rational functions g_P, g_Q. These can be chosen so that $f'_P = f_P g_P^m$ and $f'_Q = f_Q g_Q^m$.

$$\begin{aligned}
\frac{f'_P(D'_Q)}{f'_Q(D'_P)} &= \frac{f_P(D_Q)g_P(D_Q)^m f_P(g_Q)g_P(g_Q)^m}{f_Q(D_P)g_Q(D_P)^m f_Q(g_P)g_Q(g_P)^m} \\
&= \frac{f_P(D_Q)g_P(mD_Q)f_P(g_Q)}{f_Q(D_P)g_Q(mD_P)f_Q(g_P)} \\
&= \frac{f_P(D_Q)}{f_Q(D_P)}
\end{aligned}$$

Bilinearity: Now we show that the pairing is bilinear. Choose $P, Q, R \in E[m]$. Then $D_{P+R} = D_P + D_R + (g_{PR})$.

$$\begin{aligned} e_m(P+R,Q) &= \frac{f_{P+R}(D_Q)}{f_Q(D_{P+R})} \\ &= \frac{f_P(D_Q) f_R(D_Q) g_{PR}(D_Q)^m}{f_Q(D_P) f_Q(D_R) f_Q(g_{PR})} \\ &= e_m(P,Q) e_m(R,Q) \end{aligned}$$

The argument for the second factor is essentially identical.

Values in μ_m: A consequence of bilinearity is that

$$e_m(\mathcal{O},Q) = e_m(\mathcal{O},Q)^2,$$

which implies that

$$e_m(\mathcal{O},Q) = 1.$$

To show that the pairing takes values in μ_m, we check that

$$e_m(P,Q)^m = e_m(mP,Q) = e_m(\mathcal{O},Q) = 1$$

Alternating: Using the formula (16.1), it suffices to find a point T such that $T \neq \mathcal{O}, \pm P$ and such that $T = -T$. If $m \neq 2$, since $P \in E[m]$, any $T \in E[2]$ will suffice. If $m = 2$, and the characteristic of K is not 2, then there are three nontrivial points of order 2, so again we can choose $T \neq \pm P$.

Skew-symmetry: To show skew-symmetry, use bilinearity to calculate

$$e_m(P,Q) e_m(Q,P) = e_m(P+Q, P+Q) = 1$$

from the alternating property.

Equivalence of definitions: We wish to find a function g_P depending on point $P \in E[m]$ in such a way that

$$g_P^m = f_{m,P} \circ [m], \tag{16.2}$$

and

$$\frac{g_P(X+Q)}{g_P(X)} = e_m(P,Q), \tag{16.3}$$

where e_m here denotes the first definition of the Weil pairing (Definition 16.1.1).

To do so, first fix a value $T \neq \mathcal{O}, \pm P$, and define h_X to be the function with divisor

$$m(X) - (m-1)(T) - ([m]X - [m-1]T).$$

Then set

$$g_P(X) = \frac{f_{m,P}(X)}{h_X((P) - (\mathcal{O}))}.$$

Then

$$\begin{aligned} g_P(X)^m &= \frac{f_{m,P}(m(X))}{h_X(m(P)-m(\mathcal{O}))} \\ &= f_{m,P}(m(X)-(h_X)) \\ &= f_{m,P}((m-1)(T)+([m]X-[m-1]T) \\ &= f_{m,P}([m]X-[m-1]T) f_{m,P}(T)^{m-1}. \end{aligned}$$

Thus, replace g_P with a scalar multiple so that

$$g_P^m = f_{m,P} \circ \tau_{-[m-1]T} \circ [m].$$

Now, choose T to be $[m]R$ for some R, so that we obtain

$$g_P^m = f_{m,P} \circ [m] \circ \tau_{[1-m]R}.$$

We have not yet shown that (16.2) is satisfied. First, the divisor

$$(h_{X+Q})-(h_X) = m((Q+X)-(X))$$

is linearly equivalent to zero, so it is the divisor of some function $\hat{h}_{Q,X}$. Now

$$\begin{aligned} \frac{g_P(X+Q)}{g_P(X)} &= \frac{f_{m,P}(X+Q)h_X((P)-(\mathcal{O}))}{f_{m,P}(X)h_{X+Q}((P)-(\mathcal{O}))} \\ &= \frac{f_{m,P}((Q+X)-(X))}{\hat{h}_{Q,X}((P)-(\mathcal{O}))} \\ &= e_m(P,Q) \end{aligned}$$

Thus, we have given an equivalent definition for the Weil pairing. But this definition is independent of X, so (16.3) holds for $g_P \circ \tau_X$ for any X. Hence there is a function g_P satisfying (16.2) and (16.3). This alternate definition makes the remaining part of the proof easier.

Non-degeneracy: Under the new definition of the Weil pairing just given, non-degeneracy is a consequence of the fact that the map

$$E[m] \to \mathrm{Aut}[\bar{K}(E)/[m]^*\bar{K}(E)], \qquad T \mapsto \tau_T^*$$

is an isomorphism (see [63, Thm III.4.10 b)]). In particular, fix P and assume that $e_m(P,Q)=1$ for all $Q \in E[m]$. Then $g_P(X+Q)=g_P(X)$ for all $Q \in E[m]$, so $g_P = h \circ [m]$ for some $h \in \bar{K}(E)$. Hence

$$(h \circ [m])^m = g_P^m = f_{m,P} \circ [m]$$

implying that $f = h^m$. So then h has divisor $(P)-(\mathcal{O})$, implying $P=\mathcal{O}$.

Compatibility: We have

$$(g_P \circ [n])^{mn} = (f_{m,P} \circ [m] \circ [n])^n = f_{mn,P} \circ [mn].$$

Thus, to calculate e_{mn} instead of e_m, we replace g_P with $g_P \circ [n]$. Then

$$e_{mn}(P,Q) = \frac{g_P \circ [n](X+Q)}{g_P \circ [n](X)} = \frac{g_P([n]X+[n]Q)}{g_P([n]X)} = e_m(P,[n]Q).$$

Galois invariance: Let $\sigma \in \mathrm{Gal}(\bar{K}/K)$. Then

$$f_{m,P^\sigma} = f^\sigma_{m,P}, \qquad g_{P^\sigma} = g_P^\sigma.$$

So

$$e_m(P^\sigma,Q^\sigma) = \frac{g_P^\sigma(X^\sigma+Q^\sigma)}{g_P^\sigma(X^\sigma)} = \left(\frac{g_P(X+Q)}{g_P(X)}\right)^\sigma = e_m(P,Q)^\sigma.$$

□

16.2 Weil pairing via duality

The Weil pairing arises from the Cartier duality of the kernels of an isogeny and its dual. In this section we describe explicitly how the Weil pairing arises in this way for elliptic curves. The closest reference to this material is Mumford [51, IV.§20, p.183-5] and Milne [48, §11,16].

The duality arises as follows. Let us assume we are working over an algebraically closed field. Let $f : A \to B$ be an isogeny of abelian varieties with a finite kernel. Consider the exact sequence

$$1 \longrightarrow N \longrightarrow A \xrightarrow{f} B \longrightarrow 1.$$

This induces a long exact sequence

$$\cdots \longrightarrow \mathrm{Hom}(A,\mathbb{G}_m) \longrightarrow \mathrm{Hom}(N,\mathbb{G}_m) \longrightarrow \mathrm{Ext}^1(B,\mathbb{G}_m) \longrightarrow \mathrm{Ext}^1(A,\mathbb{G}_m).$$

It is the case that $\mathrm{Hom}(A,\mathbb{G}_m) = 0$. Under the interpretation of Cartier duality for abelian varieties and finite group schemes (the kernel N is one such), we obtain a short exact sequence

$$1 \longrightarrow \hat{N} \longrightarrow \hat{B} \xrightarrow{\hat{f}} \hat{A} \longrightarrow 1.$$

This sequence is exact on the right because $\dim \hat{A} = \dim \hat{B}$ and $\hat{N}$ is finite.

We will look more closely at exactly how the duality works for the sequence

$$1 \longrightarrow E[m] \longrightarrow E \xrightarrow{[m]} E \longrightarrow 1.$$

This is an extension of groups, and so any section σ to $[m]$ gives a factor set $F_m : E \times E \to E[m]$. The section is just a choice for each P of some $\sigma(P)$ such that $[m]\sigma(P) = P$. For convenience we will denote $\sigma(P)$ simply as P'.

We have several isomorphic descriptions of the dual of an elliptic curve:

$$\hat{E} \cong \mathrm{Pic}^0(E) \cong \mathrm{Ext}^1(E,\mathbb{G}_m).$$

The dual isogeny to $[m]$, which we write $\widehat{[m]}$, has a sequence

$$\begin{array}{ccccccccc} 1 & \longrightarrow & \mathrm{Pic}^0(E)[\widehat{m}] & \longrightarrow & \mathrm{Pic}^0(E) & \xrightarrow{\widehat{[m]}} & \mathrm{Pic}^0(E) & \longrightarrow & 1 \\ & & \cong\uparrow\downarrow & & \cong\uparrow\downarrow & & \cong\uparrow\downarrow & & \\ 1 & \longrightarrow & \mathrm{Hom}(E[n],\mathbb{G}_m) & \longrightarrow & \mathrm{Ext}^1(E,\mathbb{G}_m) & \longrightarrow & \mathrm{Ext}^1(E,\mathbb{G}_m) & \longrightarrow & 1 \end{array}$$

We wish to make explicit the isomorphism at the left, which, combined with the isomorphism $\mathrm{Pic}^0(E) \cong E$, gives the duality of the kernels. First, suppose that $(P+S)-(S) \in \mathrm{Pic}^0(E)[\widehat{m}]$ (so that $P \in E[m]$). The image of this under the injection into $\mathrm{Pic}^0(E)$ is associated with an extension of groups in $Ext^1(E,\mathbb{G}_m)$ via the middle isomorphism. This is just the generalised Jacobian associated with the modulus $\mathfrak{m} = (P+S)+(S)$ (by the general theory of duality as described in Chapters 13 and 14), and it gives rise to a factor set which we will call $F_P : E \times E \to \mathbb{G}_m$.

We have a diagram

$$\begin{array}{ccc} E[m] & \dashrightarrow & \mathbb{G}_m \\ {\scriptstyle F_m}\uparrow & \nearrow_{F} & \\ E \times E & & \end{array}$$

in which the dotted arrow and F_m together determine a factor set F: this is exactly the connecting homomorphism $\mathrm{Hom}(E[m],\mathbb{G}_m) \to Ext^1(E,\mathbb{G}_m)$ in the long exact sequence above. We wish to show that there exists a dotted arrow determining $F = F_P$ arising from the point P. Then we will have constructed a map from $E[m]$ to $\mathrm{Hom}(E[m],\mathbb{G}_m)$. We wish to then show that this map is an isomorphism. In doing so, we will use the fact that $P \in E[m]$ and Theorem 13.1.1. Finally, we will see that we have constructed a pairing that agrees with the Weil pairing.

Recall that the factor set F_P is given (in the notation of Section 13.2) by

$$F_P(P_0,Q_0) = h_{P_0,Q_0}((P+S)-(S)).$$

The map F_m has already been described. We now construct the dotted map, which we will call $\phi : E[m] \to \mathbb{G}_m$. Let g_P be as described in the definition of the Weil pairing (but using $(P+S)-(S)$ in place of $(P)-(\mathcal{O})$), and let

$$\phi(T) = \frac{g_P(T+X)}{g_P(X)} \text{ for } T \in E[m].$$

(Fortunately, we've already shown that this is well defined in Section 16.1.)

Next we show that the triangle commutes. Denote by $\Delta_{a,b}$ the divisor $(\{([a]P,[b]P)\})$ on $E \times E$. Let D be a divisor on $E \times E$ of the form

$$D = \Delta_{1,m} - \Delta_{m,1} + (m^2-1)(\{\mathcal{O}\} \times E) - (m^2-1)(E \times \{\mathcal{O}\}).$$

The divisor D is principal. Let $\mathfrak{a} = (P+S)-(S)$ and $\mathfrak{b} = ((P_0+Q_0)')-(P_0')-(Q_0')+(\mathcal{O})$ be divisors on E. Note that the latter divisor is linearly equivalent to $(Q+X)-(X)$ for $Q = (P_0+Q_0)'-P_0'-Q_0' \in E[m]$. Let $A = B = E$. In the language of Theorem 13.1.1, we can calculate

$$D|_{\mathfrak{a}\times B} = [m]^*((P+S)-(S)) - (mP+mS)-(mS) = [m]^*((P+S)-(S)).$$

We can also calculate

$$\begin{aligned} D|_{A\times\mathfrak{b}} &= -[m]^*[((P_0+Q_0)') - (P_0') - (Q_0') + (\mathcal{O})] + (m(P_0+Q_0)') - (mP_0') - (mQ_0') + (\mathcal{O}) \\ &= -[m]^*[((P_0+Q_0)') - (P_0') - (Q_0') + (\mathcal{O})] + (P_0+Q_0) - (P_0) - (Q_0) + (\mathcal{O}). \end{aligned}$$

Notice that the first part of $D|_{A\times\mathfrak{b}}$, that is, the part in the form of a pullback by $[m]^*$, is the divisor of a function which must vanish on $(P+S)-(S)$ since $[m]P=\mathcal{O}$. Therefore, we apply Theorem 13.1.1 to conclude that

$$F_P(P_0,Q_0) = f_{D|A\times\mathfrak{b}}(\mathfrak{a}) = f_{D|\mathfrak{a}\times B}(\mathfrak{b}) = g_P(\mathfrak{b}) = \phi(T).$$

There are a couple of small caveats here. First, we have used a definition for F_P which works only almost everywhere, i.e., a rational factor set (see Section 13.2). Second, we have used a divisor $\mathfrak{b}$ equivalent to $(Q+X)-(X)$ in the definition of $\phi(T)$. That the definition is invariant under a change of $(Q+X)-(X)$ by linear equivalence is a more general statement than that of the independence of the definition from X. We leave it to the reader to resolve these small issues.

Now we wish to show that the map we have constructed is an isomorphism. Note that any map $E[m]\to\mathbb{G}_m$ must actually take values in μ_m. Since we are working in an algebraically closed field, this is a group homomorphism from $(\mathbb{Z}/m\mathbb{Z})^2$ to $\mathbb{Z}/m\mathbb{Z}$. There are m^2 such maps. Thus, the cardinalities of $E[m]$ and $\mathrm{Hom}(E[m],\mathbb{G}_m)$ agree. Thus, we will show the map is injective and that it is a group homomorphism. For injectivity, choose P_1 and P_2 in $E[m]$ and suppose $\phi(P_1)=\phi(P_2)$ so that

$$(\tau_X^* - id^*)[m]^*((P_1)-(P_2)) = 0.$$

Then $P_1=P_2$. To show ϕ is a group homomorphism, calculate

$$\left(\frac{g_{P_1+P_2}}{g_{P_1}g_{P_2}}\right) = [m]^*((P_1+P_2)-(P_1)-(P_2)+(\mathcal{O}))$$

which is the divisor of a function that vanishes on m-torsion points, and so $\phi(P_1+P_2)\phi(P_1)^{-1}\phi(P_2)^{-1}$ is a trivial map on $E[m]$. In these last arguments we have taken $S=\mathcal{O}$ for simplicity; we leave it to the reader to convince himself the general case follows as easily.

16.3 The Tate-Lichtenbaum pairing for Jacobians

Another pairing intimately related to the Weil pairing is the Tate-Lichtenbaum pairing. This pairing was first defined by Tate [69] for abelian varieties over p-adic number fields in 1958. In 1959, Lichtenbaum defined a pairing on Jacobian varieties and showed that it coincided with the pairing of Tate [40]. Descriptions can be found in Silverman [63, VIII.2, X.1] and Duquesne-Frey [16].

Let K be a field and G be the Galois group $\mathrm{Gal}(\bar{K}/K)$, and let $m\geq 2$ be an integer. Let $\mathcal{J}$ be the Jacobian of a curve C. In analogy to the Kummer sequence for fields, consider the short exact sequence of G-modules

$$0 \longrightarrow \mathcal{J}(\bar{K})[m] \longrightarrow \mathcal{J}(\bar{K}) \xrightarrow{[m]} \mathcal{J}(\bar{K}) \longrightarrow 0.$$

Taking Galois cohomology, we have a long exact sequence,

$$1 \longrightarrow \mathcal{J}(K)[m] \longrightarrow \mathcal{J}(K) \xrightarrow{m} \mathcal{J}(K) \longrightarrow H^1(G, \mathcal{J}(\bar{K})[m]) \longrightarrow H^1(G, \mathcal{J}(\bar{K})) \xrightarrow{m} H^1(G, \mathcal{J}(\bar{K}))$$

from which we can extract a short exact sequence

$$0 \longrightarrow \mathcal{J}(K)/m\mathcal{J}(K) \xrightarrow{\delta} H^1(G, \mathcal{J}(\bar{K})[m]) \xrightarrow{\alpha} H^1(G, \mathcal{J}(\bar{K}))[m] \longrightarrow 0. \tag{16.4}$$

This is the *Kummer sequence* for $\mathcal{J}/K$.

The connecting homomorphism δ is given by Galois cohomology as follows. Suppose $P \in \mathcal{J}(K)$. Choose $Q \in \mathcal{J}(\bar{K})$ such that $[m]Q = P$. Then $\delta(P)$ is a 1-cocycle $c : G \to \mathcal{J}[m]$ given by

$$c_\sigma = Q^\sigma - Q.$$

A different choice of Q (say Q') will alter this cocycle by a coboundary since $Q - Q' \in \mathcal{J}[m]$. The kernel of this map is $m\mathcal{J}(K)$ since in this case c_σ is a coboundary.

This gives a pairing

$$K : \mathcal{J}[m] \times G \to \mathcal{J}[m], \qquad K(P, \sigma) = c_\sigma$$

called the *Kummer pairing*.

The map α in (16.4) arises from the inclusion $\mathcal{J}(\bar{K})[m] \mapsto \mathcal{J}(\bar{K})$. (The image of this map is a cocycle that, under multiplication by m, becomes a coboundary since it takes values in $\mathcal{J}(\bar{K})[m]$.)

Now, recall that $\mathcal{J}(\bar{K})[m]$ is self-dual under the Weil pairing e_m (by Section 16.2). Given two cocycles $c_1, c_2 \in H^1(G, \mathcal{J}(\bar{K})[m])$, we obtain a cocycle $c \in H^2(G, \bar{K}^*)[m]$ by

$$c(\sigma_1, \sigma_2) = e_m(c_1(\sigma_1), c_2(\sigma_2)).$$

This is in fact a cup product, so we denote it by $\cup$. One must check that a different choice of 1-cocycle representatives gives another 2-cocycle that differs by a 2-coboundary: this depends on the bilinearity of the Weil pairing.

We can define the Tate-Lichtenbaum pairing (Tate's version) by

$$\tau_m : H^1(G, \mathcal{J}(\bar{K}))[m] \times \mathcal{J}(K)/m\mathcal{J}(K) \to H^2(G, \bar{K}^*)[m],$$

$$\tau_m(\gamma, P) = \delta(P) \cup \alpha^{-1}(\gamma).$$

One must check that this is independent of the preimage under α^{-1}, which again depends on the bilinearity of the Weil pairing.

Now for Lichtenbaum's side. We still have $\mathcal{J}$ a Jacobian of a curve C defined over a field K.

Consider the short exact sequence of G-modules:

$$1 \longrightarrow \mathrm{Ppl}_{C(\bar{K})} \longrightarrow \mathrm{Div}^0_{C(\bar{K})} \longrightarrow \mathrm{Pic}^0_{C(\bar{K})} \longrightarrow 0. \tag{16.5}$$

From Galois cohomology there is a connecting homomorphism

$$\delta : H^1(G, \mathrm{Pic}^0_{C(\bar{K})}) \to H^2(G, \mathrm{Ppl}_{C(\bar{K})}).$$

This takes a map $\gamma : G \to \mathrm{Pic}^0_{C(\bar{K})}$ to a factor set $\delta(\gamma) : G \times G \to \mathrm{Ppl}_{C(\bar{K})}$ by precomposing (in both variables) the factor set for the extension (16.5) with γ.

This defines a pairing

$$\tau_m : H^1(G, \mathrm{Pic}^0_{C(\bar{K})}) \times \mathrm{Pic}^0_{C(\bar{K})} \to H^2(G, \bar{K}^*)$$

as follows. Let $\gamma \in H^1(G, \mathrm{Pic}^0_{C(\bar{K})})$. Let D be a divisor in $\mathrm{Pic}^0_{C(\bar{K})}$. Then $\delta(\gamma)$ at any $(\sigma_1, \sigma_2) \in G \times G$ is a principal divisor and so it can be evaluated at D (i.e., take any rational function with divisor $\delta(\gamma)$ and evaluate this at D). Of course D must be chosen so its support is disjoint from the support of $\delta(\gamma)$. We set

$$\tau_m(\gamma, D) = \delta(\gamma)(D)$$

as functions on $G \times G$.

Now, $\mathrm{Pic}^0_{C(\bar{K})} = \mathcal{J}_C(\bar{K})$. So far, in Lichtenbaum's pairing, there has been no mention of the integer m. Necessarily, Lichtenbaum's pairing induces a pairing

$$\tau_m : H^1(G, \mathcal{J}_C(\bar{K}))[m] \times \mathcal{J}_C(\bar{K})/m\mathcal{J}_C(\bar{K}) \to H^2(G, \bar{K}^*)[m]. \tag{16.6}$$

To see this, note that if $m\gamma$ is trivial, then $\delta(\gamma)^m = \delta(m\gamma)$ is trivial. Therefore, if $D = mD'$, then $\delta(\gamma)(D) = \delta(\gamma)(mD') = \delta(\gamma)(D')^m$ is trivial.

Lichtenbaum shows that this pairing agrees with Tate's version [40, pp. 126-127].

This is what is often referred to as the Tate-Lichtenbaum pairing. However, we will be interested in a slight modification (yet again) of this pairing, and when we refer to the Tate-Lichtenbaum pairing, we will mean this definition introduced in the next section for elliptic curves.

16.4 The Tate-Lichtenbaum pairing for elliptic curves

In this section, we define the Tate-Lichtenbaum pairing on elliptic curves via yet another definition: this one is an explicit definition in terms of divisors. We show how it coincides with Lichtenbaum's pairing in the case of elliptic curves, then use the elementary definition to show the basic properties. Parts of this exposition follow [24].

Definition 16.4.1. Let $m > 1$ be an integer. Let E be an elliptic curve defined over a field K containing the m-th roots of unity μ_m. Suppose that $P \in E(K)[m]$. Choose divisors D_P and D_Q of disjoint support such that

$$D_P \sim (P) - (\mathcal{O}), \qquad D_Q \sim (Q) - (\mathcal{O}).$$

Then $mD_P \sim 0$, hence there is a function f_P such that

$$(f_P) = mD_P.$$

The Tate-Lichtenbaum pairing

$$\tau_m : E(K)[m] \times E(K)/mE(K) \to K^*/(K^*)^m$$

is defined by

$$\tau_m(P,Q) = f_P(D_Q).$$

Comparing with Definition 16.1.1 will reveal the source of the proverb that the Tate-Lichtenbaum pairing is "half" of the Weil pairing.

Before we prove a host of properties, let us see how this pairing relates to the pairing given in the last section. Fix a $\sigma \in G = \mathrm{Gal}(\bar{K}/K)$. We will simply 'evaluate' the pairing (16.6) at σ and $(m-1)\sigma$. The group $H^1(G, E(\bar{K}))[m]$ evaluated at this point becomes the group $E(\bar{K})[m]$ (write $\gamma(\sigma) = P_\sigma$). Furthermore $\delta(\gamma)(\sigma,(m-1)\sigma)$ is a rational function with divisor $m(P_\sigma) - m(\mathcal{O})$ (write f_P), and the pairing becomes

$$\tau_m(P,Q) = f_P(D_Q)$$

as in the definition above. It takes values in $\bar{K}^*$, but considered up to $(\bar{K}^*)^m$. There are, of course, some things to check. For example, σ must be chosen so that all $P \in E(K)[m]$ satisfy $\gamma(\sigma) = P$ for some γ.

Proposition 16.4.1. *Denote an algebraic closure of K by $\bar{K}$. Definition 16.4.1 is well-defined, and has the following properties:*

1. *Bilinearity: for $P, P' \in E(K)[m]$ and $Q, Q' \in E(K)$*

$$\begin{aligned}\tau_m(P+P',Q) &= \tau_m(P,Q)\tau_m(P',Q),\\ \tau_m(P,Q+Q') &= \tau_m(P,Q)\tau_m(P,Q').\end{aligned}$$

2. *Non-degeneracy: for a finite field K, and nonzero $P \in E(K)[m]$, there exists $Q \in E(K)$ such that*

$$\tau_m(P,Q) \neq 1.$$

   Furthermore, for a finite field K and $Q \in E(K)\backslash mE(K)$, there exists a $P \in E(K)[m]$ such that

$$\tau_m(P,Q) \neq 1.$$

3. *Galois invariance: for $P, Q \in E[m]$, and $\sigma \in \mathrm{Gal}(\bar{K}/K)$,*

$$\tau_m(P,Q)^\sigma = \tau_m(P^\sigma, Q^\sigma).$$

Proof. **Well-definition:** First, we show the definition does not depend on the choice of D_Q. For, suppose that $D \sim D'$ for two divisors D and D' of degree zero defined over K, and suppose furthermore that the supports of D and D' are disjoint from the support of f_P. Then $D' = D + (g)$ for some g with support disjoint from that of f_P. Therefore

$$\frac{f_P(D')}{f_P(D)} = f((g)) = g((f)) = g(mD_P) = g(D_P)^m$$

and so $\frac{f_P(D')}{f_P(D)}$ is an element of $(K^*)^m$. This ensures that the definition does not depend on the choice of D_Q.

Second, we show that the definition does not depend on the choice of representative of Q in $E(K)/mE(K)$. Suppose that there are two divisors D and D' of degree zero defined over K, and suppose furthermore that the supports of D and D' are disjoint from the support of f_P. Suppose that $D \sim (Q) - (\mathcal{O})$ and $D' \sim (Q + [m]R) - (\mathcal{O})$. Then $D' \sim D + m(R) - m(\mathcal{O})$. By the previous part, $f_P(D') = f_P(D + m(R) - m(\mathcal{O}))$ up to m-th powers, and

$$f_P(D + m(R) - m(\mathcal{O})) = f_P(D) f_P((R) - (\mathcal{O}))^m,$$

so $f_P(D')$ and $f_P(D)$ are equal up to m-th powers.

Finally, we must show that the definition does not depend on the choice of D_P. For, let D and D' be two divisors defined over K, of degree zero, and suppose that $D \sim D'$ and $mD \sim mD' \sim 0$. Let f and f' be functions whose divisors are mD and mD' respectively. Then let g be the function f'/f, so that $f' = fg^m$. Then for any divisor D with support disjoint from f' and f, we have

$$f'(D) = f(D)g(D)^m.$$

Thus the values $f'(D)$ and $f(D)$ are equal up to m-th powers.

These three arguments show that the Tate-Lichtenbaum pairing is defined up to m-th powers in K^*.

Bilinearity: Suppose that $P, P' \in E(K)[m]$ and $Q, Q' \in E(K)$. Then $D_{P+P'} \sim D_P + D'_P$ so $mD_{P+P'} \sim mD_P + mD'_P$. In what follows recall that '=' denotes equality up to m-th powers. We use the results of the last part of the proof ('well-definition'). We have

$$\tau_m(P + P', Q) = f_{P+P'}(D_Q) = f_P(D_Q) f_{P'}(D_Q) = \tau_m(P, Q)\tau_m(P', Q).$$

For the other factor, $D_{Q+Q'} \sim D_Q + D'_Q$ so

$$\tau_m(P, Q + Q') = f_P(D_{Q+Q'}) = f_P(D_Q + D'_Q) = f_P(D_Q) f_P(D'_Q) = \tau_m(P, Q)\tau_m(P', Q).$$

Non-degeneracy: This proof follows [30, Thm 4] in the case of a finite field. For local fields, Lichtenbaum shows this in [40].

Galois invariance: Let $\sigma \in \mathrm{Gal}(\bar{K}/K)$. We have

$$\tau_m(P^\sigma, Q^\sigma) = f_{P^\sigma}(D_{Q^\sigma}) = f_P^\sigma(D_Q^\sigma) = (f_P(D_Q))^\sigma = \tau_m(P, Q)^\sigma.$$

□

Chapter 17

Pairings via elliptic nets

In this chapter we will apply the results of Chapter 15 to obtain formulæ for the Tate-Lichtenbaum and Weil pairings on an elliptic curve in terms of ellipic nets. The first section contains an abstract construction of two pairings on biextensions. We then show that these pairings, for the Poincaré biextension for an elliptic curve, are exactly the Tate-Lichtenbaum and Weil pairings. In the next section we use the elliptic net description of the Poincaré biextension (Theorem 15.1.1), to give elliptic net formulae for the pairings. Finally we give a few examples.

17.1 Pairings from biextensions

In this section we give an abstract construction for two pairings arising from any biextension. The abstract construction of the second pairing (which we call q_m below) and its relation to the Weil pairing has been described by Gorchinskiĭ in the case of a symmetric biextension [27, §3.2]. However, the abstract construction of the first pairing (which we call p_m below) appears to be the more natural of the two, and is not, to the author's knowledge, described anywhere. It becomes the Tate-Lichtenbaum pairing for the Poincaré biextension.

Let X be a biextension of $B \times C$ by A, so that

$$X \xrightarrow{\pi} B \times C$$

is an A-torsor. First we address some general observations that will smooth the discussion. Most importantly, each slice X_b (resp. X_c) is an extension of C (resp. B) by A. Hence a fibre of X over $(b,0)$ (resp. $(0,c)$) is naturally identified with A and we have a natural choice of identity element (the element which acts on X_b (resp. X_c) trivially under $+_2$ (resp. $+_1$)), which we may choose to call 0_b (resp. 0_c), and which is also the identity of the slice X_b (resp. X_c).

This is *not* true of a general fibre over a point (b,c): here we can only identify the difference of two points of the fibre with an element of A via the action taking one to the other. This identification does satisfy some consistency, however, as follows. Consider any x, y in the same fibre over (b,c) of

X_c, and let z be another point of X_c; then $x +_2 z$ and $y +_2 z$ are in the same fibre and furthermore

$$x^a = y \implies (x +_2 z)^a = y +_2 z. \tag{17.1}$$

This follows from the fact that X_c is an extension of C by A. Finally, the group laws $+_1$ and $+_2$ necessarily restrict to the same group law on the fibre $(0,0)$. If we have an element of a fibre $(b,0)$ (resp. $(0,c)$), then $+_2$ (resp. $+_1$) agrees with the group law of A under the identification of the fibre with A. Of course, if we have two elements over the general fibre (b,c), then we may add them with either group law $+_1$ or $+_2$ and the results will differ (in fact they will lie on different fibres).

We now wish to define a pairing. For any positive integer m, let $B[m]$ denote the m-torsion points of the abelian group B. Let m_2 denote multiplication by m under the group law $+_2$ defined on the slice X_c for any $c \in C$. Define m_1 similarly.

Theorem 17.1.1. *Let m be a positive integer. Let A, B and C be abelian groups. Let X be a biextension of $B \times C$ by A given by $\pi : X \to B \times C$. Let $s \in B$, and let σ be a section to π. For each $b \in B[m]$ and $c \in C$, define $p_m(b,c)$ to be the element $a \in A$ whose action takes $m_2(\sigma(b,c) +_2 \sigma(s,c)$ to $\sigma(s,c)$ in the fibre over (s,c) in X.*

Then p_m defines a bilinear pairing

$$p_m : B[m] \times C/mC \to A/mA$$

which is independent of the choice of s and σ, and satisfies $p_m(0,c) = p_m(b,0) = 0$. There exists a pairing

$$p'_m : B/mB \times C[m] \to A/mA$$

defined symmetrically.

Furthermore, there exists a bilinear pairing

$$q_m : B[m] \times C[m] \to A[m]$$

given by $p_m(b,c) - p'_m(b,c)$ (where these are considered as elements of A as described in the first paragraph, not as cosets modulo mA). This pairing is also independent of the choice of s and σ, is skew-symmetric, and satisfies $q_m(0,c) = q_m(b,0) = 0$.

Proof. On X_c, the element

$$m_2\sigma(b,c) +_2 \sigma(s,c)$$

lies above (s,c), as does $\sigma(s,c)$. There is some $a \in A$ such that $(m_2\sigma(b,c) +_2 \sigma(s,c))^a = \sigma(s,c)$. Let this a be the pairing $p_m(b,c)$. For notational ease, we may identify any difference of points in the same fibre with an element of A, and recall that any translation of these is identified with the same element of A (see (17.1) above). Then we may legitimately write

$$p_m(b,c) = (m_2\sigma(b,c) +_2 \sigma(s,c)) -_2 \sigma(s,c) = m_2\sigma(b,c),$$

since the last expression is an element of the fibre over $(0,c)$. This demonstrates that the definition is independent of s. Thus, from now on we set $s = 0$.

This pairing is bilinear in the first argument (as an element in A/mA) because

$$p_m(b+b',c) - p_m(b,c) - p_m(b',c) = m_2\left(\sigma(b+b',c) -_2 \sigma(b,c) -_2 \sigma(b',c)\right),$$

where $\sigma(b+b',c) -_2 \sigma(b,c) -_2 \sigma(b',c)$ lies over $(0,c)$, hence $p_m(b+b',c) - p_m(b,c) - p_m(b',c) \in mA$.

It is bilinear in the second argument in a similar manner:

$$p_m(b,c+c') - p_m(b,c) - p_m(b,c') = m_2\left(\sigma(b,c+c') -_1 \sigma(b,c) -_1 \sigma(b,c')\right),$$

where this time the expression becomes m times an element which lies over $(b,0)$. Note that the compatibility property (14.1) allows us to distribute m_2 over the $+_1$.

It is not dependent on the choice of section because if we choose a different section, and define pairing $\hat{p}_m(b,c)$ associated to $\hat{\sigma}$, then we have

$$p_m(b,c) - \hat{p}_m(b,c) = m(\sigma(b,c) - \hat{\sigma}(b,c)),$$

where $\sigma(b,c) - \sigma'(b,c)$ lies over $(0,0)$, hence $p_m(b,c) - \hat{p}_m(b,c) \in mA$.

By bilinearity, $p_m(0,c) = 0 = p_m(b,0)$ for all $b \in B$ and $c \in C$.

It is defined on C/mC because if $c = mc'$ then $p_m(b,c) = mp_m(b,c') = p_m(mb,c) = p_m(0,c) = 0$.

Now, we turn to the second pairing. In general, the properties of the second do not follow immediately from the first because the first took values in A/mA. However, the proof of independence from s is exactly the same. To demonstrate independence from the choice of σ, let $\hat{q}_m$ the pairing defined using another section $\hat{\sigma}$. Then,

$$\begin{aligned} q_m(b,c) - \hat{q}_m(b,c) &= p_m(b,c) - p'_m(b,c) - \hat{p}_m(b,c) + \hat{p}'_m(b,c) \\ &= m_2(\sigma(b,c) - \hat{\sigma}(b,c)) - m_1(\sigma(b,c) - \hat{\sigma}(b,c)) = 0, \end{aligned}$$

where the final equality comes from the consideration that $\sigma(b,c) - \hat{\sigma}(b,c)$ is in the fibre over $(0,0)$ and here the group laws m_1 and m_2 restrict to the same group law on A. The compatibility condition was used subtly here to leave off subscripts on the minus signs in the third expression. That is, suppose x and y are in the fibre over (b,c) and such that m_2x, m_2y lie over $(0,c)$ and m_1x, m_1y lie over $(b,0)$ and are therefore identified with elements of A. Then $m_2(x +_1 y) = m_2x +_1 m_2y = m_2x +_2 m_2y = m_2(x +_2 y)$.

We have

$$p_m(b+b',c) - p_m(b,c) - p_m(b',c) = m_2(\sigma(b+b',c) -_2 \sigma(b,c) -_2 \sigma(b',c))$$

where the operations are performed via group law $+_2$, and

$$p'_m(b+b',c) - p'_m(b,c) - p'_m(b',c) = m_1(\sigma(b+b',c) -_2 \sigma(b,c) -_2 \sigma(b',c)).$$

The expression in brackets is an element of A (identified with the fibre over $(0,c)$). The left hand sides of these two equations lie in the fibres over $(0,c)$ and $(0,0)$ respectively. Hence they are also identified with A. Thus maps $m_2 : X_{(0,c)} \to X_{(0,c)}$ and $m_1 : X_{(0,c)} \to X_{(0,0)}$ induce maps $m_2, m_1 : A \to A$. Each of

these maps must agree with the map $m : A \to A$ and hence are equal. (To see that the second map must agree, note that the slice X_0 for $0 \in B$ is a *split* extension of C by A.) This gives bilinearity of q_m in the first argument. Symmetry gives bilinearity in the second argument. The last properties follow from bilinearity, and finally, the values are in $A[m]$ since $mq_m(b,c) = q_m(b,mc) = q_m(b,0) = 0$. □

For the purposes of computation, let us consider the factor system associated to σ, called (ϕ, ψ). Then, identifying X with $A \times B \times C$ according to this section, and assuming that $\sigma(b,c) = (1,b,c)$, we obtain

$$m_2\sigma(b,c) = (p_m(b,c), 0, c) \in A \times B \times C.$$

Loosely speaking, p_m measures the *monodromy* of the cycle $mb = 0$ in the group law on X_c. Thus, to compute $p_m(b,c)$, we can perform factor set computations, i.e.

$$p_m(b,c) = \psi(b,b,c)\psi(2b,b,c)\psi(3b,b,c)\cdots\psi((m-1)b,b,c).$$

Similarly, to compute the pairing q_m, we can perform two such computations, so that

$$p_m(b,c) = \frac{\psi(b,b,c)\psi(2b,b,c)\psi(3b,b,c)\cdots\psi((m-1)b,b,c)}{\psi(b,c,c)\psi(b,c,2c)\psi(b,c,3c)\cdots\psi(b,c,[m-1]c)}.$$

Theorem 17.1.2. *The pairings p_m and q_m in Theorem 17.1.1, in the case of the Poincaré biextension for elliptic curves, are the Tate-Lichtenbaum and Weil pairings respectively.*

Proof. Let E be an elliptic curve and let X be the Poincaré biextension over $E \times E$. The computation of $p_m(P,Q)$ happens on the slice X_Q, which is equivalent to an extension $\mathfrak{J}_{\mathfrak{m}}$ for some modulus $\mathfrak{m} = (Q+S)+(S)$. Let D_Q be the divisor $(Q+S)-(S)$. Using the corresponding factor set, and choosing S to avoid any possible intersection of the supports of all divisors concerned in the following calculation, we obtain that

$$p_m(P,Q) = f_{P,P}(D_Q)f_{[2]P,P}(D_Q)\cdots f_{[m-1]P,P}(D_Q) = f_P(D_Q) = \tau_m(P,Q). \tag{17.2}$$

The second equality above follows from the equality of divisors

$$([2]P)+(P)-([3]P)-(\mathcal{O})+([3]P)+(P)-([4]P)-(\mathcal{O})+\ldots+([m-1]P)+(P)-2(\mathcal{O}) = m(P)-m(\mathcal{O}).$$

To show that q_m is the Weil pairing, compare Definition 16.1.1. □

17.2 Tate-Lichtenbaum and Weil pairings from elliptic nets

Theorem 17.2.1. *Let E be an elliptic curve over a field K containing the m-th roots of unity. Let $P \in E[m]$ and $Q \in E$. Let S be any point on E. The quantity*

$$\frac{\mathcal{W}(mp+q+s)\mathcal{W}(s)}{\mathcal{W}(mp+s)\mathcal{W}(q+s)} \tag{17.3}$$

is well-defined as an element of $K^*/(K^*)^m$, and is equal to the Tate-Lichtenbaum pairing $\tau_m(P,Q)$.

Proof. First we verify that the quantity (17.3) is well-defined, when considered as an element of $K^*/(K^*)^m$. To do so, verify that

$$(mp+q+s)^2+s^2-(mp+s)^2-(q+s)^2 \equiv 0 \mod m$$

as a polynomial relation in variables p, q, s.

We use Theorem 15.1.1. Consider the Poincaré biextension of $E \times E$ by $\mathbb{G}_m$. Let $\tilde{P}$ and $\tilde{S}$ be lifts of (P,Q) and (S,Q) to the Poincaré biextension X. Then $[m]_2(\tilde{P})$ is in the fibre over $(\mathcal{O},Q)$. The point $[m]_2(\tilde{P})+_2(\tilde{S})$ is in the fibre over (S,Q) and can be compared to $\tilde{S}$ to yield an element of $\mathbb{G}_m$. This element is the Tate-Lichtenbaum pairing, by Theorems 17.1.1 and 17.1.2. Using the elliptic net factor system for the biextension, we can calculate this element to be

$$\begin{aligned}
&\Lambda(P,P,Q)\Lambda([2]P,P,Q)\Lambda([3]P,P,Q)\cdots\Lambda([m-1]P,P,Q)\Lambda(\mathcal{O},S,Q) \qquad (17.4)\\
&= \left(\frac{\mathcal{W}(2p+q)\mathcal{W}(p)\mathcal{W}(p)\mathcal{W}(q)}{\mathcal{W}(2p)\mathcal{W}(p+q)\mathcal{W}(p+q)}\right)\left(\frac{\mathcal{W}(3p+q)\mathcal{W}(2p)\mathcal{W}(p)\mathcal{W}(q)}{\mathcal{W}(3p)\mathcal{W}(2p+q)\mathcal{W}(p+q)}\right)\\
&\times\left(\frac{\mathcal{W}(4p+q)\mathcal{W}(3p)\mathcal{W}(p)\mathcal{W}(q)}{\mathcal{W}(4p)\mathcal{W}(3p+q)\mathcal{W}(p+q)}\right)\cdots\left(\frac{\mathcal{W}(mp+q)\mathcal{W}((m-1)p)\mathcal{W}(p)\mathcal{W}(q)}{\mathcal{W}(mp)\mathcal{W}((m-1)p+q)\mathcal{W}(p+q)}\right)\\
&\times\left(\frac{\mathcal{W}(mp+s+q)\mathcal{W}(mp)\mathcal{W}(s)\mathcal{W}(q)}{\mathcal{W}(mp+s)\mathcal{W}(mp+q)\mathcal{W}(q+s)}\right)\\
&= \left(\frac{\mathcal{W}(p)\mathcal{W}(q)}{\mathcal{W}(p+q)}\right)^m\left(\frac{\mathcal{W}(mp+s+q)\mathcal{W}(s)}{\mathcal{W}(mp+s)\mathcal{W}(q+s)}\right)
\end{aligned}$$

□

This could also be proven directly, but this would require a number of cases. After having shown the expression for the pairing is well-defined, we could choose a convenient elliptic net (such as that associated to the basis P,Q,S), and then verify directly that the function we obtain (as a function of P,Q,S) is exactly that of Definition 16.4.1. This is essentially the proof given in the author's paper [65], where the terms 'generalised Jacobian' and 'biextension' are not mentioned.

Theorem 17.2.2. *Let E be an elliptic curve over a field K containing the field of definition of $E[m]$ and of characteristic coprime to m. Let $P,Q \in E[m]$. The quantity*

$$\frac{\mathcal{W}(mp+q+s)\mathcal{W}(p+s)\mathcal{W}(mq+s)}{\mathcal{W}(mp+s)\mathcal{W}(q+s)\mathcal{W}(p+mq+s)} \qquad (17.5)$$

is well-defined, independent of S and is equal to the Weil pairing $e_m(P,Q)$.

Proof. Verify that

$$(mp+q+s)^2+(p+s)^2+(mq+s)^2-(mp+s)^2-(q+s)^2-(p+mq+s)^2=0.$$

The statement follows from Theorems 17.1.1 and 17.1.2, by applying calculation (17.4) twice. □

17.3 Partial periodicity and pairings

We can restate the formulæ for the Tate-Lichtenbaum and Weil pairings in terms of the partial periodicity properties of Chapter 10, specifically Theorem 10.2.3, to give a particularly concrete statement.

Theorem 17.3.1. *Let E be an elliptic curve over a field K containing the m-th roots of unity. Let $P \in E[m]$ and $Q \in E$ be non-zero. Let S be any other non-zero point in E such that $P+S$ and $Q+S$ do not vanish. Let $\mathbf{P}$ be a basis generating a group containing P, Q, S and suppose $\mathbf{p}\cdot\mathbf{P} = P$, $\mathbf{q}\cdot\mathbf{P} = Q$, and $\mathbf{s}\cdot\mathbf{P} = S$. The quantity*

$$\frac{g(m\mathbf{p}, \mathbf{q}+\mathbf{s})}{g(m\mathbf{p}, \mathbf{s})} = \frac{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{q}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{s})}{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{q}+\mathbf{s})}$$

is well-defined as an element of $K^*/(K^*)^m$, and is equal to the Tate-Lichtenbaum pairing $\tau_m(P,Q)$.

Theorem 17.3.2. *Let E be an elliptic curve over a field K containing the field of definition of $E[m]$ and of characteristic coprime to m. Let $P, Q \in E[m]$ be non-zero. Let S be any other non-zero point in E such that $P+S$ and $Q+S$ do not vanish. Let $\mathbf{P}$ be a basis generating a group containing P, Q, S and suppose $\mathbf{p}\cdot\mathbf{P} = P$, $\mathbf{q}\cdot\mathbf{P} = Q$, and $\mathbf{s}\cdot\mathbf{P} = S$. The quantity*

$$\frac{g(m\mathbf{p}, \mathbf{q}+\mathbf{s})g(m\mathbf{q}, \mathbf{s})}{g(m\mathbf{p}, \mathbf{s})g(m\mathbf{q}, \mathbf{p}+\mathbf{s})} = \frac{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{q}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{p}+\mathbf{s})W_{E,\mathbf{P}}(m\mathbf{q}+\mathbf{s})}{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{q}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{p}+m\mathbf{q}+\mathbf{s})}$$

is well-defined, independent of S and is equal to the Weil pairing $e_m(P,Q)$.

17.4 Example calculations

We show a few example calculations of the Weil and Tate-Lichtenbaum pairings.

Example 17.4.1. Take the elliptic curve

$$E : y^2 = x^3 + 319x + 183$$

over $\mathbb{F}_{347}$. Take two points

$$P = (1, 117), \qquad Q = (232, 138)$$

on E. These points are both of order $m = 178$.

The Weil pairing is

$$\begin{aligned} e_m(P,Q) &= \frac{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{q}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{p}+\mathbf{s})W_{E,\mathbf{P}}(m\mathbf{q}+\mathbf{s})}{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{q}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{p}+m\mathbf{q}+\mathbf{s})} \\ &= \frac{W_{E,P,Q}(m+2,1)W_{E,P,Q}(3,0)W_{E,P,Q}(2,m)}{W_{E,P,Q}(m+2,0)W_{E,P,Q}(2,1)W_{E,P,Q}(3,m)} \\ &= \frac{22\cdot 206\cdot 319}{119\cdot 82\cdot 333} = 1 \end{aligned}$$

which indicates that there is a relation $P = [n]Q$, and in fact a brief search reveals that $P = [165]Q$.

We can calculate the Tate-Lichtenbaum pairing $\tau_m(P,Q)$ using the elliptic net $W_{E,P,Q}$ with $S = (2,0)$. The formula becomes

$$\tau_m(P,Q) = \frac{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{q}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{s})}{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{q}+\mathbf{s})} = \frac{W_{E,P,Q}(m+2,1)W_{E,P,Q}(2,0)}{W_{E,P,Q}(m+2,0)W_{E,P,Q}(2,1)} = \frac{22\cdot 234}{119\cdot 82} = 73.$$

Suppose we had chosen a different elliptic net for the evaluation of the Tate-Lichtenbaum pairing: say $W_{E,P,P+Q}$ and $S = [2]P - Q$. Then we have

$$\tau_m(P,Q) = \frac{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{q}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{s})}{W_{E,\mathbf{P}}(m\mathbf{p}+\mathbf{s})W_{E,\mathbf{P}}(\mathbf{q}+\mathbf{s})} = \frac{W_{E,P,P+Q}(m+2,0)W_{E,P,P+Q}(3,-1)}{W_{E,P,P+Q}(m+3,-1)W_{E,P,P+Q}(2,0)} = \frac{119\cdot 85}{337\cdot 234} = 293.$$

Since the $\gcd(178,346) = 2$, these results are considered modulo $\mathbb{F}_{347}^*/(\mathbb{F}_{347}^*)^{178}$ which is a group of order two ($(\mathbb{F}_{347}^*)^{178}$ is all the quadratic residues). The ratio $73/293 \equiv 185 \mod 347$ is a quadratic residue, so the calculations agree.

Part IV

Cryptographic applications

Chapter 18

Tate pairing computation

In this section we discuss algorithms for computing the Tate-Lichtenbaum pairing using elliptic nets and Theorem 17.2.1. The first section is a review of Miller's algorithm for computing the Tate-Lichtenbaum pairing, which has several features in common with the elliptic net algorithm. Both are based on computations on the biextension or generalised Jacobian. The elliptic net algorithm relies on an algorithms for computing the terms of an elliptic net, and is based on algorithms of Shipsey for elliptic divisibility sequences [61]. Then we present the elliptic net algorithm for computing the Tate-Lichtenbaum pairing, and perform some rudimentary analysis on its runtime.

The elliptic net algorithm has been implemented by the author for PARI/GP (see [71]) and the code is included in Appendix B.3. It has also been implemented in C++ by Michael Scott and Augusto Jun Devegili for a pairing-friendly curve of degree 2. The implementation by Ben Lynn in the Pairing Based Cryptography Library [43] is applicable to curves of various sizes and embedding degrees and includes a program to compare the Elliptic Net algorithm with Miller's. Preliminary data agree with the complexity analysis above. It has also been implemented (and improved) by Graeme Taylor at the University of Edinburgh, who has implemented it for SAGE (see [66]) and has notes on his improvements available at [70].

18.1 Miller's algorithm

Victor Miller described an algorithm for computing the Tate-Lichtenbaum and Weil pairings in 1986 [46], and it has since been the only algorithm used for this purpose. In 2004, he published an updated version of his earlier account [47]. His algorithm is easy to describe in the context of the biextension description of the Tate-Lichtenbaum and Weil pairings given in Section 17.1. We wish to compute $f_P(D_Q)$ where f_P is a rational function with divisor $m(P) - m(0)$ and $D_Q = (Q+S) - (S)$ for some S. Miller's idea arises from the observation (17.2), reprinted here:

$$\tau_m(P,Q) = f_P(D_Q) = f_{P,P}(D_Q) f_{[2]P,P}(D_Q) \cdots f_{[m-1]P,P}(D_Q).$$

The equation above reflects the calculation of $[m]P$ from repeated additions of P. Miller's insight was that any chain of additions that results in $[m]P$ would do as well. Thus, he suggested the use of a double-and-add method. Miller's basic algorithm is given in Algorithm 18.0.1. It creates a double-and-add chain based on the binary expansion of m, then computes the point $[m]P$ with this chain, at each step computing the appropriate value $f_{A,B}(D_Q)$ and keeping a running product.

Algorithm 1 Miller's algorithm

Require: Points P and Q of an elliptic net, divisor $D_Q = (Q+S)-(S)$ with S such that D_Q is disjoint from $([n]P)$ for all n, and integer $m = (d_k d_{k-1} \dots d_1)_2$ with $d_k = 1$
Ensure: The Tate-Lichtenbaum pairing $\tau_m(P,Q)$

$f \leftarrow 1$
for $i = k-1$ down to 1 **do**
  $f \leftarrow f^2 f_{T,T}(D_Q)$
  $T \leftarrow [2]T$
  if $d_i = 1$ **then**
    $f \leftarrow f f_{T,P}(D_Q)$
    $T \leftarrow T+P$
  end if
end for
return f

18.2 Computing the values of an elliptic net

Rachel Shipsey gives a double-and-add algorithm for computing terms of an elliptic divisibility sequence [61]. In the case of interest to us now, given the initial values of an elliptic divisibility sequence, the algorithm computes the n-th term of a sequence in $\log(n)$ time. Shipsey applied her more general algorithm (which allows beginning elsewhere in the sequence) to give a solution to the elliptic curve discrete logarithm problem in certain cases.

The algorithm described here is an adaptation and generalisation of Shipsey's algorithm to calculate terms $W(m,0)$ and $W(m,1)$ of an elliptic net. We define a *block centred on k* (shown in Fig. 18.1) to consist of a first vector of eight consecutive terms of the sequence $W(i,0)$ centred on terms $W(k,0)$ and $W(k+1,0)$ and a second vector of three consecutive terms $W(i,1)$ centred on the term $W(k,1)$. We define two functions:

1. **Double**(V)**:** Given a block V centred on k, returns the block centred on $2k$.
2. **DoubleAdd**(V)**:** Given a block V centred on k, returns the block centred on $2k+1$.

We assume the elliptic net satisfies $W(1,0) = W(0,1) = 1$. The first vectors of Double(V) and

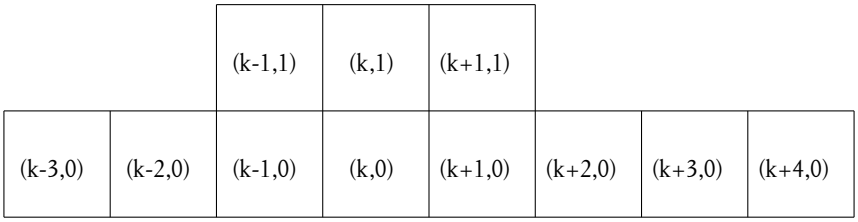


Figure 18.1: A block centred on k

DoubleAdd(V) are calculated according to the following special cases of (3.1):

$$W(2i-1,0) = W(i+1,0)W(i-1,0)^3 - W(i-2,0)W(i,0)^3 \ , \tag{18.1}$$

$$W(2i,0) = (W(i,0)W(i+2,0)W(i-1,0)^2 - W(i,0)W(i-2,0)W(i+1,0)^2)/W(2,0) \ . \tag{18.2}$$

The formulæ needed for the computations of the second vectors are instances of (3.1):[1]

$$W(2k-1,1) = (W(k+1,1)W(k-1,1)W(k-1,0)^2 - W(k,0)W(k-2,0)W(k,1)^2)/W(1,1) \ , \tag{18.3}$$

$$W(2k,1) = W(k-1,1)W(k+1,1)W(k,0)^2 - W(k-1,0)W(k+1,0)W(k,1)^2 \ , \tag{18.4}$$

$$W(2k+1,1) = (W(k-1,1)W(k+1,1)W(k+1,0)^2 - W(k,0)W(k+2,0)W(k,1)^2)/W(-1,1) \ , \tag{18.5}$$

$$W(2k+2,1) = (W(k+1,0)W(k+3,0)W(k,1)^2 - W(k-1,1)W(k+1,1)W(k+2,0)^2)/W(2,-1) \ . \tag{18.6}$$

Equations (18.1) and (18.2), applied for $i = k-1,\ldots,k+3$, allow calculation of the first vectors of Double(V) and DoubleAdd(V) in terms of $W(2,0)$ and the terms of V. Equations (18.3)–(18.6) allow calculation of the second vectors in terms of $W(1,1)$, $W(-1,1)$, $W(2,-1)$ and the terms of V.

The algorithm to calculate $W(m,1)$ and $W(m,0)$ for any positive integer m is shown in Algorithm 18.0.2. The formula for the last term of the first vector of V in line 1 is from (1.2). Note that elliptic nets satisfy $W(-n,-m) = -W(n,m)$ by Proposition 3.1.1. In Section 18.4 we will consider possible optimisations.

18.3 Computation of the Tate-Lichtenbaum pairing

To compute the Tate-Lichtenbaum pairing using Theorem 17.2.1, we must choose a particular elliptic net in which to do calculations. The following corollary to Theorem 17.2.1 represents one convenient choice.

[1] The values p,q,r,s substituted into (3.1) to obtain equations (18.3) - (18.6) are $[p,q,r,s] = [(k,0),(k-1,0),(1,0),(0,1)]$, $[(k+1,0),(k,0),(1,0),(-1,1)]$, $[(k+1,0),(k,0),(-1,0),(0,1)]$, and $[(k+2,0),(k,1),(1,0),(0,0)]$ respectively.

Algorithm 2 Elliptic Net Algorithm

Require: Initial terms $a = W(2,0)$, $b = W(3,0)$, $c = W(4,0)$, $d = W(2,1)$, $e = W(-1,1)$, $f = W(2,-1)$, $g = W(1,1)$ of an elliptic net satisfying $W(1,0) = W(0,1) = 1$ and integer $m = (d_k d_{k-1} \ldots d_1)_2$ with $d_k = 1$
Ensure: Elliptic net elements $W(m,0)$ and $W(m,1)$
1: $V \leftarrow [[-a,-1,0,1,a,b,c,a^3c-b^3];[1,g,d]]$
2: **for** $i = k-1$ down to 1 **do**
3: **if** $d_i = 0$ **then**
4: $V \leftarrow \text{Double}(V)$
5: **else**
6: $V \leftarrow \text{DoubleAdd}(V)$
7: **end if**
8: **end for**
9: **return** $V[0,3]$ and $V[1,1]$ {terms $W(m,0)$ and $W(m,1)$ respectively}

Corollary 18.3.1. *Let E be an elliptic curve defined over a finite field K, m a positive integer, $P \in E(K)[m]$ and $Q \in E(K)$. If W_P is the elliptic net associated to E,P, then we have*

$$\tau_m(P,P) = \frac{W_P(m+2)W_P(1)}{W_P(m+1)W_P(2)} \ . \tag{18.7}$$

Further, if $W_{P,Q}$ is the elliptic net associated to E,P,Q, then we have

$$\tau_m(P,Q) = \frac{W_{P,Q}(m+1,1)W_{P,Q}(1,0)}{W_{P,Q}(m+1,0)W_{P,Q}(1,1)} \ . \tag{18.8}$$

Proof. For the first formula, taking $q = p$ and $s = 2p$, we obtain

$$\tau_m(P,P) = \frac{W((m+2)p)W(p)}{W((m+1)p)W(2p)} \ .$$

For the second, take $s = p$, obtaining

$$\tau_m(P,Q) = \frac{W((m+1)p+q)W(p)}{W((m+1)p)W(p+q)} \ .$$

□

We remind the reader of the following definition:

Definition 18.3.1 (Reminder of Definition 9.1.2)**.** Let W_1 and W_2 be elliptic nets. Suppose $\alpha,\beta \in K^*$, and $f : A \to \mathbb{Z}$ is a quadratic form. If

$$W_1(\mathbf{v}) = \alpha\beta^{f(\mathbf{v})}W_2(\mathbf{v})$$

for all $\mathbf{v}$, then we say W_1 *is equivalent to* W_2 and write $W_1 \sim W_2$.

We can alter the elliptic net we are using for the Tate-Lichtenbaum pairing computation by an equivalence without altering the outcome.

Consider an elliptic curve E over a finite field $\mathbb{F}_q$ of characteristic not 2 or 3, in Weierstrass form

$$y^2 = x^3 + Ax + B$$

and points $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ on $E(\mathbb{F}_q)$ with $Q \neq \pm P$. We must calculate the values a, b, c, d, e, f, and g required as input for the Elliptic Net Algorithm. These are terms of the elliptic net associated to E, P, Q. The necessary formulæ are given by the functions $\Psi_{m,n}$. See Proposition 6.1.4, which gives

$$W(1,0) = 1 \ , \tag{18.9}$$
$$W(2,0) = 2y_1 \ , \tag{18.10}$$
$$W(3,0) = 3x_1^4 + 6Ax_1^2 + 12Bx_1 - A^2 \ , \tag{18.11}$$
$$W(4,0) = 4y_1(x_1^6 + 5Ax_1^4 + 20Bx_1^3 - 5A^2x_1^2 - 4ABx_1 - 8B^2 - A^3) \ . \tag{18.12}$$

$$W(0,1) = W(1,1) = 1 \ , \tag{18.13}$$
$$W(2,1) = 2x_1 + x_2 - \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 \ , \tag{18.14}$$
$$W(-1,1) = x_1 - x_2 \ , \tag{18.15}$$
$$W(2,-1) = (y_1 + y_2)^2 - (2x_1 + x_2)(x_1 - x_2)^2 \ . \tag{18.16}$$

Suppose that P has order m. Then we use the Elliptic Net Algorithm, with input $m+1$ and a, b, c, d, e, f, g given by (18.10)–(18.16).[2] The output is used to evaluate formula (18.8) of Corollary 18.3.1, giving the Tate-Lichtenbaum pairing.

18.4 Some implementation considerations

For an integer m and finite field $\mathbb{F}_q$, we define the *embedding degree* k to be the least integer such that $m|(q^k - 1)$, thus ensuring the m-th roots of unity are contained in $\mathbb{F}_{q^k}^*$. In cryptographic applications of the Tate-Lichtenbaum pairing, it is usual to use a curve defined over $\mathbb{F}_q$ of embedding degree $k > 1$, and points $P \in E(\mathbb{F}_q)$, $Q \in E(\mathbb{F}_{q^k})$: throughout what follows we make this assumption.

First, note that no inversions are actually needed in equations (18.1)–(18.6), since the inverses of $W(2,0)$, $W(2,1)$, $W(-1,1)$ and $W(2,-1)$ may be precomputed before the double-and-add loop is begun. Therefore these inversions are replaced by multiplications.

Now we consider optimisations in the functions Double and DoubleAdd. The largest savings can be gained by first computing a number of products which appear frequently in the formulæ:

$$W(i,0)^2 \text{ and } W(i-1,0)W(i+1,0) \quad \text{for } i = k-2, \ldots, k+3 \ ,$$
$$W(k,1)^2 \text{ and } W(k-1,1)W(k+1,1) \ .$$

[2] In this case, $g = 1$. However, in Section 18.4 we will replace this elliptic net with an equivalent one for which $W(1,1) \neq 1$. For this reason, it is convenient to state Algorithm 18.0.2 in sufficient generality and include a variable g.

With these 14 computations, each term of the 11 to be calculated requires only two multiplications and an addition (plus multiplications by $W(2,0)^{-1}$, $W(2,-1)^{-1}$, $W(1,1)^{-1}$ and $W(-1,1)^{-1}$). The resulting Double and DoubleAdd algorithms are shown in Algorithm 18.0.3.

Algorithm 3 Double and DoubleAdd

Require: Block V centred at k of an elliptic net satisfying $W(1,0) = W(0,1) = 1$, values $A = W(2,0)^{-1}, E = W(-1,1)^{-1}, F = W(2,-1)^{-1}, G = W(1,1)^{-1}$ and boolean add

Ensure: Block centred at $2k$ if $add == 0$ and centred at $2k+1$ if $add == 1$

$S \leftarrow [0,0,0,0,0,0]$
$P \leftarrow [0,0,0,0,0,0]$
$S_0 \leftarrow V[1,1]^2$
$P_0 \leftarrow V[1,0]V[1,2]$
for $i = 0$ to 5 **do**
  $S[i] \leftarrow V[0,i+1]^2$
  $P[i] \leftarrow V[0,i]V[0,i+2]$
end for
if $add == 0$ **then**
  for $i = 1$ to 4 **do**
    $V[0,2i-2] \leftarrow S[i]P[i+1] - S[i+1]P[i]$
    $V[0,2i-1] \leftarrow (S[i]P[i+2] - S[i+2]P[i])A$
  end for
  $V[1,0] \leftarrow (S_0P[3] - S[3]P_0)G$
  $V[1,1] \leftarrow S[3]P_0 - S_0P[3]$
  $V[1,2] \leftarrow (S[4]P_0 - S_0P[4])E$
else
  for $i = 1$ to 4 **do**
    $V[0,2i-2] \leftarrow (S[i]P[i+2] - S[i+2]P[i])A$
    $V[0,2i-1] \leftarrow S[i+1]P[i+2] - S[i+2]P[i+1]$
  end for
  $V[1,0] \leftarrow S[3]P_0 - S_0P[3]$
  $V[1,1] \leftarrow (S[4]P_0 - S_0P[4])E$
  $V[1,2] \leftarrow (S_0P[5] - S[5]P_0)F$
end if
return V

Finally, we may try to avoid some of the extra multiplications by $W(2,0)^{-1}$, $W(1,1)^{-1}$, $W(2,1)^{-1}$ and $W(2,-1)^{-1}$ entirely. Recall that by Theorem 17.2.1, applying an equivalence to the net will not alter the Tate-Lichtenbaum pairing result. Let $\eta = W(-1,1)$. Apply the equivalence given by $\alpha = 1$, $\beta = \eta$ and $f(n,m) = mn$. Clearly, this preserves the conditions[3] that $W(1,0) = W(0,1) = 1$ (and leaves terms $W(n,0)$ unchanged, so they are still in $\mathbb{F}_q$), but changes $W(-1,1)$ to 1, which saves one multiplication in $\mathbb{F}_{q^k}$ per iteration. If $W(2,0)$ has a cube root ν in $\mathbb{F}_q$, then the equivalence $\alpha = \nu^{-1}$, $\beta = \nu$ and $f(n,m) = m^2 + n^2 + mn$ will change $W(2,0)$ to 1, while preserving $W(1,0) = W(0,1) = W(-1,1) = 1$, saving four $\mathbb{F}_q$ multiplications per iteration. Note that these equivalences may result in $W(1,1) \neq 1$.

Finally, we consider the applicability of some of the usual optimisations of Miller's algorithm. In

[3] These were needed to derive formulæ (18.1)–(18.6).

Table 18.1: Comparison of Operations for Double and DoubleAdd steps

Algorithm	Double	DoubleAdd
Optimised Miller's [37]	$4S+(k+7)M+S_k+M_k$	$7S+(2k+19)M+S_k+2M_k$
Elliptic Net Algorithm	$6S+(6k+26)M+S_k+\frac{3}{2}M_k$	$6S+(6k+26)M+S_k+2M_k$

Miller's algorithm, a final exponentiation is applied, in order to compute a unique value for the Tate-Lichtenbaum pairing; the same exponentiation must be applied here. In the case of Miller's, this exponentiation eliminates multiplicative factors living in the base field $\mathbb{F}_q$. In our case, the $\mathbb{F}_q$ computations do not give rise to strictly multiplicative factors (the algorithm requires much addition and subtraction), and so we cannot use this final exponentiation as a justification for the saving of $\mathbb{F}_q$ computations. Windowing methods (as in [6] and [28]) may lead to improvement. A triple-and-add adaptation (as in [26] and [4]) does not seem promising, by the nature of the recurrence relation. However, efficiency improvements are likely to be found by studying the characteristic 2 and 3 cases.

18.5 Complexity

Since the algorithm involves a fixed number of precomputations, and a double-and-add loop with a fixed number of computations per step, the algorithm is linear time in the size of m, as is Miller's algorithm. Miller's algorithm also consists of a double-and-add loop, and we call the two internal steps Double and DoubleAdd, as for the Elliptic Net Algorithm. In Miller's algorithm the cost of DoubleAdd is almost twice that of Double. By contrast, in the Elliptic Net Algorithm these steps take the same time, so the complexity is independent of Hamming weight. This makes the choice of appropriate curves for cryptographical implementations somewhat easier [18], and may help discourage side channel attacks.

Denote squaring and multiplication in $\mathbb{F}_q$ by S and M. Denote squaring and multiplication in $\mathbb{F}_{q^k}$ by S_k and M_k. Assume that multiplying an element of $\mathbb{F}_q$ by one of $\mathbb{F}_{q^k}$ takes k multiplications in $\mathbb{F}_q$. Recall that E is defined over $\mathbb{F}_q$, $P \in E(\mathbb{F}_q)$, and $Q \in E(\mathbb{F}_{q^k})$. Then any term $W(n,0)$, being a term in the elliptic divisibility sequence associated to E,P, has a value in $\mathbb{F}_q$. Under the optimisations discussed in Section 18.4, each Double or DoubleAdd step requires $6S+(6k+26)M+S_k+2M_k$. Furthermore, under the condition that $2y_P \in \mathbb{F}_q$ is a cube, then precomputing its cube root will save four multiplications in $\mathbb{F}_q$ per step.

The Elliptic Net Algorithm requires no inversions. Miller's algorithm in affine coordinates requires one or two $\mathbb{F}_q$ inversion per step. In situations where inversions are costly (depending on implementation, they may cost anywhere from approximately 4 to 80 multiplications [11]), one may implement Miller's algorithm in homogeneous coordinates.

For the purpose of comparison, we consider an optimised implementation of Miller's algorithm in Jacobian coordinates analysed by Neal Koblitz and Alfred Menezes [37]. In their implementation, they

Table 18.2: $\mathbb{F}_q$ Multiplications per Step

Embedding degree	2	4	6	8	10	12
Optimised Miller's	18-38	31-58	46-82	64-109	84-140	106-174
Elliptic Net	51-52	76-80	104-112	136-147	171-186	207-228

assume $x(Q) \in E(\mathbb{F}_{q^{k/2}})$ (this is possible by using a twist of the curve, see for example [5]). Applying this additional assumption to the elliptic net algorithm, $W(1,1)$ will be an element of $\mathbb{F}_{q^{k/2}}$, reducing one of the multiplications in Double to one half the time. The comparison is summarised in Tables 18.1 and 18.2. In the latter, a squaring is assumed to be comparable to a multiplication (although it is more usually assumed to be 0.8 times as fast), and a multiplication in $\mathbb{F}_{q^k}$ is assumed to take $k^{1.5}$ multiplications in $\mathbb{F}_q$ (see [37]). The number of steps constitutes a range because the Double and DoubleAdd steps may differ in cost.

Chapter 19

The elliptic curve discrete logarithm problem

This chapter contains joint work with Kristin Lauter performed during an internship at Microsoft Research, Redmond, Washington, September 10, 2007 - December 14, 2007.

The purpose of our study in this chapter is to better understand the following problem.

Problem 19.0.1 (Elliptic Curve Discrete Logarithm Problem (ECDLP)). *Let E be an elliptic curve over a finite field K. Suppose one is given points $P, Q \in E(K)$ such that $Q \in \langle P \rangle$. Determine k such that $Q = [k]P$.*

The first section of this chapter discusses perfectly periodic sequences. Then, we turn to the question of hard problems for elliptic divisibility sequences and elliptic nets, and compare these problems with Problem 19.0.1.

19.1 Perfect periodicity

Definition 19.1.1. A periodic elliptic divisibility sequence whose rank of zero-apparition is equal to its period, is called *perfectly periodic*. A periodic elliptic net is called *perfectly periodic* if its lattice of zero-apparition is equal to its lattice of periodicity.

We will often put a tilde over a sequence to remind the reader that it is perfectly periodic (e.g. $\widetilde{W}(k)$).

As an example, let W be a non-degenerate elliptic divisibility sequence. Consider the equivalent sequence $W'(n) = \alpha^{n^2-1} W(n)$ where α satisfies $\alpha^2 = a, \alpha^m = b$ for a, b from Theorem 10.2.2. It

follows that this sequence is a perfectly periodic elliptic divisiblity sequence. Suppose that $\gcd(q-1,m)=1$. In this case the conditions of Theorem 10.2.2 determine such an α uniquely, and it lies in K. Otherwise (if $\gcd(q-1,m)\neq 1$), two such α's will exist, equal up to sign. The two resulting perfectly periodic sequences will be equal (only up to sign at odd-indexed locations).

The moral of the last paragraph is that any elliptic divisibility sequence is equivalent to a perfectly periodic one. We can give an explicit expression for such a perfectly periodic sequence.

Theorem 19.1.1. *Let K be a finite field of q elements, and E an elliptic curve defined over K. Suppose $\#E(K)$ is relatively prime to $q-1$. Define a function*

$$\phi : E \to K$$

by

$$\phi(P) = \left(\frac{W_{E,P}(q-1)}{W_{E,P}(q-1+\mathrm{ord}(P))} \right)^{\frac{1}{\mathrm{ord}(P)^2}}. \tag{19.1}$$

For a point P of prime order not less than 4, the sequence $\phi([n]P)$ is a perfectly periodic elliptic divisibility sequence equivalent to $W_{E,P}(n)$. Specifically,

$$\phi([n]P) = \phi(P)^{n^2-1} W_{E,P}(n). \tag{19.2}$$

More generally, let $\mathbf{P} \in E(K)^n$ be a collection of nonzero points, no two equal or inverses, and all elements of a single cyclic group. The n-array $\phi(\mathbf{v}\cdot\mathbf{P})$ (as $\mathbf{v}$ ranges over $\mathbb{Z}^n$) forms a perfectly periodic elliptic net equivalent to $W_{E,\mathbf{P}}(\mathbf{v})$. Specifically,

$$\phi(\mathbf{v}\cdot\mathbf{P}) = W_{E,\mathbf{P}}(\mathbf{v}) \prod_{i=1}^{n} \phi(P_i)^{v_i^2 - v_i\left(\sum_{j\neq i} v_j\right)} \prod_{1\leq i<j\leq n} \phi(P_i+P_j)^{v_i v_j}. \tag{19.3}$$

Proof. The proof uses Theorem 10.1.1. We will demonstrate the method of proof in the rank one case before proceeding to the general case. Take $T=(l)$, so

$$W_{E,[l]P}(n) W_{E,P}(l)^{n^2} = W_{E,P}(nl).$$

By symmetry,

$$W_{E,[n]P}(l) W_{E,P}(n)^{l^2} = W_{E,P}(nl).$$

Let $m=\mathrm{ord}(P)$. Thus, combining the above and using $l=q-1$ and $q-1+m$ in turn,

$$\begin{aligned} \frac{W_{E,[n]P}(q-1) W_{E,P}(n)^{(q-1)^2}}{W_{E,P}(q-1)^{n^2}} &= W_{E,[q-1]P}(n) = W_{E,[q-1+m]P}(n) \\ &= \frac{W_{E,[n]P}(q-1+m) W_{E,P}(n)^{(q-1+m)^2}}{W_{E,P}(q-1+m)^{n^2}} \end{aligned}$$

Rearranging,

$$\phi([n]P) = \phi(P)^{n^2-1} W_{E,P}(n).$$

Therefore, $\phi([n]P)$ is an elliptic divisibility sequence. By definition, $\phi([n]P)$ has period $\mathrm{ord}(P)$ which is equal to the rank of apparition of $W_{E,P}$ and $\phi([n]P)$. So $\phi([n]P)$ is perfectly periodic.

For the rank n case, let m be the order of the cyclic group containing all the points under consideration. In Theorem 10.1.1, let $t = 1$ and $s = n$ and take $T = (v_1 \quad v_2 \quad v_3 \quad \cdots \quad v_n)$ to obtain

$$W_{E,\mathbf{P}}(l\mathbf{v}) = W_{E,\mathbf{v}\cdot\mathbf{P}}(l)W_{E,\mathbf{P}}(\mathbf{v})^{l^2}.$$

Now take $t = s = n$ in Theorem 10.1.1 , and $T = l\mathrm{Id}_n$ to obtain

$$W_{E,\mathbf{P}}(l\mathbf{v}) = W_{E,l\mathbf{P}}(\mathbf{v})\prod_{i=1}^{n} W_{E,\mathbf{P}}(le_i)^{v_i^2 - v_i(\Sigma_{j\neq i} v_j)} \prod_{1\leq i<j\leq n} W_{E,\mathbf{P}}(le_i + le_j)^{v_i v_j}.$$

Note that

$$W_{E,\mathbf{P}}(le_i) = W_{E,P_i}(l), \qquad W_{E,\mathbf{P}}(le_i + le_j) = W_{E,P_i+P_j}(l).$$

Combining the above, we have

$$W_{E,l\mathbf{P}}(\mathbf{v}) = \frac{W_{E,\mathbf{v}\cdot\mathbf{P}}(l)W_{E,\mathbf{P}}(\mathbf{v})^{l^2}}{\prod_{i=1}^{n} W_{E,P_i}(l)^{v_i^2 - v_i(\Sigma_{j\neq i} v_j)} \prod_{1\leq i<j\leq n} W_{E,P_i+P_j}(l)^{v_i v_j}}.$$

Comparing this in the case of $l = q-1$ and $l = q-1+m$ gives the required result, as before. □

Corollary 19.1.2. *Suppose that E is an elliptic curve over a field $K = \mathbb{F}_q$ and $P \in E(K)$ is of order $m \geq 4$. The period of the sequence $W_{E,P}$ is $m\,\mathrm{ord}_{K^*}(\phi(P))$.*

Proof. First, $\phi([n]P)$ has period exactly m. Since, if the period were $m' < m$, then $W_{E,P}(m') = 0$, a contradiction. The result then follows directly from equation (19.2). □

19.2 Some hard problems

Elliptic nets are closely related to points on elliptic curves. We have already seen in several cases that computations relating to elliptic curves (such as pairings) can be carried out by computations of the associated elliptic nets. We will define several computational problems for elliptic nets.

Problem 19.2.1 (EDS Association). *Let E be an elliptic curve over a finite field K. Suppose one is given points $P, Q \in E(K)$ such that $Q \in \langle P\rangle$, $Q \neq \mathcal{O}$, and $\mathrm{ord}(P) \geq 4$. Determine $W_{E,P}(k)$ for the value of $0 < k < \mathrm{ord}(P)$ such that $Q = [k]P$.*

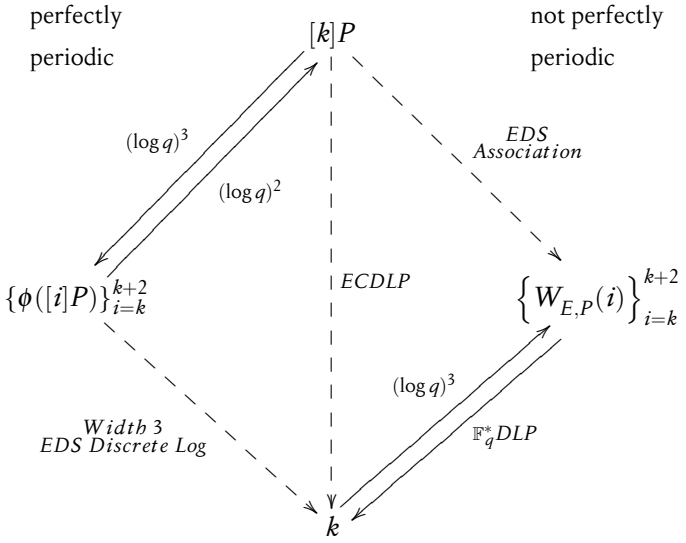
Problem 19.2.2 (EDS Residue). *Let E be an elliptic curve over a finite field K. Suppose one is given points $P, Q \in E(K)$ such that $Q \in \langle P\rangle$, $Q \neq \mathcal{O}$, and $\mathrm{ord}(P) \geq 4$. Determine the quadratic residuosity of $W_{E,P}(k)$ for the value of $0 < k < \mathrm{ord}(P)$ such that $Q = [k]P$.*

Problem 19.2.3 (Width s EDS Discrete Log). *Given an elliptic divisibility sequence W and terms $W(k)$, $W(k+1)$, …, $W(k+s-1)$, determine k.*

Recall that a perfectly periodic elliptic divisibility sequence is one which has a finite period n and whose first positive index k at which $W(k) = 0$ is $k = n$. If a sequence is not perfectly periodic, then it has $n > k$.

Note that the choice of segment $0 < k < \mathrm{ord}(P)$ is not crucial in Problem 19.2.1 (EDS Association): it could be restated for any segment $i\,\mathrm{ord}(P) < k < (i+1)\,\mathrm{ord}(P)$. This problem is trivial for a perfectly periodic sequence or net (since $\widetilde{W}(k) = \phi(Q)$ is computable in $\log q$ time). For the non-perfectly periodic case, the problem appears to be much harder. As for Problem 19.2.3 (EDS Discrete Log), on the other hand, for non-perfectly periodic elliptic divisibility sequences, it can be solved by computing an $\mathbb{F}_q^*$ discrete log. For this problem, it is the case of perfect periodicity that seems very difficult.

We will see that these hard problems are related according to the following diagram.



We demonstrate the complexity of solving the problems associated to the solid lines in the following series of theorems. The solid line labelled $\mathbb{F}_q^*$DLP has the complexity of a discrete logarithm problem in $\mathbb{F}_q^*$ (this is sub-exponential by index calculus). No sub-exponential algorithms are known for the dotted lines.

Lemma 19.2.4. *Let E be an elliptic curve defined over K, and $P \in E(K)$ be a point of order not less than 4. The x-coordinate of $[n]P$, denoted $x([n]P)$, can be calculated from $W_{E,P}(n-1)$, $W_{E,P}(n)$, and $W_{E,P}(n+1)$ in a fixed number of field operations.*

Proof. From Lemma 6.2.2:

$$\frac{W_{E,P}(n-1)W_{E,P}(n+1)}{W_{E,P}(n)^2} = x(P) - x([n]P). \tag{19.4}$$

□

Theorem 19.2.5 (Shipsey [61]). *Let E be an elliptic curve over K, and $P \in E(K)$ a point of order not less than 4. Given a value m, the term $W_{E,P}(m)$ in the elliptic divisibility sequence associated to E, P can be calculated in $O((\log m)(\log q)^2)$ time.*

Proof. For completeness, we give a simplified version of Shipsey's algorithm here. Following Shipsey, denote by $\left\langle W_{E,P}(n)\right\rangle$ the segment or *block centred at k* of eight terms $W_{E,P}(k-3)$, $W_{E,P}(k-2)$, …, $W_{E,P}(k+3)$, $W_{E,P}(k+4)$ of the sequence. The block centred at m can be calculated from the block centred at 1 via a double-and-add algorithm based on an addition chain for m. The calculation of the new block from the previous depends on two instances of the recurrence (one such calculation for each term of the new block):

$$W(2i-1,0) = W(i+1,0)W(i-1,0)^3 - W(i-2,0)W(i,0)^3 \ ,$$
$$W(2i,0) = (W(i,0)W(i+2,0)W(i-1,0)^2 - W(i,0)W(i-2,0)W(i+1,0)^2)/W(2,0) \ .$$

To begin we must calculate the block centred at 1. Recalling that $W(0)=0$, $W(1)=1$ and $W(-n) = -W(n)$, we must calculate $W(i)$ for $i=2,3,4$. Formulæ are given in Proposition 6.1.4. This algorithm takes $O(\log m)$ steps, each of which involves a fixed number of $\mathbb{F}_q^*$ multiplications and additions, which take $O((\log q)^2)$ time at worst. □

Theorem 19.2.6. *Let E be an elliptic curve over K, and $P \in E(K)$ a point of order not less than* 4. *Given a point $Q=[k]P$, the term $\phi(Q) = \widetilde{W}(k)$ can be calculated in $O((\log q)^3)$ time.*

Proof. The formula for $\phi(Q)$ requires calculating two terms of $W_{E,Q}$, which, by Theorem 19.2.5, takes $\log(q-1+\mathrm{ord}(Q))$ steps. Since $\mathrm{ord}(Q)$ is on the order of q, this takes $O((\log q)^3)$ time at worst. The other necessary operation is to find the inverse of $\mathrm{ord}(Q)^2$ modulo $q-1$, and to raise to that exponent. Both these are also $O(\log q)$ operations. □

Theorem 19.2.7. *Let E be an elliptic curve over K, and $P \in E(K)$ a point of order not less than* 4. *Given terms $\widetilde{W}(k), \widetilde{W}(k+1), \widetilde{W}(k+2)$, in a perfectly periodic sequence associated to E, P, the point $Q=[k]P$ can be calculated in $O((\log q)^2)$ time.*

Proof. This follows from Lemma 19.2.4. Note that the left hand side of the expression (19.4) is invariant under an elliptic divisibility sequence equivalence. Therefore we can calculate $x([k+1]P)$. Now we must determine which of the two points with this x-coordinate is actually $[k+1]P$. First, take one of the two candidate points, and proceed on the assumption that it is $[k+1]P$. Using the addition formula for elliptic curves, calculate $x([k+1]P+P) = x([k+2]P)$. Compare this with (19.4) to determine $\widetilde{W}(k+3)$. Also determine $\widetilde{W}(k+4)$ in this manner. Then, if the terms $\widetilde{W}(k),\ldots,\widetilde{W}(k+4)$ satisfy the recurrence instance

$$\widetilde{W}(k+4)\widetilde{W}(k) = \widetilde{W}(k+1)\widetilde{W}(k+3)\widetilde{W}(2)^2 - \widetilde{W}(3)\widetilde{W}(1)\widetilde{W}(k+2)^2,$$

our assumption about the point we chose is correct. If this recurrence does not hold, then the point we chose was incorrect, and the other one is the point $[k+1]P$ we seek. Finally, knowing $[k+1]P$, we can calculate $Q=[k]P=[k+1]P-P$. The number of operations in the field is bounded by a constant, hence the time taken is $O((\log q)^2)$ at worst. □

The following theorem is implicit in the work of Shipsey; see Section 19.3.2 for an explanation.

Theorem 19.2.8. *Suppose P has prime order not dividing $q-1$, and $\phi(P)$ is a primitive root in $\mathbb{F}_q^*$. Given $W_{E,P}(k), W_{E,P}(k+1), W_{E,P}(k+2)$, where it can be assumed that $0 < k < ord(P)$, calculating k can be reduced to a single discrete logarithm in $\mathbb{F}_q^*$ in $O((\log q)^3)$ time.*

Proof. We can deduce the x-coordinate of the point $Q = [k]P$ by Lemma 19.2.4. Choosing one of the two possible y-coordinates, we have either $Q = [k]P$ or $Q = [-k]P$. To determine which is correct, use the trick of the proof of Theorem 19.2.7. Suppose it is the former; then, from 19.1.1, we have

$$\frac{\phi([k+1]P)}{\phi([k]P)} = \phi(P)^{2k+1}\frac{W_{E,P}(k+1)}{W_{E,P}(k)}.$$

So k satisfies an equation of the form $A = B^{2k+1}$ where A and B are known, and B has order $q-1$. Therefore, we are reduced to solving a discrete logarithm of the form $A = B^x$ for $0 \leq x < q-1$, with the understanding that k will be one of $(x-1)/2$ or $(x+q-1)/2$. (In fact, if $q-1 < m$, there may be at most two other possible values of k to check: the above values plus $q-1$.) □

Remark 19.2.1. Let $m = \text{ord}(P)$. Suppose that $\gcd(m, q-1) = 1$. As an integer k ranges over representatives of a single coset in $\mathbb{Z}/m\mathbb{Z}$, it ranges over all possible cosets of $\mathbb{Z}/(q-1)\mathbb{Z}$. Therefore, we cannot expect to find the set of k such that $Q = [k]P$ (i.e., a coset in $\mathbb{Z}/m\mathbb{Z}$) by solving an equation of the form $A = B^k$ in $\mathbb{F}_q^*$ (i.e., solving modulo $q-1$). One solution to this problem is to attempt to solve for an *integer* k (instead of a coset) – say, for example, the smallest non-negative k with $Q = [k]P$. This is in essence what the preceeding theorem does. With this in mind, we set some terminology.

Definition 19.2.1. Let Q be a multiple of P on an elliptic curve E. The *minimal multiplier* of Q with respect to P is the smallest non-negative value of k such that $Q = [k]P$.

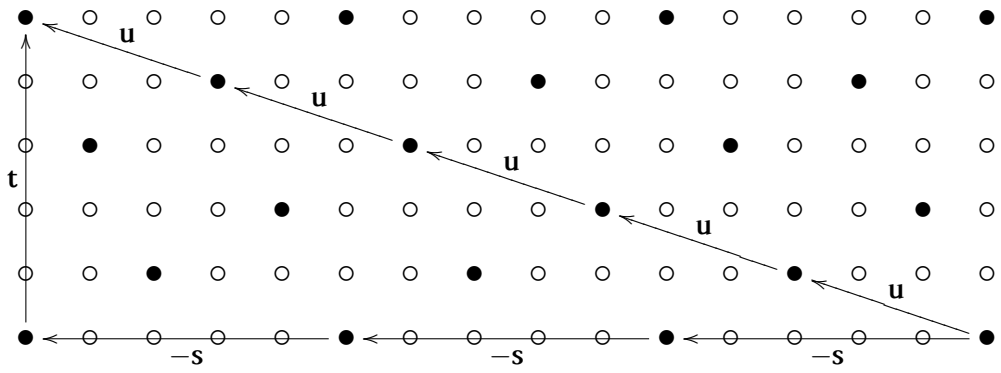
Note that the minimal multiplier satisfies $0 \leq k < \text{ord}(P)$.

19.3 The $\mathbb{F}_q^*$ discrete logarithm, The Tate-Lichtenbaum pairing and MOV and Frey-Rück attacks

Theorem 19.2.8 uses terms of the elliptic divisiblity sequence to give a discrete logarithm problem in $\mathbb{F}_q^*$. We demonstrate some variations on this theme, and relate these types of equations to the Tate-Lichtenbaum pairing, and to an ECDLP attack given by Shipsey [61].

19.3.1 An $\mathbb{F}_q^*$ DLP equation of the form $A = B^k$ from periodicity properties

The $\mathbb{F}_q^*$ DLP equations we consider are consequences of Theorem 10.1.1, but many can be conveniently understood in terms of its corollary Theorem 10.2.3. The following example involves the terms $W_{E,P}(k)$ and $W_{E,P}(k+1)$, and requires knowledge of $Q = [k]P$. The following diagram is suggestive for the discussion.



In this picture of $\mathbb{Z}^2$, $\mathbf{u} = (-3,1)$, $\mathbf{s} = (5,0)$ and $\mathbf{t} = (0,5)$. Vectors $\mathbf{u}$ and $\mathbf{s}$ generate the lattice of zero-apparition Λ for some elliptic net W associated to points P and $Q = [3]P$ of order 5. The vector $\mathbf{t}$ is also in Λ. One coset of $\mathbb{Z}^2$ modulo Λ is shown as the solid discs.

Theorem 10.2.3 shows the transformation relative to translation by a vector $\mathbf{r} \in \Lambda$: it relates $W(\mathbf{v}+\mathbf{r})$ to $W(\mathbf{v})$ for each $\mathbf{v}$. This theorem can be applied repeatedly, and different 'paths' from one point to another must agree. In the picture above, the translation property which relates $W(\mathbf{v}+(-15,5))$ to $W(\mathbf{v})$ can be calculated by applying the transformation associated to $\mathbf{u}$ five times (the diagonal path) or by applying the transformation associated to $-\mathbf{s}$ three times followed by that associated to $\mathbf{t}$ once (the sides of the triangle).

In the general case, we have $Q = [k]P$. Then the lattice of zero-apparition Λ for $W = W_{E,P,Q}$ includes vectors $\mathbf{u} = (-k,1)$, $\mathbf{s} = (m,0)$ and $\mathbf{t} = (0,m)$. Suppose $\mathbf{r} = (r_1, r_2)$ is an element of Λ for $W = W_{E,P,Q}$. By Theorem 10.2.3, we have for all $l \in \mathbb{Z}$ and $\mathbf{v} \in \mathbb{Z}^2$,

$$W(l\mathbf{r}+\mathbf{v}) = W(\mathbf{v}) a_{\mathbf{r}}^{l\nu_1} b_{\mathbf{r}}^{l\nu_2} c_{\mathbf{r}}^{l^2} \tag{19.5}$$

where

$$a_{\mathbf{r}} = \frac{W(r_1+2,r_2)}{W(r_1+1,r_2)W(2,0)}, \qquad b_{\mathbf{r}} = \frac{W(r_1,r_2+2)}{W(r_1,r_2+1)W(0,2)}, \qquad c_{\mathbf{r}} = \frac{W(r_1+1,r_2+1)}{a_{\mathbf{r}} b_{\mathbf{r}} W(1,1)}.$$

We expect appropriate relationships between $a_{\mathbf{u}}$, $b_{\mathbf{u}}$, $c_{\mathbf{u}}$, $a_{\mathbf{s}}$, $b_{\mathbf{s}}$, etc. The $\mathbb{F}_p^*$ DLP equation we seek is one such relationship. We have

$$a_{\mathbf{s}} = \frac{W(m+2,0)}{W(m+1,0)W(2,0)}, \qquad a_{\mathbf{t}} = \frac{W(2,m)}{W(1,m)W(2,0)}, \qquad a_{\mathbf{u}} = \frac{W(2-k,1)}{W(1-k,1)W(2,0)}.$$

For each $i \in \mathbb{Z}$, we apply (19.5) to obtain

$$\frac{W(-ik+1,i-1)W(0,-1)}{W(1,-1)W(-ik,i-1)} = a_{\mathbf{u}}^i \tag{19.6}$$

Set $i = m$ in (19.6), and apply (19.5) four times:

$$\begin{aligned} a_{\mathbf{u}}^{m} &= \frac{W(-mk+1,m-1)W(0,-1)}{W(1,-1)W(-mk,m-1)} \\ &= \left(\frac{W(-mk+1,m-1)}{W(-mk+1,-1)}\right)\left(\frac{W(-mk+1,-1)}{W(1,-1)}\right)\left(\frac{W(0,-1)}{W(-mk,-1)}\right)\left(\frac{W(-mk,-1)}{W(-mk,m-1)}\right) \\ &= \frac{a_{\mathbf{t}}^{-mk+1}b_{\mathbf{t}}^{-1}c_{\mathbf{t}}^{1}a_{\mathbf{s}}^{-k}b_{\mathbf{s}}^{k}c_{\mathbf{s}}^{k^2}}{a_{\mathbf{t}}^{-mk}b_{\mathbf{t}}^{-1}c_{\mathbf{t}}^{1}a_{\mathbf{s}}^{0}b_{\mathbf{s}}^{k}c_{\mathbf{s}}^{k^2}} = a_{\mathbf{t}}a_{\mathbf{s}}^{-k} \end{aligned}$$

Setting $i = 1$ in (19.6), we obtain an expression

$$a_{\mathbf{u}} = \frac{W(-k+1,0)W(0,-1)}{W(1,-1)W(-k,0)} = -\frac{W_{E,P}(k-1)}{W_{E,P}(k)W(1,-1)}$$

which, when substituted into the last calculation, yields

$$\left(\frac{W(m+1,0)W(2,0)}{W(m+2,0)}\right)^{k} = \left(\frac{W_{E,P}(k-1)}{W_{E,P}(k)}\right)^{m}\left(-\frac{W(1,m)W(2,0)}{W(2,m)W(1,-1)^{m}}\right). \tag{19.7}$$

19.3.2 An $\mathbb{F}_q^*$ DLP equation from Shipsey's thesis

The possibility of such an equation was observed by Rachel Shipsey in her thesis [61, p.80]. She uses one-dimensional periodicity properties to derive the following equation:

$$\frac{W_{E,P}((m+1)(k+1))W_{E,P}(k)}{W_{E,P}((m+1)k)W_{E,P}(k+1)} = W_{E,P}(m+1)^{2k+1} \tag{19.8}$$

Shipsey then argues that without knowledge of k the left hand side can be calculated up to a factor of

$$\left(\frac{W_{E,P}(k)}{W_{E,P}(k-1)}\right)^{m(m+2)}.$$

This is very much of the same spirit as equation (19.7), and in fact, Theorem 10.1.1 can be used to rewrite (19.8) in this form:

$$\frac{W_{E,P,Q}(m+1,m+1)}{W_{E,P,Q}(0,m+1)}\left(\frac{W_{E,P}(k+1)}{W_{E,P}(k)}\right)^{m(m+2)} = W_{E,P}(m+1)^{2k+1}. \tag{19.9}$$

By Lemma 19.2.4, knowledge of $Q, W_{E,P}(k), W_{E,P}(k-1)$ determines $W_{E,P}(k+1)$, and so this is very much equivalent to Shipsey's analysis. Note that the unknown terms in (19.9) are raised to the exponent $m+2$. At first blush, this may appear to lead to an ECDLP attack for $q-1 = m+2$ (where the unknown terms will disappear). However, this is not allowed by Remark 19.2.1. In fact, it turns out that if $q-1 = m+2$, then $W_{E,P}(m+1) = 1$ (this eventually follows from Theorem 10.1.1 also).

19.3.3 $\mathbb{F}_q^*$ DLP equations and the Tate-Lichtenbaum pairing

The Tate-Lichtenbaum pairing and Weil pairing are used in the MOV [45] and Frey-Rück [23] attacks on the ECDLP. These use the Weil and Tate-Lichtenbaum pairings, respectively, to translate an instance of the ECDLP into an $\mathbb{F}_q^*$ DLP equation, where index calculus methods may be used. The basic

idea, illustrated here for the Tate-Lichtenbaum pairing, is that $Q = [k]P$ implies $\tau_m(Q,S) = \tau_m(P,S)^k$ by bilinearity. If S can be chosen so that $\tau_m(P,S)$ is non-trivial, and if the Tate-Lichtenbaum pairing takes values in a manageably small finite field, then index calculus methods can be used to determine k. In particular, this attack applies for curves E over $\mathbb{F}_q$ where $m = q-1$.

In (19.9) and (19.7), all the terms may be calculated from knowledge of m, P and Q except for $W_{E,P}(k)$ and $W_{E,P}(k-1)$. However, notice that these unknown terms are raised to the power m. Therefore, in the case that $m = q-1$, no extra information is needed and the ECDLP is reduced to an $\mathbb{F}_q^*$ DLP; this works in exactly the cases that the MOV or Frey-Rück attack applies.

These sorts of 'alternate versions' of the MOV/Frey-Rück attack do have a relation to the Tate-Lichtenbaum pairing. In light of Theorem 17.2.1, equations (19.7) and (19.9) can be re-written as statements in terms of the Tate-Lichtenbaum pairing.

For convenience, we restate Theorem 17.2.1.

Theorem 19.3.1 (Restatement of Theorem 17.2.1)**.** *Let E be an elliptic curve, $m \geq 4$, and $P_1 \in E[m]$. Let $P_2, P_3 \in E$ be such that $P_3 \notin \{\mathcal{O}, P_2\}$. Let W be an elliptic net of rank n, associated to points $\mathbf{T} \in E(K)^n$. Let $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3 \in Z^n$ be such that $P_i = \mathbf{p}_i \cdot \mathbf{T}$ for each i. Let $\tau_m : E[m] \times E/mE \to K^*/(K^*)^m$ be the Tate pairing. Then*

$$\tau_m(P_1,P_2) = \frac{W(m\mathbf{p}_1+\mathbf{p}_2+\mathbf{p}_3)W(\mathbf{p}_3)}{W(m\mathbf{p}_1+\mathbf{p}_3)W(\mathbf{p}_2+\mathbf{p}_3)}.$$

Equation (19.7)**:** Apply Theorem 19.3.1 twice, each time with basis $\mathbf{T} = (P,Q)$. First, for the left-hand side, use $P_1 = P, P_2 = -P, P_3 = [2]P$ and coordinates $\mathbf{p}_1 = (1,0), \mathbf{p}_2 = (-1,0), \mathbf{p}_3 = (2,0)$. For the right-hand side, use $P_1 = Q, P_2 = -P, P_3 = [2]P$, and coordinates $\mathbf{p}_1 = (0,1), \mathbf{p}_2 = (-1,0), \mathbf{p}_3 = (2,0)$. This rewrites (19.7) as

$$\tau_m(P,-P)^k = \tau_m(Q,-P).$$

Equation (19.9)**:** This is somewhat more complicated. From Theorem 10.2.2 with $m = q-1$ and Theorem 19.3.1 with various parameters,

$$W_{E,P}(m+1)^2\tau_m(P,P)^{-2} = \left(\frac{W_{E,P}(m+1)^2 W_{E,P}(2)}{W_{E,P}(m+2)}\right)^2 = b^2 = a^m = 1,$$

$$\tau_m(P,Q) = \frac{W_{E,P,Q}(m+1,1)W_{E,P,Q}(1,0)}{W_{E,P,Q}(m+1,0)W_{E,P,Q}(1,1)}, \qquad \tau_m(Q,P) = \frac{W_{E,P,Q}(1,m+1)W_{E,P,Q}(0,1)}{W_{E,P,Q}(0,m+1)W_{E,P,Q}(1,1)},$$

$$1 = \tau_m(P,\mathcal{O}) = \tau_m(P,[m]Q) = \frac{W_{E,P,Q}(m+1,m+1)W_{E,P,Q}(1,1)}{W_{E,P,Q}(m+1,1)W_{E,P,Q}(1,m+1)}.$$

All of which, taken together, rewrites (19.9) as

$$\tau_m(P,Q)\tau_m(Q,P) = \tau_m(P,P)^{2k}.$$

Equation (19.2) does not, however, lend itself to this sort of re-writing in terms of pairings, as it requires the assumption that $\gcd(m, q-1) = 1$. If we were to redefine it without taking m^2-th roots (in order to avoid this assumption), the equation becomes effectively trivial.

19.4 ECDLP through EDS Association

The previous sections have demonstrated that there are a variety of ways to translate an ECDLP into an $\mathbb{F}_q^*$ DLP. The $\mathbb{F}_q^*$ DLP equation is in terms of elements of the sequence $W_{E,P}$. For example in (19.7), the elements are $W_{E,P}(k)$ and $W_{E,P}(k-1)$. The problem of finding these terms (with knowledge of $Q=[k]P$ but not k) is *EDS Association*. In this example, however, it is only their quotient that is needed. Depending on the form of the $\mathbb{F}_q^*$ DLP equation, different such information (certain terms or ratios of terms) suffices. We formalise the most general statement of this in the following theorem.

Proposition 19.4.1. *Fix an elliptic curve E defined over $\mathbb{F}_q$, and $P \in E(\mathbb{F}_q)$ of order greater than three and relatively prime to $q-1$. Suppose $\phi(P)$ has order $q-1$ in $\mathbb{F}_q^*$. With knowledge of any product*

$$\prod_{i=1}^{N} W_{E,P}(p_i(k))^{e_i},$$

where the $e_i \in \mathbb{Z}$, and $p_i(x) \in \mathbb{Z}[x]$, and $t(x) = \sum_{i=1}^{N} e_i p_i(x)^2$ is a non-constant linear polynomial in $\mathbb{Z}[x]$, the value of k can be determined in subexponential time in q.

Proof. By Theorem 19.1.1, $t(k)$ satisfies an equation in $\mathbb{F}_q^*$ of the form $A = B^{t(k)}$. The left hand side A is the known product in the hypothesis of the theorem, while $B = \phi(P)$ (whose computation takes time $O((\log q)^3)$ by Theorem 19.2.6). Solving this discrete logarithm for $t(k)$ can be done sub-exponentially by index calculus methods. Solving for k from $t(k)$ is direct since $t(k)$ is linear in k. □

It is evident that the most costly step is the index calculus step, which in many cases has run time $r(q) = \exp(c(\log q)^{1/3}(\log\log q)^{2/3})$ [13, p.306].

19.5 ECDLP and quadratic residues

We will show that determining only one bit of information – the residuosity – about a term $W_{E,P}(k)$ may suffice to solve the ECDLP. First, we observe a hypothetical method of attack for ECDLP.

Proposition 19.5.1. *Let P be a point of odd order relatively prime to $q-1$. Given an oracle which can determine the parity of the minimal multiplier of any non-zero point Q in $\langle P\rangle$ in time $O(T(q))$, the elliptic curve discrete logarithm for any such Q can be determined in time $O(T(q)\log q + (\log q)^2)$.*

Proof. Suppose that k is the minimal multiplier of Q with respect to P. The basic algorithm is:

1. If $Q = P$, stop.
2. Call the oracle to determine the parity of k. If k is even, find Q' such that $[2]Q' = Q$. If k is odd, find Q' such that $[2]Q' = Q - P$.
3. Set $Q = Q'$ and return to step 1.

In Step 2, since the cyclic group $\langle P \rangle$ has odd order, there is a unique Q'. It can be found in $O(\log q)$ time (see [36] for methods). Furthermore, $Q' = [k']P$ where

$$k' = \begin{cases} k/2 & k \text{ even} \\ (k-1)/2 & k \text{ odd} \end{cases}.$$

Then k' is the minimal multiplier for Q' with respect to P. At the end of this process, the value of the original k can be deduced from the sequence of steps taken. For each even step, record a '0', and for each odd step a '1', writing from right to left, and adding a final '1': this will be the binary representation of k. The number of steps is $\log_2 k = O(\log q)$. □

Proposition 19.5.2. *Fix an elliptic curve E defined over $\mathbb{F}_q$ of characteristic not equal to two, and $P \in E(\mathbb{F}_q)$ of order greater than three and relatively prime to $q-1$. Suppose that $\phi(P)$ is a quadratic non-residue. Then, with knowledge of the quadratic residuosity of any product of the form*

$$\prod_{i=1}^{N} W_{E,P}(p_i(k))^{e_i}, \tag{19.10}$$

where the $e_i \in \mathbb{Z}$, and $p_i(x) \in \mathbb{Z}[x]$ of degree at most D, and $t(x) = \sum_{i=1}^{N} e_i p_i(x)^2$ is not constant as a function $\mathbb{Z}/2\mathbb{Z} \to \mathbb{Z}/2\mathbb{Z}$, the parity of k can be determined in time $O(D)$.

Proof. By Theorem 19.1.1, the value $t(k)$ satisfies an equation in $\mathbb{F}_q^*$ of the form $A = B^{t(k)}$. The quadratic residuosity of A is known. Now, $B = \phi(P)$ is a quadratic non-residue. The parity of $t(k)$ can be calculated from these values in constant time (i.e., consider the question in K^* modulo $(K^*)^2$). The parity of k is determined by checking the parity of $t(0)$ and $t(1)$. This final step takes time $O(D)$. □

Corollary 19.5.3. *Let E be an elliptic curve over a field of characteristic not equal to two. Let P be a point of odd order such that $\phi(P)$ is a quadratic non-residue, and let k be the minimal multiplier of a multiple Q of P. Given P, Q and an oracle which can determine the quadratic residuosity of $W_{E,P}(k)$ in time $O(T(q))$, the elliptic curve discrete logarithm for any such Q can be determined in time $O(T(q)\log q + (\log q)^2)$.*

Proof. This follows from Proposition 19.5.2 with $N = 1, e_1 = 1, p_1(x) = x$ and Proposition 19.5.1. □

A few remarks are in order.

1. The hypotheses on the $t(x)$ of Proposition 19.5.2 and Proposition 19.4.1 are mutually exclusive.
2. If $\phi(P)$ is a quadratic residue, one solution to this obstacle is to replace the initial problem of $Q = [k]P$ with the equivalent problem of $[n]Q = [k]([n]P)$ for any n such that $\phi([n]P)$ is a quadratic non-residue. The perfectly periodic sequence can be calculated term-by-term until such an n is found.
3. It may be tempting to try to apply this method to the case that the order of P divides $q-1$. Unfortunately, this is not possible. If the order m of the group $\langle P \rangle$ is even, multiplication by 2 is not an automorphism, and so there is no unique 'half' of a point (this is the same difficulty

that prevents this sort of parity attack on an $\mathbb{F}_q^*$ discrete log). If $m|(q-1)$ is odd, then k satisfies a discrete logarithm equation of the form $A = B^k$ in the group $K^*/(K^*)^m$, which has an odd number of elements. Therefore, this does not determine the parity of k.

19.6 The EDS Residue problem

In light of the preceeding section, it is natural to define the problem of EDS Residue (Problem 19.2.2). In Section 19.8 we will show that it is equivalent to the elliptic curve discrete logarithm in subexponential time. How might one determine the quadratic residuosity of $W_{E,P}(k)$? Our first observation is that knowledge of the residuosity of one term $W_{E,P}(k)$ would determine the residuosity of the next term.

Proposition 19.6.1. *Suppose Q is a known element of $\langle P \rangle$, but that its minimal multiplier k is unknown. The quadratic residuosity of $W_{E,P}(k+1)/W_{E,P}(k)$ can be calculated in $O((\log q)^3)$ time.*

Proof. From (19.2) with $n = k$ and $n = k+1$, we have

$$\frac{\phi(Q)}{\phi(Q+P)} = \phi(P)^{2k+1} \left(\frac{W_{E,P}(k+1)}{W_{E,P}(k)} \right).$$

The calculation of the terms $\phi(P), \phi(Q)$, and $\phi(P+Q)$ each take $O((\log q)^3)$ time. □

Therefore, based on knowledge of Q but not k, the sequence

$$S(n) = \left(\frac{W_{E,P}(n)}{q} \right) \left(\frac{W_{E,P}(k)}{q} \right)$$

for $n = k, \ldots, k+N$ may be may be calculated in $O(N \log q)$ time. Then the sequence

$$\left(\frac{W_{E,P}(n)}{q} \right)$$

is either $S(n)$ or $-S(n)$. To determine which is to determine the quadratic residuosity of $W_{E,P}(k)$.

Therefore, if some bias, or some pattern, for quadratic residues of the elliptic divisibility sequence $W_{E,P}(n)$ were known, then the correct choice of the two sequences above could be determined. However, as yet we have no evidence to suggest that the ratio of quadratic residues among the terms is not $1/2$ in general.

Question 19.6.2. *What proportion of terms in an elliptic divisibility sequence or elliptic net over a finite field of odd characteristic are quadratic residues?*

19.7 ECDLP through EDS Discrete Log in the case of perfect periodicity

Problem 19.2.3 (EDS Discrete Log) is less unusual in flavour than the other problems considered here: general discrete logarithm attacks will apply. Recall the proof of Theorem 19.2.5, in which *blocks*

centred at k are defined – denote this as $B(k)$. From $B(k)$, the recurrence relation can be used to calculate $B(2k)$ or $B(2k+1)$. In fact, Shipsey goes further, and shows how two blocks $B(k), B(k')$ can be added to obtain a block $B(k+k')$ in a similarly efficient manner (see [61, p. 23]). This means that the sequence of blocks $B(n)$ is a sequence along which we can move easily by addition and $\mathbb{Z}$-multiplication. Therefore, algorithms such as Baby-Step-Giant-Step and Pollard's ρ can be applied to this problem.

19.8 Equivalence of hard problems

Theorem 19.8.1. *Let E be an elliptic curve over a finite field $K = \mathbb{F}_q$ of characteristic $\neq 2$. If any one of the following problems is solvable in sub-exponential time, then all of them are:*

1. *Problem 1.5.1: ECDLP*
2. *Problem 1.5.2: EDS Association for non-perfectly periodic sequences*
3. *Problem 1.5.3: EDS Residue for non-perfectly periodic sequences*
4. *Problem 1.5.4 ($s = 3$): Width 3 EDS Discrete Log for perfectly periodic sequences*

Proof. (3) $\implies$ (1): Corollary 19.5.3.

(1) $\implies$ (2): If k is known, we can assume $0 < k \leq \mathrm{ord}(P)$, and then $W_{E,P}(k)$ can be calculated in $O((\log k)(\log q)^2) = O((\log q)^3)$ time.

(2) $\implies$ (3): Residuosity of a value in $\mathbb{F}_q^*$ can be determined in sub-exponential time (see [33] for algorithms).

(1) $\implies$ (4): Theorem 19.2.7.

(4) $\implies$ (1): Theorem 19.2.6 allows calculation of $\phi([k]P)$, $\phi([k+1]P)$, and $\phi([k+2]P)$ in sub-exponential time. □

Part V

Appendices

Appendix A

Formulary

Let E be the elliptic curve defined over the rationals with Weierstrass equation

$$y^2 + a_1 xy + a_3 y - x^3 - a_2 x^2 - a_4 x - a_6 = 0.$$

As usual, let

$$\begin{aligned} b_2 &= a_1^2 + 4a_2, \quad b_4 = 2a_4 + a_1 a_3, \\ b_6 &= a_3^2 + 4a_6, \quad b_8 = a_1^2 a_6 + 4a_2 a_6 - a_1 a_3 a_4 + a_2 a_3^2 - a_4^2. \end{aligned}$$

A.1 Elliptic net recurrence relation

The elliptic net recurrence relation:

$$\begin{aligned} &W(p+q+s)\,W(p-q)\,W(r+s)\,W(r) \\ &\quad + W(q+r+s)\,W(q-r)\,W(p+s)\,W(p) \\ &\quad + W(r+p+s)\,W(r-p)\,W(q+s)\,W(q) = 0. \end{aligned} \tag{3.1}$$

Nelson form: (from (3.1) by $s \leftarrow 2a$, $r \leftarrow b-a$, $p \leftarrow c-a$, $q \leftarrow d-a$)

$$\begin{aligned} &W(a+b)\,W(a-c)\,W(c+d)\,W(c-d) \\ &\quad + W(a+c)\,W(a-c)\,W(d+b)\,W(d-b) \\ &\quad + W(a+d)\,W(a-d)\,W(b+c)\,W(b-c) = 0. \end{aligned} \tag{3.2}$$

Ward's elliptic divisibility sequences recurrence relation: (from (3.1) by $p \leftarrow n$, $q \leftarrow m$, $s \leftarrow 0$, $r \leftarrow 1$).

$$W(n+m)W(n-m)W(1)^2 = W(n+1)W(n-1)W(m)^2 - W(m+1)W(m-1)W(n)^2. \tag{1.2}$$

Miscellaneous special cases:

$$W(p+q)W(p-q)W(r)^2 = W(p+r)W(p-r)W(q)^2 - W(q+r)W(q-r)W(p)^2,$$
$$W(n+2)W(n-2)W(1)^2 = W(n+1)W(n-1)W(2)^2 - W(3)W(1)W(n)^2,$$
$$\begin{aligned} W(n+m+1)W(n-m)W(2)W(1) = {} & W(n+2)W(n-1)W(m+1)W(m) \\ & - W(m+2)W(m-1)W(n+1)W(n), \end{aligned}$$
$$\begin{aligned} W(n+3)W(n-2)W(2)W(1) = {} & W(n+2)W(n-1)W(3)W(2) \\ & - W(4)W(1)W(n+1)W(n), \end{aligned}$$
$$W(2n)W(2)W(1)^2 = W(n)\Big(W(n+2)W(n-1)^3 - W(n-2)W(n+1)^2\Big), \quad (2.6)$$
$$W(2n+1)W(1)^3 = W(n+2)W(n)^3 - W(n-1)W(n+1)^3, \quad (2.7)$$
$$\begin{aligned} W(nm)W(2) = {} & W\left(\tfrac{nm}{2}\right)\Big(W\left(\tfrac{nm}{2}+2\right)W\left(\tfrac{nm}{2}-1\right)^2 \\ & - W\left(\tfrac{nm}{2}-2\right)W\left(\tfrac{nm}{2}+1\right)^2\Big), \end{aligned} \quad (2.8)$$
$$\begin{aligned} W(nm)W(n) = {} & W\left(\tfrac{n(m+1)}{2}+1\right)W\left(\tfrac{n(m+1)}{2}-1\right)W\left(\tfrac{n(m-1)}{2}\right)^2 \\ & - W\left(\tfrac{n(m-1)}{2}+1\right)W\left(\tfrac{n(m-1)}{2}-1\right)W\left(\tfrac{n(m+1)}{2}\right)^2, \end{aligned} \quad (2.9)$$
$$W(1,-1)W(1,1)^3 = W(0,1)^3W(2,1) - W(1,0)^3W(1,2), \quad (4.2)$$
$$W(2i-1,0) = W(i+1,0)W(i-1,0)^3 - W(i-2,0)W(i,0)^3, \quad (18.1)$$
$$\begin{aligned} W(2i,0)W(2,0) = {} & W(i,0)W(i+2,0)W(i-1,0)^2 \\ & - W(i,0)W(i-2,0)W(i+1,0)^2, \end{aligned} \quad (18.2)$$
$$\begin{aligned} W(2k-1,1)W(1,1) = {} & W(k+1,1)W(k-1,1)W(k-1,0)^2 \\ & - W(k,0)W(k-2,0)W(k,1)^2, \end{aligned} \quad (18.3)$$
$$\begin{aligned} W(2k,1) = {} & W(k-1,1)W(k+1,1)W(k,0)^2 \\ & - W(k-1,0)W(k+1,0)W(k,1)^2, \end{aligned} \quad (18.4)$$
$$\begin{aligned} W(2k+1,1)W(-1,1) = {} & W(k-1,1)W(k+1,1)W(k+1,0)^2 \\ & - W(k,0)W(k+2,0)W(k,1)^2, \end{aligned} \quad (18.5)$$
$$\begin{aligned} W(2k+2,1)W(2,-1) = {} & W(k+1,0)W(k+3,0)W(k,1)^2 \\ & - W(k-1,1)W(k+1,1)W(k+2,0)^2. \end{aligned} \quad (18.6)$$

A.2 Complex function formulæ

Weierstrass σ-function definition of net polynomials:

1. $n = 1$:

$$\Omega_\nu(z;\Lambda) = \frac{\sigma(\nu z;\Lambda)}{\sigma(z;\Lambda)^{\nu^2}}. \quad (5.3)$$

2. $n = 2$:

$$\Omega_{u,v}(z,w;\Lambda) = \frac{\sigma(uz+vw;\Lambda)}{\sigma(z;\Lambda)^{u^2-uv}\sigma(z+w;\Lambda)^{uv}\sigma(w;\Lambda)^{v^2-uv}}. \tag{5.4}$$

3. general n:

$$\Omega_{\mathbf{v}}(\mathbf{z};\Lambda) = \frac{\sigma(v_1z_1+\ldots+v_nz_n;\Lambda)}{\prod_{i=1}^{n}\sigma(z_i;\Lambda)^{2v_i^2-\Sigma_{j=1}^{n}v_iv_j}\prod_{\substack{1\le i,j\le n\\ i\neq j}}\sigma(z_i+z_j;\Lambda)^{v_iv_j}}. \tag{5.2}$$

Complex function identities:

$$\wp(z)-\wp(w) = -\frac{\sigma(z+w)\sigma(z-w)}{\sigma(z)^2\sigma(w)^2}, \tag{5.5}$$

$$\wp(nz)-\wp(mz) = -\frac{\Omega_{m+n}(z)\Omega_{m-n}(z)}{\Omega_m(z)^2\Omega_n(z)^2}, \tag{5.6}$$

$$\zeta(x+a)-\zeta(a)-\zeta(x+b)+\zeta(b) = \frac{\sigma(x+a+b)\sigma(x)\sigma(a-b)}{\sigma(x+a)\sigma(x+b)\sigma(a)\sigma(b)}, \tag{5.7}$$

$$\zeta(x+a+b)-\zeta(x+a)-\zeta(x+b)+\zeta(x) = \frac{\sigma(2x+a+b)\sigma(a)\sigma(b)}{\sigma(x+a+b)\sigma(x+a)\sigma(x+b)\sigma(x)}. \tag{5.8}$$

A.3 Net polynomials

1. for $n = 1$:

$$\Psi_1 = 1, \tag{6.8}$$

$$\Psi_2 = 2y+a_1x+a_3, \tag{6.9}$$

$$\Psi_3 = 3x^4+b_2x^3+3b_4x^2+3b_6x+b_8, \tag{6.10}$$

$$\Psi_4 = (2y+a_1x+a_3)(2x^6+b_2x^5+5b_4x^4+10b_6x^3+10b_8x^2+(b_2b_8-b_4b_6)x+b_4b_8-b_6^2); \tag{6.11}$$

2. for $n = 2$:

$$\Psi_{(1,-1)} = x_2-x_1, \tag{6.12}$$

$$\Psi_{(2,1)} = 2x_1+x_2-\left(\frac{y_2-y_1}{x_2-x_1}\right)^2-a_1\left(\frac{y_2-y_1}{x_2-x_1}\right)+a_2, \tag{6.13}$$

$$\Psi_{(2,-1)} = (y_1+y_2)^2-(2x_1+x_2)(x_1-x_2)^2\ ; \tag{6.14}$$

3. for $n = 3$:

$$\Psi_{(1,1,1)} = \frac{y_1(x_2 - x_3) + y_2(x_3 - x_1) + y_3(x_1 - x_2)}{(x_1 - x_2)(x_1 - x_3)(x_2 - x_3)}, \tag{6.15}$$

$$\Psi_{(-1,1,1)} = \frac{y_1(x_2 - x_3) - y_2(x_3 - x_1) - y_3(x_1 - x_2)}{(x_2 - x_3)} + a_1 x_1 + a_3, \tag{6.16}$$

$$\Psi_{(1,-1,1)} = \frac{-y_1(x_2 - x_3) + y_2(x_3 - x_1) - y_3(x_1 - x_2)}{(x_3 - x_1)} + a_1 x_2 + a_3, \tag{6.17}$$

$$\Psi_{(1,1,-1)} = \frac{-y_1(x_2 - x_3) - y_2(x_3 - x_1) + y_3(x_1 - x_2)}{(x_1 - x_2)} + a_1 x_3 + a_3. \tag{6.18}$$

A.4 Formulæ relating curves and nets

Define

$$\phi_m = x(P)\Psi_m^2 - \Psi_{m+1}\Psi_{m-1}, \tag{2.3}$$

$$4y\omega_m = \Psi_{m+2}\Psi_{m-1}^2 - \Psi_{m-2}\Psi_{m+1}^2. \tag{2.4}$$

Then

$$[m]P = \left(\frac{\phi_m(P)}{\Psi_m(P)^2}, \frac{\omega_m(P)}{\Psi_m(P)^3}\right), \tag{2.5}$$

$$x([m]P) - x([n]P) = -\frac{\Psi_{m+n}(P)\Psi_{m-n}(P)}{\Psi_m^2(P)\Psi_n^2(P)}. \tag{6.21}$$

Curve from sequence (see Section 8.2):

1. in rank $n = 1$:

$$C : y^2 + a_1 xy + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6, \qquad P = (0,0),$$

where

$$a_1 = \frac{W(4) + W(2)^5 - 2W(2)W(3)}{W(2)^2 W(3)}$$

$$a_2 = \frac{W(2)W(3)^2 + W(4) + W(2)^5 - W(2)W(3)}{W(2)^3 W(3)}$$

$$a_3 = W(2), \qquad a_4 = 1, \qquad a_6 = 0$$

2. in rank $n = 2$:

$$C : y^2 + a_1 xy + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6, \qquad P_1 = (0,0), \qquad P_2 = (W(2,1) - W(1,2), 0),$$

where

$$a_1 = \frac{W(2,0) - W(0,2)}{W(2,1) - W(1,2)}, \qquad a_2 = 2W(2,1) - W(1,2), \qquad a_3 = W(2,0)$$

$$a_4 = (W(2,1) - W(1,2))W(2,1), \qquad a_6 = 0$$

3. alternative in rank $n = 2$ and characteristic $\neq 2$:

$$C : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, \qquad P_1 = (v, 0), \qquad P_2 = (-v, 0),,$$

where

$$2v = W(2,1) - W(1,2),$$
$$a_1 = \frac{W(2,0) - W(0,2)}{W(2,1) - W(1,2)}, \qquad 2a_2 = W(2,1) + W(1,2), \qquad 2a_3 = W(2,0) + W(0,2)$$
$$4a_4 = -(W(2,1) - W(1,2))^2, \qquad 8a_6 = -(W(2,1) - W(1,2))^2(W(2,1) + W(1,2))$$

A.5 Transformation property for elliptic nets

Let T be any $n \times m$ matrix. Let $\mathbf{P} \in E^m$, $\mathbf{v} \in \mathbb{Z}^n$.

$$W_{E,\mathbf{P}}(T^{tr}(\mathbf{v})) = W_{E,T(\mathbf{P})}(\mathbf{v}) \prod_{i=1}^{n} W_{E,\mathbf{P}}(T^{tr}(\mathbf{e}_i))^{v_i^2 - v_i(\Sigma_{j \neq i} v_j)} \prod_{1 \leq i < j \leq n} W_{E,\mathbf{P}}(T^{tr}(\mathbf{e}_i + \mathbf{e}_j))^{v_i v_j} \tag{10.2}$$

A.6 Partial periodicity

Periodicity formulæ for non-degenerate elliptic nets:

1. rank $n = 1$ with $W_{E,P}(r) = 0$:

$$W_{E,P}(sr + k) = W_{E,P}(k) a^{sk} b^{s^2} \tag{10.3}$$

where

$$a = \frac{W_{E,P}(r+2)}{W_{E,P}(r+1) W_{E,P}(2)}, \qquad b = \frac{W_{E,P}(r+1)^2 W_{E,P}(2)}{W_{E,P}(r+2)} \tag{10.4}$$

2. rank $n = 2$ with $W_{E,P,Q}(\mathbf{r}) = 0$:

$$W_{E,P,Q}(l\mathbf{r} + \mathbf{k}) = W_{E,P,Q}(\mathbf{k}) a_{\mathbf{r}}^{lk_1} b_{\mathbf{r}}^{lk_2} c_{\mathbf{r}}^{l^2} \tag{10.10}$$

where

$$a_{\mathbf{r}} = \frac{W_{E,P,Q}(r_1 + 2, r_2)}{W_{E,P,Q}(r_1 + 1, r_2) W_{E,P,Q}(2,0)}, \quad b_{\mathbf{r}} = \frac{W_{E,P,Q}(r_1, r_2 + 2)}{W_{E,P,Q}(r_1, r_2 + 1) W_{E,P,Q}(0,2)}, \tag{10.11}$$

$$c_{\mathbf{r}} = \frac{W_{E,P,Q}(r_1 + 1, r_2 + 1)}{a_{\mathbf{r}} b_{\mathbf{r}} W_{E,P,Q}(1,1)}. \tag{10.12}$$

Perfectly periodic elliptic divisibility sequence and elliptic net over $\mathbb{F}_q$:

$$\phi(P) = \left(\frac{W_{E,P}(q-1)}{W_{E,P}(q - 1 + \mathrm{ord}(P))} \right)^{\frac{1}{\mathrm{ord}(P)^2}}, \tag{19.1}$$

$$\phi(\mathbf{v} \cdot \mathbf{P}) = W_{E,\mathbf{P}}(\mathbf{v}) \prod_{i=1}^{n} \phi(P_i)^{v_i^2 - v_i\left(\Sigma_{j \neq i} v_j\right)} \prod_{1 \leq i < j \leq n} \phi(P_i + P_j)^{v_i v_j}. \tag{19.3}$$

A.7 Elliptic net biextension factor system

$$\Lambda(P,Q,R) = \frac{\mathcal{W}(p+q+r)\mathcal{W}(p)\mathcal{W}(q)\mathcal{W}(r)}{\mathcal{W}(p+q)\mathcal{W}(q+r)\mathcal{W}(r+p)} \tag{15.1}$$

A.8 Tate-Lichtenbaum and Weil pairing formulæ

$$\tau_m(P,Q) = \frac{\mathcal{W}(mp+q+s)\mathcal{W}(s)}{\mathcal{W}(mp+s)\mathcal{W}(q+s)}, \tag{17.3}$$

$$e_m(P,Q) = \frac{\mathcal{W}(mp+q+s)\mathcal{W}(p+s)\mathcal{W}(mq+s)}{\mathcal{W}(mp+s)\mathcal{W}(q+s)\mathcal{W}(p+mq+s)} \tag{17.5}$$

Special cases:

$$\tau_m(P,P) = \frac{W_P(m+2)W_P(1)}{W_P(m+1)W_P(2)} \ , \tag{18.7}$$

$$\tau_m(P,Q) = \frac{W_{P,Q}(m+1,1)W_{P,Q}(1,0)}{W_{P,Q}(m+1,0)W_{P,Q}(1,1)} \ . \tag{18.8}$$

A.9 Discrete logarithm type equations

Suppose $[m]P = \mathcal{O}$ and $Q = [k]P$.

$$\left(\frac{W_{E,P,Q}(m+1,0)W_{E,P,Q}(2,0)}{W_{E,P,Q}(m+2,0)}\right)^k = \left(\frac{W_{E,P}(k-1)}{W_{E,P}(k)}\right)^m \left(-\frac{W_{E,P,Q}(1,m)W_{E,P,Q}(2,0)}{W_{E,P,Q}(2,m)W_{E,P,Q}(1,-1)^m}\right), \tag{19.7}$$

$$W_{E,P}(m+1)^{2k+1} = \frac{W_{E,P,Q}(m+1,m+1)}{W_{E,P,Q}(0,m+1)}\left(\frac{W_{E,P}(k+1)}{W_{E,P}(k)}\right)^{m(m+2)} . \tag{19.9}$$

Appendix B

PARI/GP scripts

B.1 Computations with elliptic divisibility sequences

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ PARI/GP Script for Elliptic Divisibility Sequences   v. 2.0        \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ This script performs various manipulations related to elliptic     \\
\\ divisibility sequences.                                            \\
\\                                                                    \\
\\ See http://www.math.brown.edu/~stange/                             \\
\\ or contact <stange at math dot brown dot edu> for info.            \\
\\                                                                    \\
\\ Throughout this script, it is assumed the sequences take values    \\
\\ in a field.                                                        \\
\\                                                                    \\
\\ Feel free to distribute this script.                               \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

\\ Set debug = 1 for some information on what is happening
debug = 0;
```

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ SECTION ONE:  TRANSLATING BETWEEN CURVES AND SEQUENCES   \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ curvetoeds(curve,point)
\\
\\ Given an elliptic curve 'curve' and a point 'point' on that curve,
\\ returns the first four terms of the associated elliptic divisibility
\\ sequences as a row vector.
\\
\\ The curve may be given either in initialized or non-initialized form.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\


{

curvetoeds(curve, point) =

        local(bvals, initvals);

        bvals = vector(8);
        initvals = vector(4);

        \\ the variable 'bvals' stores the usual b_2, b_4, b_6 and b_8
        \\ defined for elliptic curves
        \\ (see Silverman, Arithmetic of Elliptic Curves, 1986, p. 46)

        bvals[2] = curve[1]^2 + 4 * curve[2];
        bvals[4] = 2*curve[4] + curve[1]*curve[3];
        bvals[6] = curve[3]^2 + 4*curve[5];
        bvals[8] = curve[1]^2*curve[5] + 4*curve[2]*curve[5]
                - curve[1]*curve[3]*curve[4] + curve[2]*curve[3]^2
                - curve[4]^2;
```

```
        \\ the variable 'initvals' stores the first four terms of the
        \\ sequence; these formulae work for general Weierstrass eqns

        initvals[1] = 1;
        initvals[2] = 2*point[2] + curve[1]*point[1] + curve[3];
        initvals[3] = 3*point[1]^4 + bvals[2]*point[1]^3
                + 3*bvals[4]*point[1]^2 + 3*bvals[6]*point[1]
                + bvals[8];
        initvals[4] = initvals[2]*( 2*point[1]^6 + bvals[2]*point[1]^5
                + 5*bvals[4]*point[1]^4 + 10*bvals[6]*point[1]^3
                + 10*bvals[8]*point[1]^2 + (bvals[2]*bvals[8]
                - bvals[4]*bvals[6])*point[1] +bvals[4]*bvals[8]
                - bvals[6]^2);

        return(initvals);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edstocurve(eds)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
\\ form of a vector of four terms (1st through 4th terms), with first
\\ term one, returns a curve such that that curve has the point (0,0)
\\ on it and the pair are associated to that sequence.  (I.e. eds
\\ is associated to edstocurve(eds) and point [0,0].)
\\
\\ The curve returned is not initialised.
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\
\\ This uses formulae due to Christine Swart in her thesis, "Elliptic
\\ Curves and related sequences", University of London, 2003.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
{

edstocurve(eds) =

        local(curve,a,b,c);
        curve = vector(5);
        a = eds[2];
        b = eds[3];
        c = eds[4];

        if(eds[1] != 1, print("First term not 1"); return(0); );

        curve[1] = (c+a^5 - 2*a*b)/(a^2*b);
        curve[2] = (a*b^2 + c + a^5 - a*b)/(a^3*b);
        curve[3] = a;
        curve[4] = 1;
        curve[5] = 0;

        return(curve);

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ elllong(shortcurve)
\\
\\ Given a vector representing a cubic curve, calculates the standard
\\ values bi, ci, delta and j. (See Silverman, Arithmetic of Elliptic
\\ Curves, 1986, p. 46.)
\\
\\ The vector returned is of the form
\\
\\ [a1,a2,a3,a4,a6,b2,b4,b6,b8,c4,c6,delta,j]
\\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
{

elllong(shortcurve) =

        local(a1,a2,a3,a4,a6,b2,b4,b6,b8,c4,c6,delta,j);

        a1 = shortcurve[1];
        a2 = shortcurve[2];
        a3 = shortcurve[3];
        a4 = shortcurve[4];
        a6 = shortcurve[5];

        b2 = a1^2 + 4*a2;
        b4 = 2*a4 + a1*a3;
        b6 = a3^2 + 4*a6;
        b8 = a1^2*a6 + 4*a2*a6 - a1*a3*a4 + a2*a3^2 - a4^2;

        c4 = b2^2 - 24*b4;
        c6 = -b2^3 + 36*b2*b4 - 216*b6;

        delta = -b2^2*b8 - 8*b4^3 - 27*b6^2 + 9*b2*b4*b6;

        if( delta == 0,
                j = "inf";
        ,
                j = c4^3/delta;
        );

        return([a1,a2,a3,a4,a6,b2,b4,b6,b8,c4,c6,delta,j]);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edstocurvefull(eds)
\\
```

```
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
\\ form of a vector of four terms (1st through 4th terms), with first
\\ term one, returns a curve such that that curve has the point (0,0)
\\ on it and the pair are associated to that sequence.  (I.e. eds
\\ is associated to edstocurve(eds) and point [0,0].)
\\
\\ The vector returned is of the form
\\
\\ [a1,a2,a3,a4,a6,b2,b4,b6,b8,c4,c6,delta,j]
\\
\\ where the constants are as denoted in Silverman, Arithmetic of
\\ Elliptic Curves, 1986, p. 46
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\
\\ This uses formulae due to Christine Swart in her thesis, "Elliptic
\\ Curves and related sequences", University of London, 2003.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\


{

edstocurvefull(eds) =


        if(eds[1] != 1, print("First term not 1"); return(0); );

        return(elllong(edstocurve(eds)));

}






\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsjinv(eds)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
```

```
\\ form of a vector of four terms (1st through 4th terms), with first
\\ term one, returns the j-invariant of the associated curve.
\\
\\ If first term is not one, sequence is scaled to make it one.
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
edsjinv(eds) =

        if(eds[1] != 1, eds = eds/eds[1]; );

        return( edstocurvefull(eds)[13] );

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsdisc(eds)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
\\ form of a vector of four terms (1st through 4th terms), with first
\\ term one, returns the discriminant of the associated curve.
\\
\\ If first term is not one, sequence is scaled to make it one.
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
edsdisc(eds) =

        if(eds[1] != 1, eds = eds/eds[1]; );

        return( edstocurve(eds)[12] );
```

```
}



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsc4(eds)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
\\ form of a vector of four terms (1st through 4th terms), with first
\\ term one, returns the value c4 of the associated curve.
\\
\\ If first term is not one, sequence is scaled to make it one.
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
edsc4(eds) =

        if(eds[1] != 1, eds = eds/eds[1]; );

        return( edstocurve(eds)[10] );

}



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsc6(eds)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
\\ form of a vector of four terms (1st through 4th terms), with first
\\ term one, returns the value c6 of the associated curve.
\\
\\ If first term is not one, sequence is scaled to make it one.
```

```
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
edsc6(eds) =

        if(eds[1] != 1, eds = eds/eds[1]; );

        return( edstocurve(eds)[11] );

}
```

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsgtwo(eds)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
\\ form of a vector of four terms (1st through 4th terms), with first
\\ term one, returns the value g_2 for the associated curve.
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\
\\ If first term is not one, sequence is scaled to make it one.
\\
\\ These use formulae due to M. Ward in his paper "Memoir on Elliptic
\\ Divisibility Sequences."
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edsgtwo(eds)=

        local(a,b,c);
```

```
        a = eds[2];
        b = eds[3];
        c = eds[4];

        if(eds[1] != 1, eds = eds/eds[1]; );

        return(
                (a^20 + 4*a^15*c - 16*a^12*b^3 + 6*a^10*c^2
                - 8*a^7*b^3*c + 4*a^5*c^3 + 16*a^4*b^6 + 8*a^2*b^3*c^2
                + c^4)/(12*a^8*b^4)
        );

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsgthree(eds)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
\\ form of a vector of four terms (1st through 4th terms), with first
\\ term one, returns the value g_3 for the associated curve.
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\
\\ If first term is not one, sequence is scaled to make it one.
\\
\\ These use formulae due to M. Ward in his paper "Memoir on Elliptic
\\ Divisibility Sequences."
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\


{

edsgthree(eds)=

        local(a,b,c);
```

```
        a = eds[2];
        b = eds[3];
        c = eds[4];

        if(eds[1] != 1, eds = eds/eds[1]; );

        return(
                - (a^30 + 6*a^25*c - 24*a^22*b^3 + 15*a^20*c^2
                - 60*a^17*b^3*c + 20*a^15*c^3 + 120*a^14*b^6
                - 36*a^12*b^3*c^2 + 15*a^10*c^4
                - 48*a^9*b^6*c + 12*a^7*b^3*c^3 + 64*a^6*b^9
                + 6*a^5*c^5 + 48*a^4*b^6*c^2 + 12*a^2*b^3*c^4
                + c^6)/(216*a^12*b^6)
        );


}









\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ SECTION TWO:  CALCULATING VALUES OF SEQUENCES            \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsget(eds,n)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds' in the
\\ form of a vector of four terms (1st through 4th terms), and an integer
\\ n, returns the value of the sequence at index n.
\\
\\ Uses Shipsey's method (see edsblockships below).
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edsget(eds,n) =

        if(debug, print("Calling edsget for W(", n, ")"); );

        if( n==0, return(0); );
        if( n < 5 && n > 0, return(eds[n]); );
        if( n < 0, return( -edsget(eds,-n) ); );
        return( edsblockships(eds,n)[5] );

}









\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsgen(eds,len)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds'
\\ (in the form of the first four terms as a vector), and a positive
\\  integer 'len', returns a vector of length 'len' containing the
\\ first len terms of the sequence.
\\
```

```
\\ Uses the recurrence to generate terms one after another linearly.
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\


{

edsgen(eds, len) =

        local(X, i);

        \\ Stupid to ask it to generate a shorter sequence than 4
        if( len < 4,
                print("Length too short, returning eds.");
                return(eds);
        );

        X = vector(len);

        \\ The first four terms of the resulting vector are
        \\ already at hand
        X[1]=eds[1];
        X[2]=eds[2];
        X[3]=eds[3];
        X[4]=eds[4];

        \\ For the rest of the terms, use the recurrence relation
        for(i=5,len,

                if( X[i-4] != 0,
                \\ In this case it is safe to divide by X[i-4]

                        X[i] = (X[i-3]*X[i-1]*X[2]^2
                                - X[i-2]^2*X[1]*X[3])
                                /X[i-4]/X[1]/X[1];

                ,
                \\ If X[i-4] == 0 then X[i-5] != 0 for
                \\ non-degenerate sequences
```

```
                        X[i] = (X[i-1]*X[i-4]*X[2]*X[3]
                                - X[i-2]*X[i-3]*X[1]*X[4])
                                /X[i-5]/X[2]/X[1];

                );

        );

        return(X);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsblocklin(eds, len)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds'
\\ (in the form of the first four terms as a vector), and an integer
\\ 'len', returns a vector of length five containing the terms
\\ len-4 up through len of the sequence.
\\
\\ Uses the recurrence to generate terms one after another linearly.
\\ Very slow runtime for large len (it is O(len)).
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\
\\ Accepts negative len.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edsblocklin(eds, len) =


        local(X,Xnew,Xbase,i);
```

```
X = vector(5);
Xnew = vector(5);
Xbase = vector(5);

\\ elliptic divisibility sequences satisfy
\\ an antisymmetry property, so we calculate out in the
\\ positive direction and then reverse the block and put
\\ on minus signs
if( len < 0,

        Xnew = edsblocklin(eds, -len+4);
        for(i=1,5,
                X[i] = -Xnew[6-i];
        );

        return(X);

);

\\ 'X' stores at each round the most recent block of
\\ five terms; start with the terms of eds,
\\ plus the zeroth term, which is 0
X[1]=0;
X[2]=eds[1];
X[3]=eds[2];
X[4]=eds[3];
X[5]=eds[4];


\\ if we are requesting a block very close to the origin,
\\ the calculation is easy
if( len == 4, return(X); );
if( len == 3, return([-X[2],0,X[2],X[3],X[4]]); );
if( len == 2, return([-X[3],-X[2],0,X[2],X[3]]); );
if( len == 1, return([-X[4],-X[3],-X[2],0,X[2]]); );
if( len == 0, return([-X[5],-X[4],-X[3],-X[2],0]); );
```

```
        \\ 'Xbase' keeps a permanent record of these first five terms
        \\  for use in the recurrence
        Xbase = X;

        \\ Loop up to requested length, updating X each time to
        \\ represent shifting one term along the sequence
        for(i=1,len-4,

                X = edsblockincrement(X,Xbase);

        );

        return(X);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsblockships(eds, len)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds'
\\ (in the form of the first four terms as a vector), and an integer
\\ 'len', returns a vector of length 4 containing the terms
\\ len-4 up through len of the sequence.
\\
\\ Uses Shipsey's double-and-add method to generate the final terms in
\\ O(log (len) ) time.  Very fast.
\\ Her method is described in her thesis (Rachel Shipsey,
\\ Elliptic Divisibility Sequences, University of London, 2000)
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\
\\ Accepts negative len.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
{

edsblockships(eds, len) =

        local(X, Xnew, Xbase, Xfinal, i, j, doubleaddchainlength,
                 doubleaddchain, scalefactor, remainder);


        \\ elliptic divisibility sequences satisfy an
        \\ antisymmetry property, so we calculate out in the
        \\ positive direction and then reverse the block and put
        \\ on minus signs
        if( len < 0,
                X = vector(5);
                Xnew = vector(5);

                Xnew = edsblockships(eds, -len+4);
                for(i=1,5,
                        X[i] = -Xnew[6-i];
                );

                return(X);

        );

        \\ if we are requesting a block very close to the origin,
        \\ the calculation is easy
        if( len == 4, return([0,eds[1],eds[2],eds[3],eds[4]]); );
        if( len == 3, return([-eds[1],0,eds[1],eds[2],eds[3]]); );
        if( len == 2, return([-eds[2],-eds[1],0,eds[1],eds[2]]); );
        if( len == 1, return([-eds[3],-eds[2],-eds[1],0,eds[1]]); );
        if( len == 0, return([-eds[4],-eds[3],-eds[2],-eds[1],0]); );



        \\ Throughout, we use blocks (segments) of length 8 of
        \\ the sequence.
        \\ A block is 'based on' k if it represents terms k-3 up
```

```
\\ through k+4 of the sequence
X = vector(8);
Xnew = vector(8);
Xbase = vector(4);


\\ if eds doesn't have first term one, we have to scale it
\\ to one that does and then un-scale at the end
scalefactor = eds[1];
for(i=1,4,
        eds[i] = eds[i]/scalefactor;
);

\\to make sure final block contains len-4 up through len
len = len-1;


\\ the initial terms of the sequence are stored in 'Xbase'
Xbase[1]=eds[1];
Xbase[2]=eds[2];
Xbase[3]=eds[3];
Xbase[4]=eds[4];
if( eds[2] == 0,
        invtwo = "inf";
,
        \\ the inverse of the second term is precomputed
        invtwo = 1/eds[2];
);

\\ 'X' starts as a block based on 1
\\ (i.e. a block of length 8 whose 4th term is
\\ the first term of the sequence)
X[1] = -eds[2];
X[2] = -eds[1];
X[3] = 0;
X[4] = eds[1];
X[5] = eds[2];
X[6] = eds[3];
X[7] = eds[4];
```

```
X[8] = eds[2]^3*eds[4] - eds[3]^2*eds[1]*eds[3];

\\ The length of the double-add chain is computed
doubleaddchainlength = ceil(log(len+1)/log(2));

\\ The chain will be stored in this vector
doubleaddchain = vector(doubleaddchainlength);

\\ compute the double-add-chain
remainder = len;
i = 0;
while(remainder !=0,
        bit = lift(Mod(remainder,2));
        doubleaddchain[doubleaddchainlength - i] = bit;
        remainder = (remainder-bit)/2;
        i = i+1;
);

\\ Show the double-add-chain if debug data is on
if( debug == 1,
        print("Double and add chain is: " doubleaddchain);
 );

\\ Loop through the double-add-chain doubling or adding
for(j=2,doubleaddchainlength,

        if(doubleaddchain[j] == 0,

                \\ We double the block
                \\ (get block based on 2*k from block
                \\ based on k)
                Xnew[1] = X[4]*X[2]^3 - X[1]*X[3]^3;
                Xnew[3] = X[5]*X[3]^3 - X[2]*X[4]^3;
                Xnew[5] = X[6]*X[4]^3 - X[3]*X[5]^3;
                Xnew[7] = X[7]*X[5]^3 - X[4]*X[6]^3;
                if( invtwo != "inf",
                        Xnew[2] = X[3]*(X[5]*X[2]^2
                                        -X[1]*X[4]^2)*invtwo;
                        Xnew[4] = X[4]*(X[6]*X[3]^2
```

```
                                -X[2]*X[5]^2)*invtwo;
                Xnew[6] = X[5]*(X[7]*X[4]^2
                                -X[3]*X[6]^2)*invtwo;
                Xnew[8] = X[6]*(X[8]*X[5]^2
                                -X[4]*X[7]^2)*invtwo;
        ,
                Xnew[2] = 0;
                Xnew[4] = 0;
                Xnew[6] = 0;
                Xnew[8] = 0;
        );
        X = Xnew;
,

        \\We double-add the block
        \\ (get block based on 2*k+1 from block
        \\ based on k)
        Xnew[2] = X[5]*X[3]^3 - X[2]*X[4]^3;
        Xnew[4] = X[6]*X[4]^3 - X[3]*X[5]^3;
        Xnew[6] = X[7]*X[5]^3 - X[4]*X[6]^3;
        Xnew[8] = X[8]*X[6]^3 - X[5]*X[7]^3;
        if( invtwo != "inf",
                Xnew[1] = X[3]*(X[5]*X[2]^2
                                -X[1]*X[4]^2)*invtwo;
                Xnew[3] = X[4]*(X[6]*X[3]^2
                                -X[2]*X[5]^2)*invtwo;
                Xnew[5] = X[5]*(X[7]*X[4]^2
                                -X[3]*X[6]^2)*invtwo;
                Xnew[7] = X[6]*(X[8]*X[5]^2
                                -X[4]*X[7]^2)*invtwo;
        ,
                Xnew[1] = 0;
                Xnew[3] = 0;
                Xnew[5] = 0;
                Xnew[7] = 0;
        );
        X = Xnew;
);
```

```
        );

        \\ Put the final data into a vector of length five and return
        Xfinal = vector(5);
        Xfinal[1] = X[1]*scalefactor;
        Xfinal[2] = X[2]*scalefactor;
        Xfinal[3] = X[3]*scalefactor;
        Xfinal[4] = X[4]*scalefactor;
        Xfinal[5] = X[5]*scalefactor;

        return(Xfinal);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsblockincrement(block,base)
\\
\\ Given an elliptic divisibility sequence 'base' in the form of a
\\ vector of five terms (0th through 4th terms) or four terms
\\ (1st through 4th), and a block
\\ of length 5 in that elliptic divisibility sequence or a Somos
\\ sequence translated from it,
\\ it calculates the block shifted one to the right.  The sequence
\\ must be non-degenerate.
\\
\\ Used as an internal function for edsblocklin and others.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edsblockincrement(block,base)=

        local(newblock);
```

```
        newblock = vector(5);

        \\ if base is a block of length four, make it length five
        if( length(base) == 4,
                base = [0,base[1],base[2],base[3],base[4]];
        );

        \\ calculate newest term and store in 'newblock'
        if( block[2] != 0, \\ safe to divide by block[2]

                newblock[5] = (block[3]*block[5]*base[3]^2
                                        - block[4]^2*base[2]*base[4])
                                        /block[2]/base[2]/base[2];

        ,
                \\ otherwise it is safe to divide by block[1]
                \\(both cannot be zero in a non-degenerate sequence)

                newblock[5] = (block[5]*block[2]*base[3]*base[4]
                                - block[4]*block[3]*base[2]*base[5])
                                /block[1]/base[3]/base[2];

        );

        \\ shift previous terms and store in 'newblock'
        newblock[4] = block[5];
        newblock[3] = block[4];
        newblock[2] = block[3];
        newblock[1] = block[2];

        return(newblock);


}
```

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ SECTION THREE:  TRANSFORMATIONS AND PROPERTIES OF SEQUENCES  \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsrankofapp(eds, len)
\\
\\ Given an elliptic divisibility sequence 'eds'
\\ (in the form of the first four terms as a vector), and a positive
\\  integer 'len', returns the index of the first positive-indexed zero
\\  term before index len+1 or 0 if none is found up to that distance
\\
\\ If 'len' is omitted or is zero, will run until it finds a zero or
\\ is interrupted.
\\
\\ Uses a linear method to calculate terms one after another until
\\ a zero is found.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edsrankofapp(eds, len) =

        local(X, Xnew, Xbase, i);

        X = vector(5);
        Xnew = vector(5);
        Xbase = vector(5);

        \\ 'X' stores at each round the most recent block of five terms
```

```
        \\ Start with the terms of eds, plus the zeroth term,
        \\ which is 0.  Return index if a zero is found.
        X[1]=0;
        for(i=1,4,
                X[i+1]=eds[i];
                if( X[i+1] == 0, return(i) );
        );

        \\ 'Xbase' keeps a permanent record of these first five terms
        \\ for use in the recurrence
        Xbase = X;

        if( len > 1,
                \\ loop up to requested length looking for a zero
                for(i=5,len,
                        X = edsblockincrement(X,Xbase);
                        if( X[5] == 0, return(i); );
                );
        ,
                \\ loop up to requested length looking for a zero
                i = 4;
                while(X[5] != 0,
                        X = edsblockincrement(X,Xbase);
                        i = i+1;
                );
                return(i);

        );

        return(0);

}
```

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsperiod(eds, len)
\\
\\ Given a non-degenerate elliptic divisibility sequence 'eds'
\\ (in the form of the first four terms as a vector), and a positive
\\  integer 'len', returns the period of the sequence if it is less
\\ than len+1, or 0 if the period is not found up to that distance.
\\
\\ Uses a linear method to calculate terms one after another until
\\ the period is found.
\\
\\ 'Non-degerate' means the first, second and third terms are non-zero.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edsperiod(eds, len) =

        local(X, Xnew, Xbase, i);

        X = vector(5);
        Xnew = vector(5);
        Xbase = vector(5);

        \\ 'X' stores at each round the most recent block of five terms
        \\ Start with the terms of eds, plus the zeroth term,
        \\ which is 0
        X[1]=0;
        for(i=1,4,
                X[i+1]=eds[i];
        );

        \\ 'Xbase' keeps a permanent record of these first five terms
        \\ for use in the recurrence
        Xbase = X;

        \\ loop up to requested length searching for a zero
        if( len > 0,
                for(i=1,len,
```

```
                        X = edsblockincrement(X,Xbase);
                        if( X == Xbase, return(i); );
                );
        ,
                i = 1;
                        X = edsblockincrement(X,Xbase);
                while(X != Xbase,
                        X = edsblockincrement(X,Xbase);
                        i = i+1;
                );
                return(i);

        );
        return(0);

}









\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edssubseq(eds, n)
\\
\\ Given an elliptic divisibility sequence 'eds' associated to P on E
\\ (in the form of the first four terms as a vector), and a positive
\\ integer 'n', the sequence associated to [n]P on E.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edssubseq(eds,n) =

        local(neweds, nterm);
```

```
        nterm = edsget(eds,n);
        neweds = vector(4);

        neweds[1] = 1;
        neweds[2] = edsget(eds,2*n)/nterm^4;
        neweds[3] = edsget(eds,3*n)/nterm^9;
        neweds[4] = edsget(eds,4*n)/nterm^16;

        return(neweds);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsmakeint(eds)
\\
\\ Given an elliptic divisibility sequence 'eds' associated to P on E
\\ of type "t_FRAC",
\\ return an eds equivalent to it but with integer values, and no
\\ unnecessary gcd.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edsmakeint(eds) =

        local(neweds, factors, factr,gg);

        neweds = eds;
        factr = 1;

        gg = gcd(numerator(neweds[2]),numerator(neweds[3]));
        while( gg != 1,
                neweds = edsequiv(neweds,g^(-1));
                gg = gcd(numerator(neweds[2]),numerator(neweds[3]));
        );
```

```
        while( type(neweds[4]) != "t_INT",
                factors = factor(denominator(neweds[4]));
                factr = 1;
                for(i=1,matsize(factors)[1],
                        factr = factr*factors[i,1];
                );
                neweds = edsequiv(neweds,factr);
        );


        return(neweds);

}







\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsequiv(eds, factr)
\\
\\ Given an elliptic divisibility sequence 'eds'
\\ return an eds equivalent to it by factor 'factor'^(n^2-1)
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edsequiv(eds, factr) =

        local(neweds);

        neweds = vector(4);
        neweds[1] = eds[1];

        for(i=2,4,
                neweds[i] = eds[i]*factr^(i^2-1);
        );
```

```
        return(neweds);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edsperfper(eds, stop)
\\
\\ Given an elliptic divisibility sequence 'eds'
\\ return an eds equivalent to it which has rank of apparition equal
\\ to its period, if one is found before stop is reached.  (The
\\ algorithm loops through equivalences by i=1 to stop.  If stop=0,
\\ it will go until one is found (potentially never).
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\


{

edsperfper(eds,stop) =

        local(i,r);

        r = edsrankofapp(eds);
        if(debug, print("Rank of apparition is: ", r););

        i=1;
        while(edsget(edsequiv(eds,i),r+1) != 1 && i != stop,
                i = i+1;
        );

        if( i == stop,
                return(0);
        );

        return(edsequiv(eds,i));
```

```
}
```

B.2 Computations with rank two elliptic nets

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ PARI/GP Script for Rank Two Elliptic Nets            v. 1.0        \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ This script performs various manipulations related to elliptic     \\
\\ nets.  It requires the script for elliptic divisibility sequences  \\
\\ edstools.gp version 2.0.                                           \\
\\                                                                    \\
\\ See http://www.math.brown.edu/~stange/                             \\
\\ or contact <stange at math dot brown dot edu> for info             \\
\\                                                                    \\
\\ Throughout this script, it is assumed the nets take values         \\
\\ in a field.  Sometimes this field is required to have              \\
\\ characteristic not equal to two.  Many things will work            \\
\\ for general rings, but no guarantees there or anywhere.            \\
\\ Zero divisors in particular are a big problem.                     \\
\\                                                                    \\
\\ The functions in this script are restricted to rank 2.             \\
\\                                                                    \\
\\ A rank two elliptic net is represented as a vector of four         \\
\\ entries:                                                           \\
\\                                                                    \\
\\ [ W(2,0), W(0,2), W(2,1), W(1,2) ]                                 \\
\\                                                                    \\
\\ and is called 'non-degenerate' if none of the following occurs     \\
\\ 1) W(2,1) = W(1,2)                                                 \\
\\ 2) W(2,0) = W(2,1) = 0                                             \\
\\ 3) W(0,2) = W(1,2) = 0                                             \\
```

```
\\ Degenerate curves are generally not allowed anywhere.             \\
\\                                                                    \\
\\ Feel free to distribute this script.  If you alter it, add a       \\
\\ description of the alteration.  Acknowledge my original version.   \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

\r edstools

\\ Set debug = 1 for some information on what is happening
global(debug);
debug = 0;




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ SECTION ONE:  TRANSLATING BETWEEN CURVES AND NETS        \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ curvetotwonet(curve,pointa,pointb)
\\
\\ Given an elliptic curve 'curve' and a points 'pointa' and 'pointb'
\\ on that curve,
\\ returns the terms (2,0), (0,2), (2,1), (1,2) of the net in that
\\ order as a vector of length four.
\\
\\ Will return '0' in the case that pointa, pointb, pointa+pointb or
\\ pointa-pointb is the zero point on the curve.
\\
\\ The curve may be given either in initialised or non-initialised form.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
```

```
curvetotwonet(curve, pointa, pointb) =

        local(initvals);

        \\ The vector initvals will hold the terms
        \\ (2,0), (0,2), (2,1), (1,2)
        initvals = vector(4);


        \\ It makes no sense to form a net with P, Q, P+Q or P-Q trivial
        if( pointa == [0] || pointb == [0],
                if(debug,
                        print("curvetotwonet does not accept zero
                                points");
                );
                return(0);
        );

        if( pointa[1] == pointb[1],
                if(debug,
                        print("curvetotwonet does not accept points
                                which are equal or inverses");
                );
                return(0);
        );


        \\ Formulae for net polynomials gives the initial values
        initvals[1] = 2*pointa[2] + curve[1]*pointa[1] + curve[3];
        initvals[2] = 2*pointb[2] + curve[1]*pointb[1] + curve[3];
        initvals[3] = 2*pointa[1] + pointb[1] - ((pointb[2]
                - pointa[2])/(pointb[1] - pointa[1]))^2
                - curve[1]*((pointb[2] - pointa[2])/(pointb[1]
                - pointa[1])) + curve[2];
        initvals[4] = 2*pointb[1] + pointa[1] - ((pointb[2]
                - pointa[2])/(pointb[1] - pointa[1]))^2
                - curve[1]*((pointb[2] - pointa[2])/(pointb[1]
                - pointa[1])) + curve[2];
```

```
        \\ Error trap: the resulting net is degenerate, which
        \\ should only happen
        \\ if P+Q or P-Q is [0] (trapped earlier)
        if(initvals[3] == initvals[4],
                if(debug, print("The resulting net is degenerate.
                          This should have been caught earlier."); );
                print("ERROR, PLEASE REPORT THIS BUG 4301982357
                         to stange@math.brown.edu");
        );

        return(initvals);

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonettocurve(net)
\\
\\ Given a non-degenerate elliptic net 'net' in the
\\ form of a vector of length four (terms (2,0), (0,2), (2,1), (1,2)),
\\ returns a vector of length two whose first component is a length
\\ five vector representing a curve, and whose second component is
\\ a value 'x' such that that curve has the point (x,0) and (-x,0)
\\ on it and the triple are associated to that net.  (I.e. net
\\ is associated to curve twonettocurve(net)[1] and points [x,0]
\\ and [-x,0].)
\\
\\ The curve returned is not initialised.
\\
\\ The field must be of characteristic not equal to two.
\\
\\ This uses formulae from my thesis "Elliptic Nets and Elliptic
\\ Curves."
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

twonettocurve(net) =
```

```
        local(curve,n20,n02,n21,n12,xcoord, returnvector);

        \\ set up return vector (curve and x coordinate)
        returnvector = vector(2);
        curve = vector(5);

        \\ Return zero if the net is degenerate.
        if( istwonetdegen(net),
                if(debug, print("Degenerate nets taste bad!"); );
                return(0);
        );

        \\ gives names to net vals for ease of formulae
        n20 = net[1];
        n02 = net[2];
        n21 = net[3];
        n12 = net[4];


        \\ Coefficients of curve
        curve[1] = (n20 - n02)/(n21 - n12);
        curve[2] = (n21 + n12)/2;
        curve[3] = (n20 + n02)/2;
        curve[4] = -(n21 - n12)^2/4;
        curve[5] = -(n21 - n12)^2*(n21 + n12)/8;

        \\ X coordinate of first point (and negative of second)
        xcoord = (n21 - n12)/2;

        \\ Create vector to return (curve and coordinate)
        returnvector = [curve, xcoord];

        return(returnvector);

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
\\ twonettocurvechar2(net)
\\
\\ Given a non-degenerate elliptic net 'net' in the
\\ form of a vector of length four (terms (2,0), (0,2), (2,1), (1,2)),
\\ returns a vector of length two whose first component is a length
\\ five vector representing a curve, and whose second component is
\\ a value 'x' such that that curve has the point (0,0) and (x,0)
\\ on it and the triple are associated to that net.  (I.e. net
\\ is associated to curve twonettocurve(net)[1] and points [0,0]
\\ and [x,0].)
\\
\\ The curve returned is not initialised.
\\
\\ The field may be of characteristic equal to two (or may not).
\\
\\ This uses formulae from my thesis "Elliptic Nets and Elliptic
\\ Curves."
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

twonettocurvechar2(net) =

        local(curve,n20,n02,n21,n12,xcoord, returnvector);

        \\ Set up return vector (curve and x coordinate)
        returnvector = vector(2);
        curve = vector(5);

        \\ Catch degenerate nets (return zero)
        if( istwonetdegen(net),
                if(debug, print("Degenerate nets taste bad!"); );
                return(0);
        );

        \\ Give names to variables for ease of formulae
        n20 = net[1];
        n02 = net[2];
        n21 = net[3];
```

```
        n12 = net[4];

        \\ Coefficients of curve
        curve[1] = (n20 - n02)/(n21 - n12);
        curve[2] = 2*n21 - n12;
        curve[3] = (n20);
        curve[4] = (n21 - n12)*n21;
        curve[5] = 0;

        \\ X coordinate of second point
        xcoord = (n21-n12);

        \\ create vector of curve and coordinate
        returnvector = [curve, xcoord];

        return(returnvector);

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonetjinv(net)
\\
\\ Given a non-degenerate elliptic net 'net' in the
\\ form of a vector of four terms,
\\ returns the j-invariant of the associated curve.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
twonetjinv(net) =

        return( elllong(twonettocurvechar2(net)[1])[13] );

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonetdisc(net)
\\
```

```
\\ Given a non-degenerate elliptic net 'net' in the
\\ form of a vector of four terms,
\\ returns the discriminant of the associated curve.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
twonetdisc(net) =

        return( elllong(twonettocurvechar2(net)[1])[12] );

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ SECTION TWO:  RELATING NETS AND SEQUENCES                \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonettoeds(net,a,b)
\\
\\ Given a non-degenerate elliptic net 'net' in the
\\ form of a vector of four terms,
\\ returns the normalised elliptic divisibility sequence
\\ associated to the direction (a,b).  That is, it returns
\\ [1, W(2a,2b)*W(a,b)^(-4), W(3a,3b)*W(a,b)^(-9),
\\ W(4a,4b)*W(a,b)^(-16)]
\\
\\ Returns '0' on degenerate nets.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
```

```
twonettoeds(net,a,b) =

        local( returnvec, twonetab );

        returnvec = vector(4);

        \\ Return zero if the net is degenerate.
        if( istwonetdegen(net),
                if(debug, print("Degenerate nets taste bad!"); );
                return(0);
        );

        \\ Get the value of the net W(a,b)
        twonetab = twonetget(net,a,b);

        \\ Set up the eds
        returnvec[1] = 1;
        returnvec[2] = twonetget(net,2*a,2*b)*twonetab^(-4);
        returnvec[3] = twonetget(net,3*a,3*b)*twonetab^(-9);
        returnvec[4] = twonetget(net,4*a,4*b)*twonetab^(-16);

        \\ It's possible that the resulting eds is
        \\ degenerate, so notify if debug=1
        if( returnvec[2] == 0 || returnvec[3] == 0,
                if(debug, print("The resulting elliptic
                          divisibility sequence is degenerate."); );
        );

        return(returnvec);

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ edstotwonet(eds1,eds2)
\\
\\ Given two elliptic divisibility sequences associated to the same
\\ curve, returns a net (as a 4-vector) of the curve and both points.
\\
```

```
\\ If the two eds are not from the same curve, returns zero.
\\
\\ Elliptic divisibility sequences must be normalised (first term 1).
\\ If they are not normalised, they will be scaled.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

edstotwonet(eds1,eds2) =

        local(curvepoint1, curvepoint2, returnnet, coordchange,
                fullcurve1, fullcurve2);

        \\ get the curves and points of the sequences
        curvepoint1 = [edstocurve(eds1),[0,0]];
        curvepoint2 = [edstocurve(eds2),[0,0]];

        \\ check that the curves are the same up to unihomothetic
        \\ change of variables
        \\ (actually this checks up to u = plus/minus 1)
        fullcurve1 = edstocurvefull(eds1);
        fullcurve2 = edstocurvefull(eds2);
        if( fullcurve1[13] != fullcurve2[13] ||
                fullcurve1[12] != fullcurve2[12] ||
                fullcurve1[11] != fullcurve2[11] ||
                fullcurve1[10] != fullcurve2[10],
                if( debug,
                        print( "Elliptic divisibility sequences are
                                 not from the same curve." );
                );
                return(0);
        );

        \\ get coordinate change required to go from second curve
        \\ to first
        \\ since u = plus/minus 1, this should always be possible
        coordchange = getellcoordchange(curvepoint2[1],
                         curvepoint1[1]);
        if( coordchange == 0,
```

```
                if(debug,
                        print("Coordchange failed.");
                );
                return(0);
        );

        \\ change the second curve to go to the first
        curvepoint2 = [ ellchangecurve(curvepoint2[1],coordchange),
                 ellchangepoint(curvepoint2[2],coordchange) ];

        \\ get the net associated to the curve from eds1, and the
        \\ point from eds1 and the changed point from eds2
        returnnet = curvetotwonet(curvepoint1[1], curvepoint1[2],
                 curvepoint2[2]);

        return(returnnet);

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ getellcoordchange(curve1,curve2)
\\
\\ Given two elliptic curves of the same j-invariant, calculate the
\\ change of variables required to go from first to second, if
\\ possible.  The change of variables may lie over an extension field,
\\ in which case getellcoordchange may fail and you can try defining
\\ the same curves over the extension field and trying again.
\\
\\ This algorithm will return 0 if it fails.  It may fail for many
\\ reasons, the most common being that the curves are not isomorphic
\\ or the isomorphism lies over an extension.  It will also fail
\\ if the j-invariant is zero.
\\
\\ Requires curves to be isomorphic and to be of the same pari type.
\\ Mixed t_INT and t_FRAC is ok.
\\
\\ If you're having trouble with a curve of seemingly mixed type like
\\ [1,x,x^2,...], you can make 1 of type t_POL by using 1+0*x instead
```

```
\\ for example.  Pari doesn't square-root polys well, though, so you'll
\\ have better luck making it t_SER.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

getellcoordchange(curve1,curve2) =

        local(u,r,s,t, long, short, numer, denom, numerroot, denomroot,
                 uroot, u2, u4, i, k);
        long = vector(2);
        short = vector(2);
        uroot = matrix(2,2);


        \\ initialise curves for extended data
        \\ recall that this works for singular curves
        long[1] = elllong(curve1);
        long[2] = elllong(curve2);


        \\ also store a convenient short version of curves
        short[1] = vector(5);
        short[2] = vector(5);
        for(i=1,5,
                short[1][i] = curve1[i];
                short[2][i] = curve2[i];
        );


        \\ if the j-invariants are different there's no hope anyway
        if (long[1][13] != long[2][13],
                if(debug,
                        print("different j-invariants in");
                        print(" getellcoordchange");
                );
                return(0);
        );
```

```
\\ if the j-invariant is zero, this is an annoying and
\\ difficult case which is not yet implemented.
if ( long[1][13] == 0,
        if(debug,
                print("The j-invariant is zero.");
                print("  This case not implemented.");
        );
        return(0);
);


\\ in the case that only a unihomothetic change of variables
\\ is needed, this routine is easier (no issues of
\\ type/roots)
\\ this is the basic algorithm used below also
if( long[1][13] == long[2][13] &&
        long[1][12] == long[2][12] &&
        long[1][11] == long[2][11] &&
        long[1][10] == long[2][10],

        for(i=1,2,
                u = (-1)^i;
                s = 1/2*(curve2[1]*u - curve1[1]);
                r = 1/3*(curve2[2]*u^2 - curve1[2]
                        + s*curve1[1] + s^2);
                t = 1/2*(curve2[3]*u^3 - curve1[3]
                        - r*curve1[1]);

                if(debug, print("trying: ", [u,r,s,t] ); );
                if( ellchangecurve(short[1],[u,r,s,t])
                        == short[2],
                        return([u,r,s,t]);
                );
        );
        \\ if we reached this point, u was plus/minus 1, but
        \\ somehow neither of the changes of coords worked
        if(debug,
                print("coordchange failed for u=\pm 1");
        );
```

```
        return(0);

);

\\ If execution gets here, u was not just plus/minus 1


\\ There are issues with type comparison, so if it's mixed
\\ type, return error.
\\ t_INT and t_FRAC mixed are ok

\\ obtain first type
typebase = type(short[1][1]);

\\ consider integers as if they are fractions
if( typebase == "t_INT", typebase = "t_FRAC" );

\\ loop through other types, considering integers as
\\ as fractions, and watch for mismatch with first
for(i=1,5, for(k=1,2,
        typecheck = type(short[k][i]);
        if( typecheck == "t_INT", typecheck = "t_FRAC" );
        if( typecheck != typebase,
                if(debug, print("input curves must be
                         all same types"); );
                return(0);
        );
); );


\\ fourth power of u is the ration of the c4's
u4 = long[1][10]/long[2][10];


\\ make sure taking square root of u^4 is possible
if( !issquare(u4) && (type(u4) == "t_INTMOD"
        || type(u4) == "t_POLMOD"),

        \\ in this case u^4 is not a square
```

```
        \\ and the isomorphism of the curves
        \\ is defined over an extension field
        \\ try again in an extension field
        if(debug,
                print("fourth power of u isn't a square!!");
                print("u^4 = ", u4 );
                print("try an extension field.");
        );

        return(0);
);


\\ get a square root.
u2 = sqrt(u4);

\\ if it's a rational and square,
\\ make sure to get it as type rational
if( (type(u4) == "t_FRAC" || type(u4) == "t_INT" )
        && u4 > 0,

        \\ break it up as numerator and denominator and
        \\ take the integer roots of each
        numer = numerator(u4);
        denom = denominator(u4);
        numerroot = sqrtint(numer);
        denomroot = sqrtint(denom);

        \\ set u2 as the root, if this worked
        \\ if this didn't work, give a message
        \\ (didn't work means wasn't a rational
        \\ square, that's all)
        if( (numerroot/denomroot)^2 == u4,
                u2 = numerroot/denomroot;
        ,
                if(debug,
                        print("Square root as rational");
                        print(" didn't work in");
                        print(" getellcoordchange.");
```

```
                );
        );
);


\\ loop through both square roots of u4
for(i = 1,2,

        \\ do u2 and negative u2 in turn
        u2 = (-1) * u2;

        \\ check if it's a square, and if it is,
        \\ put the two roots in uroot
        \\ if it is not, just put uroot=1 as placeholder
        if( !issquare(u2) && (type(u2) == "t_INTMOD"
                || type(u2) == "t_POLMOD"),

                \\ u2 is not a square
                \\ and we are working with a modulus
                if(debug,
                        print("Quadratic non-residue");
                        print(" in getellcoordchange.");
                        print("  May need extension");
                        print(" field?");
                );
                uroot[1,i]=1;
                uroot[2,i]=1;
        ,

                \\ we are working in t_FRAC, t_COMPLEX etc.
                \\ so try to take a square root
                \\ depending on type this may produce
                \\ a pari error for weird types
                u = sqrt(u2);

                \\ if the type was rational (and positive)
                \\ try to do the root as a rational if
                \\ possible.
```

```
                \\ if this fails, it means the root is over
                \\ an extension of the rationals
                if( (type(u2) == "t_FRAC"
                        || type(u2) == "t_INT" ) && u2 > 0,

                        numer = numerator(u2);
                        denom = denominator(u2);
                        numerroot = sqrtint(numer);
                        denomroot = sqrtint(denom);
                        if( (numerroot/denomroot)^2 == u2,
                                u = numerroot/denomroot;
                        ,
                                if(debug,
                                    print("Square root");
                                    print(" as rational");
                                    print(" didn't work");
                                    print(" in getell");
                                    print("coordchange.");
                                );
                        );
                );

                \\ store the roots in uroot
                uroot[1,i] = u;
                uroot[2,i] = -u;
        );

);

\\ at this point, if all has gone well, we've stored four
\\ roots of u^4 in uroot and we can test them all
\\ as possibilities.

\\ if all has not gone well, some roots were missed
\\ and it is possible the change of variables
\\ requires working over an extension field.

for(i=1,2,for(k=1,2,
```

```
        \\ select the root for testing
        u = uroot[i,k];

        \\ setup change of variables
        s = 1/2*(curve2[1]*u - curve1[1]);
        r = 1/3*(curve2[2]*u^2 - curve1[2] + s*curve1[1]
                + s^2);
        t = 1/2*(curve2[3]*u^3 - curve1[3] - r*curve1[1]);

        \\ report if debug is on
        if(debug, print("trying: ", [u,r,s,t] ); );

        \\ if one of them works return it
        if( ellchangecurve(short[1],[u,r,s,t]) == short[2],
                return([u,r,s,t]);
        );
    ););

    \\ if not change of coordinates was found, report this
    \\ and return 0
    if(debug, print("No change of coordinates found."); );

    return(0);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ SECTION THREE:  TRANSFORMATIONS AND PROPERTIES OF NETS   \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
\\ istwonetdegen(net)
\\
\\ Returns 1 if net is degenerate, otherwise 0.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

istwonetdegen(net) =


        \\ check each of the possible degenerate cases
        if( net[3] == net[4],
                if(debug, print("The net has P-Q = 0"); );
                return(1);
        );
        if( net[2] == 0 && net[4] == 0,
                if(debug, print("The net has P=0"); );
                return(1);
        );
        if( net[1] == 0 && net[3] == 0,
                if(debug, print("The net has Q=0"); );
                return(1);
        );
        return(0);

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ istwonetsing(net)
\\
\\ Returns 1 if net is singular, otherwise 0.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

istwonetsing(net) =

        \\ check if net is singular by looking at discriminant
```

```
        \\ of associated curve
        if( elllong(twonettocurvechar2(net)[1])[12] == 0,
                return(1);
        ,
                return(0);
        );

}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonetbasischange(net,a,b,c,d)
\\
\\ Given a non-degenerate elliptic net 'net' associated to E,P,Q in the
\\ form of a vector of four terms, and integers a,b,c,d,
\\ returns a net associated to E and aP + bQ, cP + dQ
\\
\\ Does this directly via formulas.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

twonetbasischange(nett,a,b,c,d) =

        local(newnetter,nettab, nettcd, nettacbd, nettsing);

        \\ the new (post basis change) net will be stored here
        newnetter = vector(4);

        \\ store some useful values of the old net
        nettab = twonetget(nett,a,b);
        nettcd = twonetget(nett,c,d);
        nettacbd = twonetget(nett,a+c,b+d);
        nettsing = twonetget(nett,a+c,-b-d);

        \\ don't allow basis change that involves zero terms
        if( nettab == 0 || nettcd == 0 || nettacbd == 0
                || nettsing == 0,
                if(debug,
```

```
                        print("illegal basis change (or resulting
                                net is degenerate)");
                );
                return(0);
        );

        \\ compute terms of new net
        newnetter[1] = twonetget(nett,2*a,2*b)/nettab^4;
        newnetter[2] = twonetget(nett,2*c,2*d)/nettcd^4;
        newnetter[3] = twonetget(nett,2*a+c,2*b+d)/nettab^2
                        /nettacbd^2*nettcd;
        newnetter[4] = twonetget(nett,a+2*c,b+2*d)*nettab
                        /nettacbd^2/nettcd^2;

        \\ error trap
        if( istwonetdegen(newnetter),
                if(debug,
                        print("Watch out!  New net is degenerate.
                        (This should have been caught earlier.)");
                );
                print("ERROR, PLEASE REPORT THIS BUG 98273243611
                         to stange@math.brown.edu");
        );

        return(newnetter);


}


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonetbasischangeviacurve(net,a,b,c,d)
\\
\\ Given a non-degenerate elliptic net 'net' associated to E,P,Q in the
\\ form of a vector of four terms, and integers a,b,c,d,
\\ returns a net associated to E and aP + bQ, cP + dQ
\\
\\ Does this by translating to curve and then back to net.
\\
```

```
\\ This produces the same output as twonetbasischange, but is often
\\ faster since it doens't require computing all sorts of elements
\\ of elliptic nets.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

twonetbasischangeviacurve(nett,a,b,c,d) =

        local(newnetter, curve, point1, point2, curvepoint);

        \\ vector to store new (post basis change) net
        newnetter = vector(4);

        \\ get curve and point associated to the net
        curvepoint = twonettocurve(nett);
        curve = curvepoint[1];
        point1 = [curvepoint[2],0];
        point2 = [-curvepoint[2],0];

        \\ error trap:  points should be on the curve
        if( !ellisoncurve(curve,point1) || !ellisoncurve(curve,point2),
                if(debug,
                        print("points not on curve");
                );
                print("ERROR, PLEASE REPORT THIS BUG 944466112 to
                        stange@math.brown.edu");
                return(0);
        );

        \\ make a net from the curve and the new basis points
        newnetter = curvetotwonet(curve, elladd(curve,
                 ellpow(curve, point1, a), ellpow(curve, point2, b) ),
                 elladd(curve, ellpow(curve, point1, c),
                 ellpow(curve, point2, d) ) );

        \\ error trap:  making the new net failed
        \\ maybe it was degenerate
        if( newnetter == 0,
```

```
                if(debug, print("curvetotwonet failed."); );
                return(0);
        );

        \\ error trap:  curvetotwonet should not produce a degenerate
        \\ net
        if( istwonetdegen(newnetter),
                if(debug,
                        print("Watch out!  New net is degenerate.
                        (This should have been caught earlier.)");
                );
                print("ERROR, PLEASE REPORT THIS BUG 98276112 to
                        stange@math.brown.edu");
        );

        return(newnetter);


}



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ SECTION FOUR:  CALCULATING VALUES OF NETS               \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonetget(net,x,y)
\\
\\ Given a non-degenerate elliptic net 'net' in the
\\ form of a vector of four terms, and integers x and y
\\ returns the value of the net at index (x,y).
\\
\\ Uses a log n algorithm based on shipsey's thesis.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
{

twonetget(nett,x,y) =

        local(X,k,newnett, ar, cr);

        \\ This function is recursive, so this keeps track
        \\ of the recursion if debug is on.
        \\ debug = 1 in general with twonetget will produce
        \\ waaaaaayyy too much data.
        if(debug, print("Calling twonetget on W(", x, "," ,y, ")"); );

        \\ Degenerate nets are not allowed.  It is possible (but
        \\ rarely seen?) that this algorithm will recursively call
        \\ itself on a degenerate net somewhere in the process.
        \\ I would appreciate reports on this.
        if( istwonetdegen(nett),
                if(debug,
                        print("Degenerate nets taste like belly
                         button lint!");
                );
                \\ can't return 0 since that's a value!
                return("failed");
        );


        \\ sometimes there's REALLY no work to do
        \\ just return one of these simple values near the origin
        if( [x,y] == [2,0], return( nett[1] ); );
        if( [x,y] == [2,1], return( nett[3] ); );
        if( [x,y] == [1,1], return( 1 ); );
        if( [x,y] == [1,0], return( 1 ); );
        if( [x,y] == [0,0], return( 0 ); );
        if( [x,y] == [0,-2], return( -nett[2] ); );
        if( [x,y] == [-1,-2], return( -nett[4] ); );
        if( [x,y] == [-1,-1], return( -1 ); );
        if( [x,y] == [0,-1], return( -1 ); );
```

```
\\ if the answer is really small, just do it directly
\\ by calling twonetarray to build the small values
\\ this is more efficient than the recursive algorithm
\\ for small values
if( abs(x) < 5 && abs(y) < 5,
        return( twonetarray(nett,5)[6+x,6+y] );
);


\\ make sure x >= y
if( y > x,
        return( twonetget(
                [nett[2],nett[1],nett[4],nett[3]] ,y,x) );
);


\\ make sure x >= 0
if( x < 0,
        return( -twonetget( nett, -x, -y ) );
);


\\ if the requested coordinate is of the form
\\ (positive,negative), use a basis change to change
\\ it to (pos,pos)
if( y < 0,

        \\ change basis to P, -Q to do the calculation
        \\ with two positive values

        newnett = twonetbasischange(nett,1,0,0,-1);

        return( twonetget(newnett,x,-y)*(-1)^(x*y) );

);


\\ if y is 0 or 1, we'll need to get the x-axis of
```

```
\\ the net to start things off
if( y == 0 || y == 1,

        X = twonetarray(nett,5);
        k = 5+1;

);


\\ if y=0, just need EDS associated to first point
if( y == 0,

        return( edsget([X[k+1,k],X[k+2,k],
                        X[k+3,k],X[k+4,k]],x) );

);

\\ if y=1, just need to calculate out Shipsey block
\\ along the x-axis
\\ See my Tate Pairing via Elliptic Nets paper
\\ for calculating Shipsey blocks.
if( y == 1,

        return( shipsey_block([X[k-2,k],X[k-1,k],X[k,k],
                X[k+1,k],X[k+2,k],X[k+3,k],X[k+4,k],
                X[k+5,k];X[k,k+1],X[k+1,k+1],X[k+2,k+1],
                0,0,0,0,0], nett, x ) );


);

\\ at this point, y >= 2, and x >= y
if(debug, print("got to y >=2 case"); );

if( twonetget(nett,0,y) == 0,

        \\ if W(0,y) = 0, translate by it to get a
        \\ simpler thing to return
        if(debug,
```

```
        print("We have W(0,y) = 0 for y=", y);
);

if( twonetget(nett,0,2) != 0
        && twonetget(nett,2,0) != 0,
        \\ then no division by zero in following
        ar = (twonetget(nett,2,y)
                /twonetget(nett,2,0)
                /twonetget(nett,1,y));
        cr = (twonetget(nett,1,y+1)
                *twonetget(nett,0,2)
                *twonetget(nett,0,y+1)
                /twonetget(nett,0,y+2)/ar);
        return( twonetget(nett, x, 0)*ar^x*cr);

, \\ one of those is zero

        if( twonetget(nett,2,0) != 0,
                \\ W(0,2) is zero, so W(1,2) and
                \\ W(2,2) are not, then y is even
                ar = twonetget(nett,2,2)
                        /twonetget(nett,2,0)
                        /twonetget(nett,1,2);
                cr = twonetget(nett,1,2)/ar;
                return( twonetget(nett,x,0)
                        *ar^(y/2)*cr );

        ,
                \\ in this case W(2,0) = 0
                \\ so W(3,0) and W(3,1) not zero
                br = twonetget(nett,3,1)
                        /twonetget(nett,3,0);
                ar = -twonetget(nett,2,1)/br;
                cr = -ar;

                if( Mod(x,2) == 0,
                        \\ then W(x,y) = 0
                        return( 0 );
                ,
```

```
                                return( twonetget(nett,1,y)
                                        *ar^((x-1)/2)
                                        *br^(y*(x-1)/2)
                                        *cr^(((x-1)/2)^2) );
                        );

                );
        );

);

if( twonetget(nett,1,y) == 0,

        \\ if W(1,y) = 0, translate by it to get
        \\ a simpler thing to return
        if(debug,
                print("We have W(1,y) = 0 for y=", y);
        );

        if( twonetget(nett,0,2) != 0
                && twonetget(nett,2,0) != 0,
                \\ then no division by zero in
                \\ the following
                ar = (twonetget(nett,3,y)
                        /twonetget(nett,2,0)
                        /twonetget(nett,2,y));
                cr = (twonetget(nett,2,y+1)
                        *twonetget(nett,0,2)
                        *twonetget(nett,1,y+1)
                        /twonetget(nett,1,y+2)/ar);

                return( twonetget(nett, x-1, 0)
                        *ar^(x-1)*cr);

        ,
                \\ one of those is zero
                if( twonetget(nett,2,0) != 0,
                        \\ W(0,2) is zero, so W(1,2)
                        \\ and W(2,2) are not, y odd
```

```
                        ar = twonetget(nett,2,2)
                                /twonetget(nett,2,0)
                                /twonetget(nett,1,2);
                        cr = twonetget(nett,1,2)/ar;
                        return( twonetget(nett,x,1)
                                *ar^((y-1)/2)*cr );

                ,
                        \\ in this case W(2,0) = 0
                        \\ so W(3,0) and W(3,1) not 0
                        br = twonetget(nett,3,1)
                                /twonetget(nett,3,0);
                        ar = -twonetget(nett,2,1)/br;
                        cr = -ar;

                        if( Mod(x,2) == 0,
                                return( twonetget(nett,
                                        0,y)*ar^(x/2)
                                        *br^(y*x/2)
                                        *cr^((x/2)^2) );
                        ,  \\ then W(x,y) = 0
                                return( 0 );
                        );

                );
        );

);


if( twonetget(nett,-1,y) == 0,
        \\ if W(1,-y) = 0, translate by it to get a
        \\ simpler thing to return
        if(debug,
                print("We have W(-1,y) = 0 for y=", y);
        );

        if( twonetget(nett,0,2) != 0
                && twonetget(nett,2,0) != 0,
```

```
        \\ then no division by zero in following

        ar = (twonetget(nett,1,y)
                /twonetget(nett,2,0)
                /twonetget(nett,0,y));
        cr = (twonetget(nett,0,y+1)
                *twonetget(nett,0,2)
                *twonetget(nett,-1,y+1)
                /twonetget(nett,-1,y+2)/ar);
        return( twonetget(nett, x+1, 0)
                *ar^(x+1)*cr);

,
        \\ one of those is zero
        if( twonetget(nett,2,0) != 0,
                \\ W(0,2) is zero, so W(1,2) and
                \\ W(2,2) are not, y odd
                ar = twonetget(nett,2,2)
                        /twonetget(nett,2,0)
                        /twonetget(nett,1,2);
                cr = twonetget(nett,1,2)/ar;
                return( twonetget(nett,x,1)
                        *ar^((y-1)/2)*cr );

        ,
                \\ in this case W(2,0) = 0
                \\ so W(3,0) and W(3,1) not zero
                br = twonetget(nett,3,1)
                        /twonetget(nett,3,0);
                ar = -twonetget(nett,2,1)/br;
                cr = -ar;

                if( Mod(x,2) == 0,
                        return( twonetget(nett,
                                0,y)*ar^(x/2)
                                *br^(y*x/2)
                                *cr^((x/2)^2) );
                ,
                        \\ then W(x,y) = 0
```

```
                                        return( 0 );
                                );

                        );
                );

        );


        \\ otherwise change basis to P, yQ and use Shipsey block

        newnett = twonetbasischange(nett,1,0,0,y);
        if( newnett == 0,
                \\ we just had a failed basis change since
                \\ W(1,y) or W(0,y) = 0 or W(1,-y) = 0
                if(debug,
                        print("failed basis change that
                                 should not have happened");
                );
        );

        return( twonetget(newnett,x,1)*(twonetget(nett,1,y)^x)
                /twonetget(nett,0,y)^(x-1) );


}



{

shipsey_block( theblock, nett, val ) =
\\ given: theblock, the block centred on 1, in the net 'nett'
\\ cannot accept nett[1] = 0 or net such that
\\ twonetget(nett, 2, -1) = 0
\\ returns: the central value of block centred on val in the
\\ net 'nett' i.e. W(val,1)

        local(initial_data);
```

```
        initial_data = vector(4);

        if( debug, print("Calling Shipsey Block"); );

        if( nett[1] == 0 || twonetget(nett,2,-1),
                if(debug,
                        print("Shipsey Block Request with
                                zero W(2,0) or W(2,-1)");
                );
        );

        if( nett[1] == 0,
                \\ in case W(2,0) = 0
                initial_data[1] = "inf";
        ,
                initial_data[1] = 1/nett[1];
        );
        initial_data[2] = 1;
        \\ the following division should never be zero
        \\ in non-degenerage net
        initial_data[3] = 1/twonetget(nett,-1,1);

        if( twonetget(nett,2,-1) == 0,
                \\ in case W(2,-1) = 0, we cannot
                \\ call net_loop
                if(debug,
                        print("Avoiding a W(2,-1) issue");
                );
                return( -twonetget(nett,val+2,0)
                                *twonetget(nett,-3,1)
                                /(twonetget(nett,-1,1)
                                *twonetget(nett,2,0))^(val+3) );
        ,
                initial_data[4] = 1/twonetget(nett,2,-1);
        );

        return( net_loop( theblock, initial_data, val )[2,2] );

}
```

```
{

double_or_add(V, initial_data, add) =
\\ double_or_add
\\ Given a block V centred at k, and the initial data relevant to
\\ the elliptic net, returns either a block centred at 2k or 2k+1
\\ depending on whether variable "add" is 0 or 1 respectively.


        local(doubleV, m, i, j);


        \\ Create the output block
        doubleV = matrix(2,8);


        \\ initial_data contains the precomputed inverses
        inverse_20 = initial_data[1]; \\ inverse of W(2,0)
        inverse_11 = initial_data[2]; \\ inverse of W(1,1)
        inverse_n1 = initial_data[3]; \\ inverse of W(-1,1)
        inverse_2n = initial_data[4]; \\ inverse of W(2,-1)



        \\ Fill out first vector of output block

        for(j=-1,2,

                i = j;
                m = 4; \\ index to middle of block

                if( inverse_20 == "inf",
                        \\ in case it's a net where 2P = 0,
                        \\ even terms are all 0
                        doubleV[1,m+2*i-add] = 0;
                ,
                        doubleV[1,m + 2*i - add] = ((V[1,m+i])
```

```
                        *(V[1,m+i+2])*(V[1,m+i-1])^2
                        - (V[1,m+i])*(V[1,m+i-2])
                        *(V[1,m+i+1])^2)*inverse_20;
        );

        \\ when we hit j=-1, if add=1,
        \\ calculate W(2k+5,0) instead of W(2k-3,0)
        if( i == -1 && add == 1, i = 3; );

        doubleV[1,m + 2*i - 1 - add] = (V[1,m+i+1])
                *(V[1,m+i-1])^3 - (V[1,m+i-2])
                *(V[1,m+i])^3;
);


\\Fill out second vector of output block

m2 = 2; \\ index to middle of second vector of block
m1 = 4; \\ index to middle of first vector of block

if( add == 0,
        doubleV[2,1] = ( V[2,m2+1]*V[2,m2-1]
                *V[1,m1-1]^2 - V[1,m1]*V[1,m1-2]
                *V[2,m2]^2 )*inverse_11;
);

doubleV[2,2-add] = ( V[2,m2-1]*V[2,m2+1]*V[1,m1]^2
         - V[1,m1-1]*V[1,m1+1]*V[2,m2]^2);
doubleV[2,3-add] = ( V[2,m2+1]*V[2,m2-1]*V[1,m1+1]^2
        - V[1,m1]*V[1,m1+2]*V[2,m2]^2)*inverse_n1;

if( add == 1,
        if( inverse_2n == "inf",
                if(debug,
                        print("Oh dear, W(2,-1) = 0
                         in shipsey");
                );
                doubleV[2,3] = "inf";
        ,
```

```
                        doubleV[2,3] = ( V[1,m1+1]*V[1,m1+3]
                                *V[2,m2]^2 - V[2,m2-1]*V[2,m2+1]
                                *V[1,m1+2]^2 )*inverse_2n;
                );
        );


        return(doubleV);

}



{

net_loop(V,initial_data, m) =
\\ net_loop
\\ Given a starting block V centred at 1, initial data relevant to
\\ the elliptic net, and an integer m, returns the block centred at m

        local(currentV, m_size, add, i, j);


        \\ determine the number of steps in the double-and-add loop
        m_size = ceil(log(m+1)/log(2));


        \\ the variable storing the current block
        currentV = V;


        \\ ignore the first "1" in the binary expansion of m
        m = m - 2^(m_size-1);


        \\ step through the digits in the binary expansion of m
        for(j=1,m_size-1,
                i = m_size - j; \\kludgy version of "down to"
```

```
                \\ determine if this is a double step or a
                \\ double-and-add step
                \\ based on current digit of m; set "add" accordingly
                if( m - 2^(i-1) >= 0,
                        add = 1;
                        m = m - 2^(i-1);
                ,
                        add = 0;
                );


                \\ call the double or double-and-add function to
                \\ update the current block
                currentV = double_or_add(currentV, initial_data, add);


                \\ print information about the current step
                if( debug,
                        print("The digit of $m$ for this step is "
                                add, ".");
                        \\print("The resulting block from this
                        \\       step is ", currentV);
                );

        );


        return(currentV);

}



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonetgetslow(eds,a,b)
\\
\\ Given a non-degenerate elliptic net 'net' in the
\\ form of a vector of four terms, and integers a and b
\\ returns the value of the net at index (a,b).
```

```
\\
\\ Horribly inefficient; generates the whole array to large width.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

twonetgetslow(nety,x,y) =

        local(k,width);

        if( istwonetdegen(nety),
                if(debug,
                        print("Degenerate nets taste like toe jam!");
                );
                return("failed");
        );

        width = 2*max(abs(x),abs(y))+8;
        k = width + 1;

        return(twonetarray(nety,width)[k+x,k+y]);

}



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonetarray(net,width)
\\
\\ Given a non-degenerate elliptic net 'net' in the
\\ form of a vector of four terms, and integer width > 3,
\\ returns a large matrix (2*width+1 times 2*width+1)
\\ whose entries represent the entries of the elliptic net
\\
\\ If the returned matrix is called X, then
\\ W(x,y) is stored in entry X[width+1+x,width+1+y].
\\ In particular, the array contains all W(x,y) with |x|,|y| <= width.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
{

twonetarray(startnet,width) =

        local( i,j,k,l,w,X,m );

        if(debug, print("Calling twonetarray"); );

        if( istwonetdegen(startnet),
                if(debug,
                        print("Degenerate nets taste like earwax!");
                );
                return(0);
        );

        \\ check to be sure width is an integer
        if( type(width) != "t_INT",
                if(debug, print("Width not an integer!"););
                return(0);
        );

        \\ check to be sure width is not too small
        if( width < 4,
                print("width too small, increasing to width=4");
                width = 4;
        );

        \\ set up the matrix
        w = 2*width+1;
        X = matrix(w,w);

        \\ Useful variable 'k' so that W(x,y) = X[k+x,k+y]
        k = width+1;

        \\ Error-catching background filler for matrix
        \\ (if in the end you see 'c' there's been an error,
        \\ as this should all be overwritten)
        for(l=1,w,for(m=1,w,
                X[l,m] = c;
```

```
););


\\ Initial values of the net.
X[k+0,k+0] = 0;
X[k+0,k+1] = 1;
X[k+1,k+0] = 1;
X[k+1,k+1] = 1;
X[k+2,k+0] = startnet[1];
X[k+2,k+1] = startnet[3];
X[k+0,k+2] = startnet[2];
X[k+1,k+2] = startnet[4];



\\ Starter recurrences fill out the area near the origin.
\\ No division by zero possible here if net is nondegenerate.

X[k+1,k-1] = (-X[k+2,k+1]*X[k,k+1]^3 + X[k+1,k+2]*X[k+1,k]^3)
        /(X[k+1,k+1]^3);

if(X[k+1,k-1] == 0,
        if(debug,print("Oops, P+Q=0"););
        return(0);
);

X[k+2,k-1] = (X[k+2,k]*X[k+1,k]^2*X[k+0,k+2] - X[k+2,k+1]
        *X[k+1,k-1]^2*X[k,k+1])/(X[k,k+1]*X[k+1,k+1]^2);

X[k+1,k-2] = (-X[k+2,k-1]*X[k,k+1]^3 + X[k+1,k+1]*X[k+1,k-1]^3)
        /(X[k+1,k]^3);

X[k+3,k] = ( -X[k+1,k+1]*X[k+1,k-1]*X[k+2,k]^2 + X[k+2,k+1]
        *X[k+2,k-1]*X[k+1,k]^2) / (X[k+1,k]*X[k,k+1]^2);

X[k,k+3] = ( X[k+1,k+1]*X[k+1,k-1]*X[k,k+2]^2 - X[k+1,k+2]
        *X[k+1,k-2]*X[k,k+1]^2) / ( X[k,k+1]*X[k+1,k]^2);

X[k+2,k+2] = ( - X[k+2,k]*X[k+1,k+1]*X[k+1,k+2]*X[k,k+1]
```

```
        + X[k,k+2]*X[k+2,k+1]*X[k+1,k]*X[k+1,k+1])/(X[k,k+1]
        *X[k+1,k]*X[k+1,k-1]);

X[k+2,k-2] = ( X[k+2,k]*X[k,k+1]*X[k+1,k-2]*X[k+1,k-1]
        - X[k+2,k-1]*X[k,k+2]*X[k+1,k-1]*X[k+1,k])
        /(-X[k,k+1]*X[k+1,k]*X[k+1,k+1]);

X[k+3,k+1] = ( - X[k+1,k+1]^2*X[k+2,k-1]*X[k+2,k+1]
        + X[k+2,k]^2*X[k+1,k]*X[k+1,k+2])/(-X[k+1,k-1]
        *X[k,k+1]);

X[k+1,k+3] = ( X[k+1,k+1]^2*X[k+1,k-2]*X[k+1,k+2] + X[k,k+2]
        *X[k,k+2]*X[k,k+1]*X[k+2,k+1])/(X[k+1,k-1]*X[k+1,k]^2);

X[k+2,k+3] = ( X[k+1,k]*X[k+1,k+2]*X[k,k+2]*X[k+2,k+2]
        - X[k+1,k+3]*X[k+1,k+1]*X[k+2,k+1]*X[k,k+1] )
        / ( X[k,k+1]*X[k+1,k-1]*X[k+1,k+1] );

X[k+3,k+2] = ( X[k,k+1]*X[k+2,k+1]*X[k+2,k]*X[k+2,k+2]
        - X[k+3,k+1]*X[k+1,k+1]*X[k+1,k+2]*X[k+1,k] )
        / ( -X[k+1,k]*X[k+1,k-1]*X[k+1,k+1] );

X[k+3,k+3] = ( - X[k+3,k+2]*X[k+1,k+1]*X[k+1,k+2]*X[k+1,k+1]
        + X[k+2,k+2]*X[k,k+1]*X[k+2,k+2]*X[k+2,k+1])
        /(-X[k+1,k]^2*X[k+1,k+1]);

X[k+3,k-1] = ( X[k+2,k]^3*X[k,k+2] + X[k+1,k-1]^3*X[k+3,k+1] )
        / (X[k+1,k+1]^3);

X[k+1,k-3] = -( X[k,k+2]^3*X[k+2,k] - X[k+1,k-1]^3*X[k+1,k+3] )
        / (X[k+1,k+1]^3);

if( X[k+2,k] != 0,

        \\ in this case 2P != 0 on the curve

        X[k,k+4] = ( -X[k+1,k+3]*X[k+1,k+1]*X[k,k+2]*X[k+2,k-2]
                -X[k+1,k-1]*X[k+1,k-3]*X[k,k+2]*X[k+2,k+2] )
                / (-X[k+2,k]*X[k+1,k-1]*X[k+1,k+1] );
```

```
,
        \\ now 2P = 0.  If also 2P - Q = 0,
        \\ then Q = 0 -- a degen. net.  So
        \\ division by X[k+2,k-1] ok.

        if( X[k+2,k-1] == 0,
                if(debug, print("error, Q=0"); );
                return(0);
        );

        X[k,k+4] = ( -X[k+2,k-2]*X[k+1,k+2]*X[k,k+3]*X[k+1,k+1]
                - X[k+2,k+2]*X[k+1,k-2]*X[k,k+1]*X[k+1,k-3] )
                / ( -X[k+1,k]*X[k+1,k+1]*X[k+2,k-1] );

);

if( X[k,k+2] != 0,
        \\ in this case 2Q != 0

        X[k+4,k] = ( X[k+3,k+1]*X[k+1,k+1]*X[k+2,k]*X[k+2,k-2]
                - X[k+1,k-1]*X[k+3,k-1]*X[k+2,k]*X[k+2,k+2] )
                / (X[k,k+2]*X[k+1,k-1]*X[k+1,k+1] );

,
        \\ now 2Q = 0.  If also P - 2Q = 0, then
        \\ P = 0 -- a degen. net.  So division by X[k+1,k-2] ok.

        if( X[k+1,k-2] == 0,
                if(debug, print("error, P=0"); );
                return(0);
        );
        X[k+4,k] = ( X[k+2,k-2]*X[k+2,k+1]*X[k+3,k]*X[k+1,k+1]
                - X[k+2,k+2]*X[k+2,k-1]*X[k+1,k]*X[k+3,k-1] )
                / ( X[k,k+1]*X[k+1,k+1]*X[k+1,k-2] );

);

X[k+4,k-1] = ( -X[k+3,k]*X[k+1,k-2]*X[k+2,k]^2 + X[k+1,k-1]
```

```
        *X[k+3,k+1]*X[k+2,k-1]^2 ) / (X[k,k+1]*X[k+1,k+1]^2);
X[k+1,k-4] = ( -X[k,k+3]*X[k+2,k-1]*X[k,k+2]^2 + X[k+1,k-1]
        *X[k+1,k+3]*X[k+1,k-2]^2 ) / (X[k+1,k]*X[k+1,k+1]^2);

\\if(debug, print("done starter recurrences"); );


\\ Fill out axes in positive direction with EDS recurrence
\\ The terms W(0,i) and W(i,0) for i=1,4 are already done
\\ by starter recurrences

for(i=5,width,

        \\ y-axis

        if( X[k,k+i-4] != 0,

                \\ safe to divide by W(0,i-4)
                X[k,k+i]=(X[k,k+i-1]*X[k,k+i-3]*X[k,k+2]^2
                        -X[k,k+3]*X[k,k+1]*X[k,k+i-2]^2)
                        /(X[k,k+i-4]*X[k,k+1]^2);
        ,
                \\ if W(0,i-4) = 0 then W(0,i-5) != 0, so safe
                \\ to divide by W(0,i-5)

                \\ further, if W(0,2) = 0 = W(0,i-4), then net
                \\ nondegen => i is even, so W(0,i)=0 too

                if(X[k,k+2] == 0,
                        if(debug && Mod(i,2) != 0,
                                print("Degenerate net error
                                        W(0,odd)=W(0,2)=0");
                        );
                        X[k,k+i] = 0;

                ,

                        if(i==5,
                                if(debug,
```

```
                                print("W(0,1)=0 error");
                        );
                );

                X[k,k+i]=(X[k,k+i-1]*X[k,k+i-4]*X[k,k+2]
                        *X[k,k+3]-X[k,k+4]*X[k,k+1]
                        *X[k,k+i-2]*X[k,k+i-3])
                        /(X[k,k+i-5]
                        *X[k,k+1]*X[k,k+2]);
        );
);

\\ x-axis
if( X[k+i-4,k] != 0,
        \\ safe to divide by W(i-4,0)
        X[k+i,k]=(X[k+i-1,k]*X[k+i-3,k]*X[k+2,k]^2
                -X[k+3,k]*X[k+1,k]*X[k+i-2,k]^2)
                /(X[k+i-4,k]*X[k+1,k]^2);
,
        \\ if W(i-4,0) = 0 then W(i-5,0) != 0,
        \\ so safe to divide by W(i-5,0)

        \\ further, if W(2,0) = 0 = W(i-4,0),
        \\ then net nondegen => i is even, so W(i,0)=0
        \\ too
        if(X[k+2,k] == 0,
                if(debug && Mod(i,2) != 0,
                        print("Degenerate net error
                                W(odd,0)=W(2,0)=0");
                );
                X[k+i,k] = 0;

        ,

                if(i==5,
                        if(debug,
                                print("W(1,0)=0 error");
                        );
                );
```

```
                        X[k+i,k]=(X[k+i-1,k]*X[k+i-4,k]
                                *X[k+2,k]*X[k+3,k]
                                -X[k+4,k]*X[k+1,k]
                                *X[k+i-2,k]*X[k+i-3,k])
                                /(X[k+i-5,k]
                                *X[k+1,k]*X[k+2,k]);
                );
        );

);


\\if(debug, print("done axes"); );

\\Fill out the terms 0-4 of W(-1,*).
for(j=0,4,
        X[k-1,k+j] = -X[k+1,k-j];
);


\\Using translated sequence recurrences, fill out positive
\\ rows and columns (full first quadrant)

for(i=1,4,
        X = dnetgen_helper_row(X,width,i,-1);
        X = dnetgen_helper_col(X,width,i,-1);
);

for(i=5,width,
        X = dnetgen_helper_row(X,width,i,0);
        X = dnetgen_helper_col(X,width,i,0);
);

\\if(debug, print("done translated recs positive"); );

\\ Fill out columns in negative direction

for(j=1,width,
for(m=1,width,
```

```
i = -m;

        \\ error-check
        if( debug && X[k+j,k+i+4] == 0 && X[k+j,k+i+5] == 0,
                print("Yikes degenerate: W(", j, ",", i+4,
                        ")=0=W(", j, ",", i+5, ")" );
        );

        \\ fill out column j downwards
        if( X[k+j,k+i+4] != 0,
                \\ okay to divide by W(j,i+4)
                X[k+j,k+i]=(X[k+j,k+i+1]*X[k+j,k+i+3]
                        *X[k,k+2]^2-X[k,k+3]*X[k,k+1]
                        *X[k+j,k+i+2]^2)
                        /(X[k+j,k+i+4]*X[k,k+1]^2);
        ,
                \\ W(j,i+4)=0 so okay to divide by W(j,i+5)

                \\ further, if W(0,2) = 0 = W(j,i+4), then net
                \\ nondegen => i is even, so W(j,i)=0 too

                if(X[k,k+2] == 0,
                        X[k+j,k+i] = 0;
                ,
                        X[k+j,k+i]=(X[k+j,k+i+1]*X[k+j,k+i+4]
                                *X[k,k+2]*X[k,k+3]-X[k,k+4]
                                *X[k,k+1]*X[k+j,k+i+2]
                                *X[k+j,k+i+3])
                                /(X[k+j,k+i+5]*X[k,k+1]
                                *X[k,k+2]);
                );
        );

);
);


\\Fill out the negatives.
for(i=-width,-1,for(j=-width,width,
```

```
                X[k+i,k+j]=-X[k-i,k-j];
        ););

        for(i=-width,-1,
                X[k,k+i]=-X[k,k-i];
        );

        if(debug, print("returning twonetarray"); );

        return(X);

}


{
dnetgen_helper_row(X, width, row, startblock)=
\\ internal function that fills out a row in the
\\ positive direction of a 2d array

        local(j,i);

        j=row;
        for(i=startblock+5,width,

                if( X[k+i-4,k+j] != 0,
                        \\ okay to divide by W(i-4,j)
                        X[k+i,k+j]=(X[k+i-1,k+j]*X[k+i-3,k+j]
                                *X[k+2,k]^2-X[k+3,k]*X[k+1,k]
                                *X[k+i-2,k+j]^2)/(X[k+i-4,k+j]
                                *X[k+1,k]^2);

                ,

                        \\ W(i-4,j) = 0 so okay to divide by W(i-3,j)

                        \\ further, if W(2,0) = 0 = W(i-4,j), then
                        \\ net nondegen => i is even, so W(i,j)=0 too
                        if(X[k+2,k] == 0,
                                X[k+i,k+j] = 0;
```

```
                        ,
                                X[k+i,k+j]=(X[k+i-1,k+j]
                                        *X[k+i-4,k+j]*X[k+2,k]
                                        *X[k+3,k]-X[k+4,k]*X[k+1,k]
                                        *X[k+i-2,k+j]*X[k+i-3,k+j])
                                        /(X[k+i-5,k+j]*X[k+1,k]
                                        *X[k+2,k]);
                        );
                );
        );

        return(X);

}


{
dnetgen_helper_col(X, width, col, startblock)=
\\ internal function that fills out a column in the positive
\\ direction of a 2d array

        local(j,i);

        j=col;
        for(i=startblock+5,width,
                if( X[k+j,k+i-4] != 0,
                        \\ okay to divide by W(j,i-4)
                        X[k+j,k+i]=(X[k+j,k+i-1]*X[k+j,k+i-3]
                                *X[k,k+2]^2-X[k,k+3]*X[k,k+1]
                                *X[k+j,k+i-2]^2)/(X[k+j,k+i-4]
                                *X[k,k+1]^2);

                ,

                        \\ W(j,i-4) = 0 so okay to divide by W(j,i-3)

                        \\ further, if W(0,2) = 0 = W(j,i-4), then
                        \\ net nondegen => i is even, so W(j,i)=0 too
                        if(X[k,k+2] == 0,
```

```
                                X[k+j,k+i] = 0;
                        ,
                                X[k+j,k+i]=(X[k+j,k+i-1]*X[k+j,k+i-4]
                                        *X[k,k+2]*X[k,k+3]-X[k,k+4]
                                        *X[k,k+1]*X[k+j,k+i-2]
                                        *X[k+j,k+i-3])/(X[k+j,k+i-5]
                                        *X[k,k+1]*X[k,k+2]);
                        );
                );
        );

        return(X);

}



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ twonetarrayprettify(array,typeofpretty)
\\
\\ Given an array produced by twonetarray, prettifies it depending
\\ on the value of typeofpretty:
\\
\\ cartesian:  makes x and y increase right and up from center
\\ posquad: shows positive quadrant only, cartesian style
\\ latex: outputs it in latex format
\\ maple: outputs it as matrix for maple
\\ mathematica:  outputs it as matrix for mathematica
\\
\\ For best results, write latex'd, maple'd or mathematica'd matrices
\\ to file, where the escape characters don't show (copy and paste
\\ from screen causes problems).  Use write command.
\\
\\ In general, these apply to any square matrix, and one should apply
\\ them, in the order desired, one at a time (for example, to make a
\\ nice latex table, do cartesian first, then latex.
\\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
{

twonetarrayprettify(array,typeofpretty)=

        local(newmat, newsize);

        \\ all cases except posquad use newsize
        newsize = length(array);

        if( typeofpretty == "cartesian",
                newmat = array;
                array = mattranspose(array);
                for(i=1,newsize,for(j=1,newsize,
                        newmat[i,j] = array[newsize+1-i,j];
                ););
                return(newmat);

        );
        if( typeofpretty == "posquad",
                newsize = (length(array)-1)/2;
                newmat = matrix(newsize, newsize);
                for(i=1,newsize,for(j=1,newsize,
                        newmat[i,j] = array[newsize+i,newsize+j]
                ););
                newmat = twonetarrayprettify(newmat,"cartesian");
                return(newmat);
        );
        if( typeofpretty == "latex",
                newmat = "";
                newmat = concat(newmat,"\\begin{matrix}");
                for(i=1,newsize-1,

                        newmat = concat(newmat, array[i,1]);
                        for(j=2,newsize,
                                newmat = concat(newmat," & ");
                                newmat = concat(newmat,array[i,j]);
                        );
                        newmat = concat(newmat," \\\\");
```

```
        );

        newmat = concat(newmat, array[newsize,1]);
        for(j=2,newsize,
                newmat = concat(newmat," & ");
                newmat = concat(newmat,array[newsize,j]);
        );

        newmat = concat(newmat,"\\end{matrix}");

        return(newmat);

);
if( typeofpretty == "maple",
        newmat = "";
        newmat = concat(newmat,"Matrix(");
        newmat = concat(newmat,newsize);
        newmat = concat(newmat,",");
        newmat = concat(newmat,newsize);
        newmat = concat(newmat,",[[");
        for(i=1,newsize-1,

                newmat = concat(newmat, array[i,1]);
                for(j=2,newsize,
                        newmat = concat(newmat,",");
                        newmat = concat(newmat,array[i,j]);
                );
                newmat = concat(newmat,"],[");

        );

        newmat = concat(newmat, array[newsize,1]);
        for(j=2,newsize,
                newmat = concat(newmat,",");
                newmat = concat(newmat,array[newsize,j]);
        );

        newmat = concat(newmat,"]])");
```

```
                return(newmat);


        );
        if( typeofpretty == "mathematica",
                newmat = "";
                newmat = concat(newmat,"{{");
                for(i=1,newsize-1,

                        newmat = concat(newmat, array[i,1]);
                        for(j=2,newsize,
                                newmat = concat(newmat,",");
                                newmat = concat(newmat,array[i,j]);
                        );
                        newmat = concat(newmat,"},{");

                );

                newmat = concat(newmat, array[newsize,1]);
                for(j=2,newsize,
                        newmat = concat(newmat,",");
                        newmat = concat(newmat,array[newsize,j]);
                );

                newmat = concat(newmat,"}}");

                return(newmat);

        );

        if( debug,
                print("not a valid type of prettification");
        );
        return(0);

}
```

B.3 Computation of the Tate-Lichtenbaum pairing

```
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ PARI/GP Script for Computation of Tate Pairing     v. 1.1          \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ This script uses the algorithm presented in                        \\
\\ "The Tate Pairing via Elliptic Nets" by Katherine E. Stange        \\
\\ See http://www.math.brown.edu/~stange/tatepairing/                 \\
\\ or contact <stange at math dot brown dot edu> for info             \\
\\                                                                    \\
\\ Note that the optimisations mentioned in the paper are not         \\
\\ all implemented here; rather, the algorithm is implemented in      \\
\\ its simplest form for clarity.                                     \\
\\                                                                    \\
\\ v. 1.1 fixes a bug that affects pairings on points whose order is  \\
\\ a power of two.                                                    \\
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

\\ Set debug = 1 to report steps of algorithm
global(debug);
debug = 0;




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ double_or_add
\\ Given a block V centred at k, and the initial data relevant to
\\ the elliptic net, returns either a block centred at 2k or 2k+1
\\ depending on whether variable "add" is 0 or 1 respectively.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

double_or_add(V, initial_data, add) =

        local(doubleV, m, i, j);
```

```
\\ Create the output block
doubleV = matrix(2,8);


\\ initial_data contains the precomputed inverses
inverse_20 = initial_data[1]; \\ inverse of W(2,0)
inverse_11 = initial_data[2]; \\ inverse of W(1,1)
inverse_n1 = initial_data[3]; \\ inverse of W(-1,1)
inverse_2n = initial_data[4]; \\ inverse of W(2,-1)



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ Fill out first vector of output block
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

for(j=-1,2,

        i = j;
        m = 4; \\ index to middle of block

        doubleV[1,m + 2*i - add] = ((V[1,m+i])*(V[1,m+i+2])
                        *(V[1,m+i-1])^2 - (V[1,m+i])
                        *(V[1,m+i-2])*(V[1,m+i+1])^2)
                        *inverse_20;

        \\ when we hit j=-1, if add=1, calculate
        \\ W(2k+5,0) instead of W(2k-3,0)
        if( i == -1 && add == 1, i = 3; );

        doubleV[1,m + 2*i - 1 - add] = (V[1,m+i+1])
                        *(V[1,m+i-1])^3 - (V[1,m+i-2])
                        *(V[1,m+i])^3;
);


\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
```

```
        \\Fill out second vector of output block
        \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

        m2 = 2; \\ index to middle of second vector of block
        m1 = 4; \\ index to middle of first vector of block

        if( add == 0,
                doubleV[2,1] = ( V[2,m2+1]*V[2,m2-1]*V[1,m1-1]^2
                        - V[1,m1]*V[1,m1-2]*V[2,m2]^2 )*inverse_11;
        );

        doubleV[2,2-add] = ( V[2,m2-1]*V[2,m2+1]*V[1,m1]^2
                        - V[1,m1-1]*V[1,m1+1]*V[2,m2]^2);
        doubleV[2,3-add] = ( V[2,m2+1]*V[2,m2-1]*V[1,m1+1]^2
                        - V[1,m1]*V[1,m1+2]*V[2,m2]^2)*inverse_n1;

        if( add == 1,
                doubleV[2,3] = ( V[1,m1+1]*V[1,m1+3]*V[2,m2]^2
                        - V[2,m2-1]*V[2,m2+1]*V[1,m1+2]^2 )*inverse_2n;
        );

        return(doubleV);

}




\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ net_loop
\\ Given a starting block V centred at 1, initial data relevant to
\\ the elliptic net, and an integer m, returns the block centred at m
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{

net_loop(V,initial_data, m) =

        local(currentV, m_size, add, i, j);
```

```
\\ determine the number of steps in the double-and-add loop
m_size = ceil(log(m+1)/log(2));


\\ the variable storing the current block
currentV = V;


\\ ignore the first "1" in the binary expansion of m
m = m - 2^(m_size-1);


\\ step through the digits in the binary expansion of m
for(j=1,m_size-1,
        i = m_size - j; \\kludgy version of "down to"

        \\ determine if this is a double step or a
        \\ double-and-add step based on current digit
        \\ of m; set "add" accordingly
        if( m - 2^(i-1) >= 0,
                add = 1;
                m = m - 2^(i-1);
        ,
                add = 0;
        );


        \\ call the double or double-and-add function to
        \\ update the current block
        currentV = double_or_add(currentV, initial_data, add);


        \\ print information about the current step
        if( debug == 1,
                print("The digit of $m$ for this step is ",
                        add, ".");
                print("The resulting block from this step is ",
```

```
                                currentV);
                );

        );


        return(currentV);

}



\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
\\ tate_pairing_alg
\\ Arguments: elliptic curve, two points, and integer.
\\ Returns: tate pairing of the two points with respect to the integer.
\\ Note: this is currently not implemented for curves in characteristic
\\ 2 or 3.
\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\

{
tate_pairing_alg(elliptic_curve, point_a, point_b, torsion) =

        local(V, initial_data, x1, y1, x2, y2, a4, a6, resultV);


        \\ Create a starting block for the net
        V = matrix(2,8);


        \\ Create a vector to store the pre-computed inverses
        initial_data = vector(4);


        \\ Make sure the points are on the curve
        if( ellisoncurve(elliptic_curve, point_a) == 0,
                print("The first point is not on the curve!");
        );
        if( ellisoncurve(elliptic_curve, point_b) == 0,
```

```
        print("The second point is not on the curve!");
);


\\ If the curve is not in nice Weierstrass form
\\ y^2 = x^3 + Ax + B, do a change of coordinates
if( elliptic_curve[1] != 0 || elliptic_curve[2] != 0
        || elliptic_curve[3] != 0,
        print("Curve not in two-coeff. Weierstrass form!");
        a1 = elliptic_curve[1];
        a2 = elliptic_curve[3];
        a3 = elliptic_curve[2];
        a4 = elliptic_curve[4];
        a6 = elliptic_curve[5];

        coordinate_change = [1/6,-a2/3,-a1/2,-a3/2+a1*a2/6];
        elliptic_curve = ellchangecurve(elliptic_curve,
                                coordinate_change);
        point_a = ellchangepoint(point_a, coordinate_change);
        point_b = ellchangepoint(point_b, coordinate_change);

        print("New curve: ", elliptic_curve);
        print("New first point: ", point_a);
        print("New second point: ", point_b);

);


\\ Set up the usual variable names for elliptic curves
x1 = point_a[1];
y1 = point_a[2];
x2 = point_b[1];
y2 = point_b[2];
a4 = elliptic_curve[4];
a6 = elliptic_curve[5];


\\ Fill out the first vector of the block
V[1,4] = 1;
```

```
V[1,5] = 2*y1;
V[1,6] = 3*x1^4 + 6*a4*x1^2 + 12*a6*x1-a4^2;
V[1,7] = 4*y1*(x1^6 + 5*a4*x1^4 + 20*a6*x1^3 - 5*a4^2*x1^2
                - 4*a4*a6*x1 - 8*a6^2 - a4^3);
V[1,8] = -((V[1,6])^3 - (V[1,5])^3*(V[1,7]));
V[1,3] = 0;
V[1,2] = - V[1,4];
V[1,1] = - V[1,5];


\\ Fill out the second vector of the block
V[2,1] = 1;
V[2,2] = 1;
V[2,3] = 2*x1+x2 - ((y2-y1)/(x2-x1))^2;


\\ Print starting block
if( debug == 1,
        print("The beginning block is: ", V);
);


\\ Pre-compute the inverses
initial_data[1] = 1/(V[1,5]);
initial_data[2] = 1;
initial_data[3] = 1/(x1 - x2);
initial_data[4] = 1/((y1+y2)^2 - (2*x1 + x2)*(x1-x2)^2);

\\ Print the pre-computed info
if( debug == 1,
        print("Precomputed inverses: ", initial_data);
);


\\ Call the net algorithm to obtain the block centred
\\ at "torsion"
resultV = net_loop(V, initial_data, torsion);
```

```
        \\ Apply the Tate pairing formula and return result
        return(resultV[2,3]/resultV[1,5]);

}
```

Bibliography

[1] *Groupes de monodromie en géométrie algébrique. I.* Springer-Verlag, Berlin, 1972. Séminaire de Géométrie Algébrique du Bois-Marie 1967–1969 (SGA 7 I), Dirigé par A. Grothendieck. Avec la collaboration de M. Raynaud et D. S. Rim, Lecture Notes in Mathematics, Vol. 288.

[2] Mohamed Ayad. Périodicité (mod q) des suites elliptiques et points S-entiers sur les courbes elliptiques. *Ann. Inst. Fourier (Grenoble)*, 43(3):585–618, 1993.

[3] Paulo S. L. M. Barreto. The pairing-based crypto lounge. `http://planeta.terra.com.br/informatica/paulbarreto/pblounge.html`.

[4] Paulo S. L. M. Barreto, Hae Y. Kim, Ben Lynn, and Michael Scott. Efficient algorithms for pairing-based cryptosystems. In *Advances in cryptology—CRYPTO 2002*, volume 2442 of *Lecture Notes in Comput. Sci.*, pages 354–368. Springer, Berlin, 2002.

[5] Paulo S. L. M. Barreto, Ben Lynn, and Michael Scott. On the selection of pairing-friendly groups. In *Selected areas in cryptography*, volume 3006 of *Lecture Notes in Comput. Sci.*, pages 17–25. Springer, Berlin, 2004.

[6] Ian F. Blake, Gadiel Seroussi, and Nigel P. Smart. *Elliptic curves in cryptography*, volume 265 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, Cambridge, 2000. Reprint of the 1999 original.

[7] Dan Boneh and Matt Franklin. Identity-based encryption from the Weil pairing. In *Advances in cryptology—CRYPTO 2001 (Santa Barbara, CA)*, volume 2139 of *Lecture Notes in Comput. Sci.*, pages 213–229. Springer, Berlin, 2001.

[8] Lawrence Breen. *Fonctions thêta et théorème du cube*, volume 980 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1983.

[9] Jean-Luc Brylinski and Pierre Deligne. Central extensions of reductive groups by $\mathbf{K}_2$. *Publ. Math. Inst. Hautes Études Sci.*, (94):5–85, 2001.

[10] K. Chandrasekharan. *Elliptic functions*, volume 281 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin, 1985.

[11] Mathieu Ciet, Marc Joye, Kristin Lauter, and Peter L. Montgomery. Trading inversions for multiplications in elliptic curve cryptography. *Des. Codes Cryptogr.*, 39(2):189–206, 2006.

[12] G. Cornelissen and K. Zahidi. Elliptic divisibility sequences and undecidable problems about rational points. `http://arxiv.org/abs/math/0412473v3`, 2004.

[13] Richard Crandall and Carl Pomerance. *Prime numbers*. Springer-Verlag, New York, 2001. A computational perspective.

[14] Isabelle Déchène. Arithmetic of generalized Jacobians. In *Algorithmic number theory*, volume 4076 of *Lecture Notes in Comput. Sci.*, pages 421–435. Springer, Berlin, 2006.

[15] David S. Dummit and Richard M. Foote. *Abstract algebra*. John Wiley & Sons Inc., Hoboken, NJ, third edition, 2004.

[16] Sylvain Duquesne and Gerhard Frey. Background on pairings. In *Handbook of elliptic and hyperelliptic curve cryptography*, Discrete Math. Appl. (Boca Raton), pages 115–124. Chapman & Hall/CRC, Boca Raton, FL, 2006.

[17] Sylvain Duquesne and Gerhard Frey. Implementation of pairings. In *Handbook of elliptic and hyperelliptic curve cryptography*, Discrete Math. Appl. (Boca Raton), pages 389–404. Chapman & Hall/CRC, Boca Raton, FL, 2006.

[18] Sylvain Duquesne and Tanja Lange. Pairing-based cryptography. In *Handbook of elliptic and hyperelliptic curve cryptography*, Discrete Math. Appl. (Boca Raton), pages 573–590. Chapman & Hall/CRC, Boca Raton, FL, 2006.

[19] Graham Everest, Victor Miller, and Nelson Stephens. Primes generated by elliptic curves. *Proc. Amer. Math. Soc.*, 132(4):955–963 (electronic), 2004.

[20] Graham Everest, Alf van der Poorten, Igor Shparlinski, and Thomas Ward. *Elliptic Divisibility Sequences*, pages 163–175. American Mathematical Society, Providence, 2003.

[21] Graham Everest, Peter Rogers, and Thomas Ward. A higher-rank Mersenne problem. In *Algorithmic number theory (Sydney, 2002)*, volume 2369 of *Lecture Notes in Comput. Sci.*, pages 95–107. Springer, Berlin, 2002.

[22] Gerhard Frey and Tanja Lange. Background on curves and Jacobians. In *Handbook of elliptic and hyperelliptic curve cryptography*, Discrete Math. Appl. (Boca Raton), pages 45–85. Chapman & Hall/CRC, Boca Raton, FL, 2006.

[23] Gerhard Frey and Hans-Georg Rück. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Math. Comp.*, 62(206):865–874, 1994.

[24] Steven D. Galbraith. Pairings. In *Advances in elliptic curve cryptography*, volume 317 of *London Math. Soc. Lecture Note Ser.*, pages 183–213. Cambridge Univ. Press, Cambridge, 2005.

[25] Steven D. Galbraith. The Weil pairing on elliptic curves over $\mathbb{C}$. 2005.

[26] Steven D. Galbraith, Keith Harrison, and David Soldera. Implementing the Tate pairing. In *Algorithmic number theory (Sydney, 2002)*, volume 2369 of *Lecture Notes in Comput. Sci.*, pages 324–337. Springer, Berlin, 2002.

[27] Sergey O. Gorchinskiĭ. The Poincaré bi-extension and idèles on an algebraic curve. *Mat. Sb.*, 197(1):25–38, 2006.

[28] Darrel Hankerson, Julio López Hernandez, and Alfred Menezes. Software implementation of elliptic curve cryptography over binary fields. In *Proceedings of CHES 2000*, volume 1965 of *Lecture Notes in Comput. Sci.*, pages 1–24. Springer, Berlin, 2000.

[29] Robin Hartshorne. *Algebraic geometry*. Springer-Verlag, New York, 1977. Graduate Texts in Mathematics, No. 52.

[30] F. Hess. A note on the Tate pairing of curves over finite fields. *Arch. Math. (Basel)*, 82(1):28–32, 2004.

[31] Marc Hindry and Joseph H. Silverman. *Diophantine geometry*, volume 201 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2000. An introduction.

[32] A. N. W. Hone. Elliptic curves and quadratic recurrence sequences. *Bull. London Math. Soc.*, 37(2):161–171, 2005.

[33] Toshiya Itoh and Shigeo Tsujii. An efficient algorithm for deciding quadratic residuosity in finite fields GF(p^m). *Inform. Process. Lett.*, 30(3):111–114, 1989.

[34] Camille Jordan. *Cours d'analyse de l'École polytechnique. Tome II.* Les Grands Classiques Gauthier-Villars. [Gauthier-Villars Great Classics]. Éditions Jacques Gabay, Sceaux, 1991. Calcul intégral. [Integral calculus], Reprint of the third (1913) edition.

[35] Antoine Joux. A one round protocol for tripartite Diffie-Hellman. In *Algorithmic number theory (Leiden, 2000)*, volume 1838 of *Lecture Notes in Comput. Sci.*, pages 385–393. Springer, Berlin, 2000.

[36] Julio Lopez Kenny Fong, Darrel Hankerson and Alfred Menezes. Field inversion and point halving revisited. *Technical Report, CORR 2003-18, Department of Combinatorics and Optimization, University of Waterloo, Canada*, 2003.

[37] Neal Koblitz and Alfred Menezes. Pairing-based cryptography at high security levels. In *Cryptography and coding*, volume 3796 of *Lecture Notes in Comput. Sci.*, pages 13–36. Springer, Berlin, 2005.

[38] Serge Lang. *Abelian varieties*. Interscience Tracts in Pure and Applied Mathematics. No. 7. Interscience Publishers, Inc., New York, 1959.

[39] D. H. Lehmer. The mathematical work of Morgan Ward. *Math. Comp.*, 61(203):307–311, 1993.

[40] Stephen Lichtenbaum. Duality theorems for curves over p-adic fields. *Invent. Math.*, 7:120–136, 1969.

[41] Edouard Lucas. Theorie des Fonctions Numeriques Simplement Periodiques. *Amer. J. Math.*, 1(4):289–321, 1878.

[42] Edouard Lucas. Theorie des Fonctions Numeriques Simplement Periodiques. [Continued]. *Amer. J. Math.*, 1(3):197–240, 1878.

[43] Ben Lynn. Pairing-based cryptography library. `http://crypto.stanford.edu/pbc/`.

[44] Jerrold E. Marsden and Michael J. Hoffman. *Basic complex analysis*. W. H. Freeman and Company, New York, second edition, 1987.

[45] Alfred J. Menezes, Tatsuaki Okamoto, and Scott A. Vanstone. Reducing elliptic curve logarithms to logarithms in a finite field. *IEEE Trans. Inform. Theory*, 39(5):1639–1646, 1993.

[46] Victor S. Miller. Short programs for functions on curves. 1986.

[47] Victor S. Miller. The Weil pairing, and its efficient calculation. *J. Cryptology*, 17(4):235–261, 2004.

[48] J. S. Milne. Abelian varieties. In *Arithmetic geometry (Storrs, Conn., 1984)*, pages 103–150. Springer, New York, 1986.

[49] J. S. Milne. *Arithmetic duality theorems*. BookSurge, LLC, Charleston, SC, second edition, 2006.

[50] David Mumford. Bi-extensions of formal groups. In *Algebraic Geometry (Internat. Colloq., Tata Inst. Fund. Res., Bombay, 1968)*, pages 307–322. Oxford Univ. Press, London, 1969.

[51] David Mumford. *Abelian varieties*. Tata Institute of Fundamental Research Studies in Mathematics, No. 5. Published for the Tata Institute of Fundamental Research, Bombay, 1970.

[52] Jürgen Neukirch. *Algebraic number theory*, volume 322 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin, 1999. Translated from the 1992 German original and with a note by Norbert Schappacher, With a foreword by G. Harder.

[53] K. G. Paterson. Cryptography from pairings. In *Advances in elliptic curve cryptography*, volume 317 of *London Math. Soc. Lecture Note Ser.*, pages 215–251. Cambridge Univ. Press, Cambridge, 2005.

[54] Alexander Polishchuk. *Abelian varieties, theta functions and the Fourier transform*, volume 153 of *Cambridge Tracts in Mathematics*. Cambridge University Press, Cambridge, 2003.

[55] James Propp. Robbins forum. `http://jamespropp.org/about-robbins`.

[56] Peter Rogers. Topics in elliptic divibility sequences. Master's thesis, University of East Anglia, 2003.

[57] Maxwell Rosenlicht. Generalized Jacobian varieties. *Ann. of Math. (2)*, 59:505–530, 1954.

[58] Maxwell Rosenlicht. A universal mapping property of generalized jacobian varieties. *Ann. of Math. (2)*, 66:80–88, 1957.

[59] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *Symposium on Cryptography and Information Security*. Okinawa, Japan, 2000.

[60] Jean-Pierre Serre. *Algebraic groups and class fields*, volume 117 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1988. Translated from the French.

[61] Rachel Shipsey. *Elliptic Divibility Sequences*. PhD thesis, Goldsmiths, University of London, 2001.

[62] Joseph H. Silverman. Wieferich's criterion and the *abc*-conjecture. *J. Number Theory*, 30(2):226–237, 1988.

[63] Joseph H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1992. Corrected reprint of the 1986 original.

[64] Joseph H. Silverman. *Advanced topics in the arithmetic of elliptic curves*, volume 151 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1994.

[65] Katherine E. Stange. The Tate pairing via elliptic nets. In *Pairing-Based Cryptography - PAIRING 2007*, volume 4575 of *Lecture Notes in Comput. Sci.*, pages 329–348. Springer, Berlin, 2007.

[66] William Stein. *Sage: Open Source Mathematical Software (Version 2.10.2)*. The Sage Group, 2008. `http://www.sagemath.org`.

[67] Marco Streng. Divisibility sequences for elliptic curves with complex multiplication. `http://www.math.leidenuniv.nl/~streng/`, 2007.

[68] Christine Swart. *Elliptic curves and related sequences*. PhD thesis, Royal Holloway and Bedford New College, University of London, 2003.

[69] J. Tate. *WC-groups over* $\mathbf{p}$*-adic fields*, volume 13 of *Séminaire Bourbaki; 10e année: 1957/1958. Textes des conférences; Exposés 152 à 168; 2e éd. corrigée, Exposé 156*. Secrétariat mathématique, Paris, 1958.

[70] Graeme Taylor. Algorithms for elliptic nets. Available at `http://www.maths.ed.ac.uk/~s0677951/code.htm` and `http://aleph.straylight.co.uk/ellnet.pdf`.

[71] The PARI Group, Bordeaux. *PARI/GP, version* `2.3.2`, 2007. available from `http://pari.math.u-bordeaux.fr/`.

[72] Steven Vajda. *Fibonacci & Lucas numbers, and the golden section.* Ellis Horwood Series: Mathematics and its Applications. Ellis Horwood Ltd., Chichester, 1989. Theory and applications, With chapter XII by B. W. Conolly.

[73] Alfred J. van der Poorten. Elliptic curves and continued fractions. *J. Integer Seq.*, 8(2):Article 05.2.5, 19 pp. (electronic), 2005.

[74] Morgan Ward. Memoir on elliptic divisibility sequences. *Amer. J. Math.*, 70:31–74, 1948.

[75] Charles A. Weibel. *An introduction to homological algebra*, volume 38 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 1994.

[76] André Weil. On algebraic groups of transformations. *Amer. J. Math.*, 77:355–391, 1955.

[77] Chu Wenchang, Shalosh B. Ekhad, and Robin J. Chapman. Problems and Solutions: Solutions: 10226. *Amer. Math. Monthly*, 103(2):175–177, 1996.

Index