

Publishers' page

Publishers' page

Publishers' page

CONTENTS

1. Elliptic curve torsion points and division polynomials 1
Iftikhar A. Burhanuddin and Ming-Deh A. Huang

Elliptic curve torsion points and division polynomials

Iftikhar A. Burhanuddin and Ming-Deh A. Huang

Department of Computer Science,
University of Southern California,
Los Angeles, CA 90089, USA.
burhanud,huang@usc.edu

We present two algorithms - p -adic and l -adic - to determine $E(\mathbb{Q})_{tors}$ the group of rational torsion points on an elliptic curve. Another algorithm we introduce is one which decides whether an elliptic curve over \mathbb{Q}_p has a non-trivial p -torsion part and this comes into play in the p -adic torsion computation procedure. We also make some remarks about the discriminant of the m -division polynomial of an elliptic curve and the information it reveals about torsion points.

1. Introduction

The Mordell-Weil theorem says that given an elliptic curve E over a number field K , the group of K -rational points $E(K)$ is finitely generated. This implies that the group of K -rational torsion $E(K)_{tors}$ is finite. A theorem of Mazur states the groups which can appear as $E(K)_{tors}$, when $K = \mathbb{Q}$. The purpose of this paper is to introduce methods which efficiently compute elliptic curve rational torsion.

We begin by briefly recalling the current approaches to determine $E(\mathbb{Q})_{tors}$. Firstly, one can compute torsion in a brute force fashion using the Nagell-Lutz theorem, which states that torsion points are integral and bounded in magnitude, but this technique can be computationally expensive. This naive method was superseded by Doud's complex analytic cubic time algorithm [5], where the input length is the size of the coefficients of the elliptic curve.

Garcia-Selfa et al [7] proposed a softly quadratic time algorithm ("softly" refers to the fact that sub-linear factors are ignored), where they compute with the Tate Normal Form of an elliptic curve. Their procedure uses Loos' root-finding algorithm as a blackbox routine and does not use any

2 *Burhanuddin and Huang*

information about how the discriminants of F_m (polynomials which arise in their algorithm) are related to the discriminant of the elliptic curve. And hence a different prime is selected to compute the roots of F_m for each m .

In §4 we devise a polynomial-time algorithm (polynomial in $\log p$ and the size of the discriminant of the curve) that decides whether a given elliptic curve over \mathbb{Q}_p has a non-trivial p -torsion part. The algorithm has two subroutines, the first procedure computes $\#E_0(\mathbb{Q}_p)[p]$ and the second determines $\#E(\mathbb{Q}_p)[p]$ when E has split multiplicative reduction. The triviality of this group would imply the triviality of $E(\mathbb{Q})[p]$ and therefore this decisional procedure finds its way into our rational torsion computation algorithm.

The roots of division polynomials correspond to torsion points of the elliptic curve and §2 introduces these well-studied polynomials. Our algorithms essentially perform root finding on these polynomials. We introduce in §6 an algorithm to compute the p -torsion part using a p -adic approach (except when $p = 2$, which is discussed in §5). This algorithm has a worst case (deterministic or expected) time complexity which is softly quadratic in the size of the discriminant of the elliptic curve. An l -adic algorithm is devised in §8 to compute $E(\mathbb{Q})_{tors}$ with a worst case softly quadratic running time. A randomized avatar of this method runs expectedly in softly linear number of bit operations.

The basic idea of our algorithms is given an elliptic curve E over \mathbb{Q} we view it as a curve over \mathbb{Q}_l and use Hensel lifting (whenever it is efficient) to compute $E(\mathbb{Q}_l)[m]$, the \mathbb{Q}_l -rational m -torsion points, to desired precision ($m = p = l$ in the p -adic approach). The values of m we investigate are dictated by Mazur's result and the sufficient precision to work with is supplied by the Nagell-Lutz theorem. We then check to see if these points are in $E(\mathbb{Q})[m]$, the group of m -torsion rational points on E . We discuss time complexity analysis of the above torsion computation procedures in §9.

In the l -adic algorithm the choice of the prime l rests on the fact that the prime support of the m -division polynomial equals the prime support of m and the prime support of the discriminant of the elliptic curve, which we prove in §7.1. This relationship between the discriminants enables us to use a single "good" prime to compute the m -torsion for all m . In order to relate $\Delta(f_m)$ the discriminant of f_m the m -division polynomial to Δ the discriminant of the elliptic curve, we symbolically computed the discriminants of these polynomials using Magma for small values of m . This led us to discover a formula for $\Delta(f_m)$. In §7.2 we establish the equivalence of this formula when m is odd to a lemma of Stark [19]. Finally, we discuss

current and future directions of research in §10.

We would like to thank S. Kamienny and W. Raskind for insightful discussions about the arithmetic of elliptic curves. The authors are grateful to P. Gaudry for giving us pointers to fast p -adic computation procedures, to H. Stark for providing us with a reference to his result and to W. Stein for spurring us to think about elliptic curves with split multiplicative reduction in terms of the Tate curve. We were supported in part by the following NSF grants CCR-9820778 and CCR-0306393.

2. Division Polynomials

Let K be a number field and \overline{K} be an algebraic closure of K . Let E be an elliptic curve over K given by a Weierstrass equation of the form $y^2 = x^3 + ax + b$, where $a, b \in R$, where R is the ring of integers of K .

We begin by presenting definitions and theorems concerning torsion points and polynomials which characterize them. Define division polynomials Ψ_m recursively as follows:

$$\begin{aligned}\Psi_1 &= 1, \quad \Psi_2 = 2y, \quad \Psi_3 = 3x^4 + 6ax^2 + 12bx - a^2, \\ \Psi_4 &= (2x^6 + 10ax^4 + 40bx^3 - 10a^2x^2 - 8bax - 2a^3 - 16b^2)\Psi_2, \\ \Psi_{2k+1} &= \Psi_{k+2}\Psi_k^3 - \Psi_{k-1}\Psi_{k+1}^3, \quad k \geq 2 \\ \Psi_{2k} &= (\Psi_{k+2}\Psi_{k-1}^2 - \Psi_{k-2}\Psi_{k+1}^2)\Psi_k/\Psi_2, \quad k \geq 2.\end{aligned}$$

Define for $m > 2$, $f_m = \Psi_m$, when m is odd and $f_m = \Psi_m/\Psi_2$, when m is even. Observe that f_m (also referred to as division polynomials) are univariate. Let $d = \deg f_m$, which is equal to $\frac{m^2-1}{2}$, m is odd and $\frac{m^2-4}{2}$ otherwise. The leading coefficient of f_m is m , when m is odd and $m/2$ when m is even.

The x -coordinates of the m -torsion points of E correspond to the roots of f_m in the following way [1, Corollary III.7]: Let $P \in E(\overline{K})$, such that P is not a 2-torsion point then $P \in E(\overline{K})[m] \Leftrightarrow f_m(x(P)) = 0$. Recall that $E(\overline{K})[m] = (\mathbb{Z}/m\mathbb{Z}) \times (\mathbb{Z}/m\mathbb{Z})$ [17, Corollary 6.4b].

We will now define the discriminant of a polynomial and related notions [3, §3.3.2]. Let S be an integral domain with quotient field L and \overline{L} be an algebraic closure of L . Let $g \in S[X]$ with $n = \deg(g)$, $lc(g)$ be its leading coefficient and α_i be the roots of g in \overline{L} . Define the discriminant of g to be

$$\Delta(g) = lc(g)^{n-1+\deg(g')} \prod_{1 \leq i < j \leq n} (\alpha_i - \alpha_j)^2.$$

Let $f = x^3 + ax + b$ and hence $\Delta(f) = -(4a^3 + 27b^2)$. The discriminant of the elliptic curve is defined to be $\Delta(E) = -16(4a^3 + 27b^2)$ (also denoted

4 *Burhanuddin and Huang*

as Δ). From here on we will assume that E is an elliptic curve over \mathbb{Q} given by $y^2 = x^3 + ax + b$, with $a, b \in \mathbb{Z}$. This implies that the division polynomials have integral coefficients and hence their discriminants $\Delta(f_m) \in \mathbb{Z}$, which is clear from the definition of the discriminant in terms of the Sylvester matrix [3, Lemma 3.3.4].

3. $E(\mathbb{Q})_{tors}$ and Hensel's lemma

The purpose of this section is to present some background material [17, Chapter VIII] before we introduce our elliptic curve rational torsion algorithm. A corollary of the Mordell-Weil Theorem states that $E(\mathbb{Q})_{tors}$ is a finite group. To determine this group the methods which are currently in use are guided by the following theorems.

Nagell-Lutz Theorem: Let E be an elliptic curve over \mathbb{Q} given by

$$y^2 = x^3 + ax + b, \quad a, b \in \mathbb{Z}.$$

Suppose $O \neq P \in E(\mathbb{Q})_{tors}$ then $x(P), y(P) \in \mathbb{Z}$ and either $y(P) = 0$ or $y(P)^2 | (4a^3 + 27b^2)$.

Mazur's Theorem:

$$E(\mathbb{Q})_{tors} \cong \begin{cases} \mathbb{Z}/n\mathbb{Z}, & 1 \leq n \leq 10, 12 \quad \text{case (i),} \\ \mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2n\mathbb{Z}, & 1 \leq n \leq 4 \quad \text{case (ii).} \end{cases}$$

Observe that an elliptic curve over \mathbb{Q} can be transformed into the form which appears in the Nagell-Lutz theorem using a change of coordinates [17, Pages 46-50].

Suppose $P \in E(\mathbb{Q})_{tors} \setminus E(\mathbb{Q})[2]$ then $x(P)$ will be a root of $x^3 + ax + b - y(P)^2$. The Nagell-Lutz theorem tells us that $y(P)^2 | (4a^3 + 27b^2)$ which implies $x(P) | (b - (4a^3 + 27b^2)/k)$ for some $k \in \mathbb{Z}$. If $P \in E(\mathbb{Q})[2]$ and non-trivial then reasoning similar to the above leads to $x(P) | b$ since $y(P) = 0$. Hence the coordinates of the torsion points are $O(C)$ in magnitude, where $C = \max(|a|^3, |b|^2)$.

The brute-force approach to compute torsion is to try out all the possible values for $y(P)^2$ such that it divides $4a^3 + 27b^2$. In the worst case this is computationally expensive as it involves factoring and also $4a^3 + 27b^2$ might have many square divisors giving rise to many possibilities [5].

Instead our algorithms, l -adic and p -adic perform root finding on division polynomials using the following variant of the Hensel's lemma [6, Lemma 2.1]: Let $u \in \mathbb{Z}_p$ and $h \in \mathbb{Z}_p[x]$. Let k be such that $p^k || h'(u)$ and

assume $p^{n+k}|h(u)$ for some $n > k$. Let

$$\delta = \frac{p^{-k}h(u)}{p^{-k}h'(u)}$$

and $v = u - \delta$. Then $v \equiv u \pmod{p^n}$, $p^{2n}|h(v)$ and $p^k||h'(v)$.

Hensel's lemma leads to an efficient (softly linear time) method to lift points provided the roots of the polynomial separate early enough (k is bounded by a constant or log of the required p -adic precision), which would ensure that the initialization procedure (where the root is computed modulo p^k) does not take more than softly-linear time. When the roots of the polynomial fail to separate quickly, we resort to a p -adic polynomial factorization algorithm [15]. This factorization procedure takes as input a monic polynomial and hence we will have to use the standard trick of multiplying f_m by m^{d-1} and make a change of variables replacing $m \cdot x$ by x to render it monic.

In the torsion computation algorithms both l -adic and p -adic which follow, before the output is returned we need to check whether a $E(\mathbb{Q}_l)$ torsion point calculated to appropriate precision is in $E(\mathbb{Q})_{tors}$. This is possible because a priori we know the magnitude of the rational torsion points. Also negative integers (which are an infinite power series) are detected by noticing a recurring $l-1$ in the truncated l -adic expansion. They can be recovered as follows: $\sum_{i=0}^{\infty} a_i l^i = -(l - a_0 + \sum_{i=1}^{\log_l C} (l-1 - a_i) l^i)$, where the left hand side represents the negative integer as an l -adic number.

4. Deciding whether $E(\mathbb{Q}_p)[p]$ is non-trivial

We know that $E(\overline{\mathbb{Q}}_l)[p] = \mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/p\mathbb{Z}$. Suppose $p > 2$ and $\#E(\mathbb{Q}_l)[p] = p^2$ then the Weil Pairing [17, Corollary III.8.1.1] would imply $\mu_p \subset \mathbb{Q}_l^*$, which would hold if $p|l-1$, a contradiction. Therefore in particular $\#E(\mathbb{Q}_p)[p] = 1, p$.

The question of efficiently determining whether $\#E(\mathbb{Q}_p)[p] = p$ is one of independent interest. Our motivation to look at this problem is by viewing an elliptic curve E over \mathbb{Q} as an elliptic curve over \mathbb{Q}_p , $E(\mathbb{Q}_p)[p]$ being trivial would imply $E(\mathbb{Q})[p]$ is trivial (since the latter injects into the former). And this is useful information in $E(\mathbb{Q})_{tors}$ computing procedures as we shall see in the coming sections.

The algorithms presented in this section work with an elliptic curve over \mathbb{Q}_p . We will assume that we are presented with an elliptic curve E over \mathbb{Q} and a prime $p > 2$. We make this choice to simplify the time complexity and p -adic precision analysis of the algorithms (otherwise in the worst case

- split multiplicative reduction and $p|v_p(\Delta)$ - we will require as input the coefficients of the curve over \mathbb{Q}_p to p -digits of p -adic accuracy, where v_p is the p -adic valuation of \mathbb{Q}_p). Given an elliptic curve over the rationals we will use Tate's algorithm [18, Chapter IV.9] to compute the minimal Weierstrass equation of E at p . And by abuse of notation we will denote by E both the original elliptic curve and its minimal Weierstrass equation at p . Also we will refer to the associated discriminants of the former and latter as Δ and hopefully what we mean will be clear from the context.

It is straightforward to determine the type of reduction modulo p of the elliptic curve using Δ the minimal discriminant at p . Let $y^2 = x^3 + ax + b$ be the minimal Weierstrass equation of E an elliptic curve over \mathbb{Q}_p , where $p > 3$ then E is said to have good (bad) reduction at p if $v_p(\Delta) = 0$ ($v_p(\Delta) > 0$). On the other hand, the type of bad reduction – multiplicative or additive – can be determined as follows: E has multiplicative reduction if and only if $v_p(\Delta) \geq 1$ and $v_p(ab) = 0$ and it has additive reduction if and only if $v_p(a), v_p(b) \geq 1$ [17, Exercise VII.7.1(b)]. Also E has split multiplicative reduction if and only if $-2\bar{a}\bar{b}$ is a square in \mathbb{F}_p (otherwise E has non-split multiplicative reduction) [13, Page 27]. Other local information (for example $\#E(\mathbb{Q}_p)/E_0(\mathbb{Q}_p)$) is obtained using Tate's algorithm.

The proof of following theorem will keep us occupied for the remainder of this section:

Theorem 1: There exists an algorithm which takes as input an elliptic curve over \mathbb{Q} and a prime $p > 2$ and decides whether $\#E(\mathbb{Q}_p)[p] = p$. It has a worst case time complexity which is polynomial in the size of p and the size of the discriminant of the elliptic curve.

4.1. Computing $\#E_0(\mathbb{Q}_p)[p]$

This following algorithm determines $\#E_0(\mathbb{Q}_p)[p]$, in other words computes $\#E(\mathbb{Q}_p)[p]$ when there are no p -torsion points which reduce to a singular point.

Algorithm 2: Let E be an elliptic curve over \mathbb{Q}_p given by a minimal Weierstrass equation, where $p > 2$.

Input. We are given the coefficients of E , modulo p^2 and the type of reduction.

Output. TRUE if $\#E_0(\mathbb{Q}_p)[p] = p$ and FALSE if $\#E_0(\mathbb{Q}_p)[p] = 1$.

- (1) $n \leftarrow \#\bar{E}_{ns}(\mathbb{F}_p)$.
- (2) If $p \nmid n$ return FALSE.

- (3) Pick a non-trivial point $\bar{P} \in \bar{E}_{ns}(\mathbb{F}_p)[p]$.
- (4) Lift \bar{P} to $P \in E_0(\mathbb{Q}_p) \setminus E_1(\mathbb{Q}_p)$ using Hensel's lemma such that $P \equiv \bar{P} \pmod{p}$. We only need to determine $x(P)$ modulo p^2 .
- (5) Compute $x([p-1]P) \pmod{p^2}$ using the repeated squaring trick [10, Page 23] and the elliptic curve group law formulae [17, Algorithm 2.3].
- (6) If $x([p-1]P) \equiv x(P) \pmod{p^2}$ return TRUE. Otherwise return FALSE.

Lemma 3: *The above algorithm works as desired.*

Proof: First we recall a fact about the structure of $\bar{E}_{ns}(\mathbb{F}_p)$ in the case of bad reduction [17, Exercise III.3.5]: $\bar{E}_{ns}(\mathbb{F}_p) \cong \mathbb{F}_p^+, \mathbb{F}_p^*$ or $\{t \in L^* \mid N_{L/\mathbb{F}_p}(t) = 1\}$ where $L = \mathbb{F}_p(\alpha_1, \alpha_2)$ and α_1, α_2 are the slopes of tangent lines in the non-split multiplicative reduction case.

Note that $E_1(\mathbb{Q}_p) \cong \hat{E}(p\mathbb{Z}_p) \cong \hat{G}_a(p\mathbb{Z}_p)$ is torsion-free. Now the short exact sequence $0 \rightarrow E_1(\mathbb{Q}_p) \rightarrow E_0(\mathbb{Q}_p) \rightarrow \bar{E}_{ns}(\mathbb{F}_p) \rightarrow 0$ [17, Proposition VII.2.1] gives rise to the following long exact sequence via the extended snake lemma [12, Lemma II.4.1]:

$$0 \rightarrow E_0(\mathbb{Q}_p)[p] \rightarrow \bar{E}_{ns}(\mathbb{F}_p)[p] \xrightarrow{\phi} \hat{G}_a(p\mathbb{Z}_p)/p\hat{G}_a(p\mathbb{Z}_p) \rightarrow E_0(\mathbb{Q}_p)/pE_0(\mathbb{Q}_p) \rightarrow \bar{E}_{ns}(\mathbb{F}_p)/p\bar{E}_{ns}(\mathbb{F}_p) \rightarrow 0$$

If $\gcd(n, p) = 1$ then $\bar{E}_{ns}(\mathbb{F}_p)[p] = 0$ which implies that $E_0(\mathbb{Q}_p)[p] = 0$. This case takes care of split multiplicative reduction as we have $\#\bar{E}_{ns}(\mathbb{F}_p) = \#\mathbb{F}_p^* = p - 1$ and of the non-split case as we have $\#\bar{E}_{ns}(\mathbb{F}_p) = 1, p - 1, p + 1, p^2 - 1$.

If $\gcd(n, p) \neq 1$ (due to good reduction or additive reduction) and we pick a point $P \neq O$ in $\bar{E}_{ns}(\mathbb{F}_p)[p]$, then appealing to the lemma below tells us that $E_0(\mathbb{Q}_p)[p]$ being non-trivial is equivalent to $x([p-1]P) \equiv x(P) \pmod{p^2}$. Now we observe that when we compute $x([p-1]P)$ by the squaring trick, the denominators are p -adic units (part (2) of the lemma) and the group law formulae hold modulo p^2 . This suggests that only the coefficients of elliptic curve E and of the coordinates of the point P modulo p^2 contribute towards the computation. This completes the proof of the theorem. \square

Lemma 4: *Let E be an elliptic curve over \mathbb{Q}_p given by a minimal Weierstrass equation. If $Q \in E_0(\mathbb{Q}_p) \setminus E_1(\mathbb{Q}_p)$ and $\bar{Q} \in \bar{E}_{ns}(\mathbb{F}_p)[p]$ then*

- (1) $[i]Q \in E_0(\mathbb{Q}_p) \setminus E_1(\mathbb{Q}_p)$, $i = 1, \dots, p - 1$,
- (2) $x([i]Q) \not\equiv x([j]Q) \pmod{p}$, $0 < j < i < p$ and $i + j < p$,
- (3) $x([p-k]Q) \equiv x([k]Q) \pmod{p}$ and $y([p-k]Q) \equiv -y([k]Q) \pmod{p}$ (in particular $y([p-k]Q) \not\equiv y([k]Q) \pmod{p}$), $0 < k < p$,

8 *Burhanuddin and Huang*

- (4) $[p]Q \in E_i(\mathbb{Q}_p) \setminus E_{i+1}(\mathbb{Q}_p) \Leftrightarrow v_p(x([p]Q)) = -2i \Leftrightarrow v_p(x([p-1]Q) - x(Q)) = i, i \geq 1,$
(5) $x([p-1]Q) - x(Q) \equiv 0 \pmod{p^2} \Leftrightarrow \phi = 0$, where $\phi : \overline{E}_{ns}(\mathbb{F}_p)[p] \rightarrow \hat{G}_a(p\mathbb{Z}_p)/p\hat{G}_a(p\mathbb{Z}_p).$

Proof:

- (1) Suppose $[i]Q \in E_1(\mathbb{Q}_p)$ then $[i]\overline{Q} = O$ which is a contradiction since $\gcd(i, p) = 1$.
(2) Suppose $x([i]Q) \equiv x([j]Q) \pmod{p}$. This assumption combined with the fact that $[i]\overline{Q}, [j]\overline{Q} \neq O$ implies that $[i]\overline{Q} = \pm([j]\overline{Q})$ and hence $[i \pm j]\overline{Q} = O$. This is a contradiction as $\gcd(i \pm j, p) = 1$.
(3) Let $R := [p-k]Q$. Therefore $\overline{R} = [p-k]\overline{Q} = -[k]\overline{Q}$. Hence $x(\overline{R}) = x([k]\overline{Q})$ and $y(\overline{R}) = -y([k]\overline{Q})$.
(4) From part (3) we know that $x([p-k]Q) \equiv x([k]Q) \pmod{p}$. Say $v_p(x([p-k]Q) - x([k]Q)) = i$. We also know that $y([p-k]Q) \not\equiv y([k]Q) \pmod{p}$. From the group law formulae to calculate $[p]Q$ (say using $[k]Q$ and $[p-k]Q$), it follows that $v_p(x([p]Q)) = -2i$ which is equivalent to $v_p(y([p]Q)) = -3i$ [13, Proof of Theorem 7.1(c)]. And therefore $[p]Q \in E_i(\mathbb{Q}_p) \setminus E_{i+1}(\mathbb{Q}_p)$.
(5) $x([p-1]Q) - x(Q) \equiv 0 \pmod{p^2}$ implies $[p]Q \in E_2(\mathbb{Q}_p)$ by part (4). This implies that $\phi(\overline{Q}) = (\log_E \circ \lambda \circ [p])(Q) = 0 \in \hat{G}_a(p\mathbb{Z}_p)/p\hat{G}_a(p\mathbb{Z}_p)$. Here $\lambda(R) = -x(R)/y(R)$, where $R \in E_1(\mathbb{Q})$ and $\log_E(z) = z + O(z^2)$, where $z \in \hat{E}(p\mathbb{Z}_p)$.
(On the other hand $x([p-1]Q) - x(Q) \not\equiv 0 \pmod{p^2}$ implies $[p]Q \in E_1(\mathbb{Q}_p) \setminus E_2(\mathbb{Q}_p)$ by part (4). This implies that $\phi(\overline{Q}) = (\log_E \circ \lambda \circ [p])(Q) \neq 0$. \square

In step 3 we merely need to pick $x_0 \in \mathbb{F}_p$ such that $(\frac{x_0^3 + ax_0 + b}{p}) = 1$ and computing the Legendre symbol can be done in $O(\log^2 p)$ [3, Algorithm 1.4.12]. Hensel lifting can be performed in almost linear time. Step 5 would consume $\mathcal{O}(\log^2 p)$ bit operations. And therefore in the good reduction case the overall time complexity of the algorithm is dominated by the point counting routine which takes time $\mathcal{O}(\log^4 p)$ [1, Page 117].

In the case of good reduction if $p > 5$ we have two cases $\gcd(n, p) = 1$ and $n = p$. If $p = 3, 5$ we have a third case — $\gcd(n, p) = p$ and $n \neq p$ — the only instances of which are $n = 2p$ by the Hasse bound. And this is the reason we pick $\overline{P} \in \overline{E}(\mathbb{F}_p)[p]$ in the algorithm.

We will now use the output of the above algorithm to decide whether $E(\mathbb{Q}_p)[p]$ is non-trivial with the help of the following fact [18, Corollary

IV.9.2]: Let E be an elliptic curve over \mathbb{Q}_p . Then we have the following exact sequence:

$$0 \rightarrow E_0(\mathbb{Q}_p) \rightarrow E(\mathbb{Q}_p) \rightarrow G \rightarrow 0$$

where if E has split multiplicative reduction over \mathbb{Q}_p , then G is a cyclic group of order $v(\Delta) = -v(j)$, in the additive scenario the group order is at most 4 and in the non-split multiplicative instance it is either 1 or 2.

In order to weed out the spurious cases we impose some conditions.

Lemma 5: *Algorithm 2 correctly computes $\#E(\mathbb{Q}_p)[p]$ provided either*

- E has good reduction or
- E has additive reduction and $p > 3$, or
- E has additive reduction, $p = 3$ and $\gcd(\#G, 3) = 1$, or
- E has non-split multiplicative reduction or
- $\#E_0(\mathbb{Q}_p)[p] = p$.

Proof: In the case of good reduction $E_0 = E$ and $\overline{E}_{ns} = \overline{E}$ and the lemma follows. If E has bad reduction we have the following long exact sequence:

$$0 \rightarrow E_0(\mathbb{Q}_p)[p] \rightarrow E(\mathbb{Q}_p)[p] \rightarrow G[p] \rightarrow E_0(\mathbb{Q}_p)/pE_0(\mathbb{Q}_p) \rightarrow E(\mathbb{Q}_p)/pE(\mathbb{Q}_p) \rightarrow G/pG \rightarrow 0$$

Under the first 4 assumptions of the lemma we have $G[p] = 0$ and assuming the fifth case holds then $\#E(\mathbb{Q}_p)[p] = p$, and hence in all the scenarios $E_0(\mathbb{Q}_p)[p] \cong E(\mathbb{Q}_p)[p]$. \square

4.2. The algorithm when E has split multiplicative reduction at p

To deal with the split multiplicative case we use the theory of the Tate curve [18, Sections 3-5].

Algorithm 6: An elliptic curve E over \mathbb{Q}_p with split multiplicative reduction given by a minimal Weierstrass equation, where $p > 2$.

Input. $j(E)$, the j -invariant of E up to two significant p -adic digits and $v_p(j(E))$.

Output. TRUE if $\#E(\mathbb{Q}_p)[p] = p$ and FALSE if $\#E(\mathbb{Q}_p)[p] = 1$.

- (1) $g \leftarrow -v_p(j(E))$.
- (2) If $p \nmid g$ then return FALSE.
- (3) $s_0 + s_1p \leftarrow p^g \cdot j(E) \pmod{p^2}$.
- (4) $u_1 \leftarrow -\left(\frac{1}{p} \cdot \frac{s_0^{p-1} - 1}{(p-1)s_0^{p-2}}\right) \pmod{p}$.

10 *Burhanuddin and Huang*

- (5) $t_1 \leftarrow \frac{s_1 - u_1}{s_0} \pmod{p}$.
 (6) If $t_1 = 0$ return TRUE else return FALSE.

Lemma 7: *The algorithm works as claimed.*

Proof: Now let us view E as the Tate curve E_q , with $q \in \mathbb{Q}_p^*$ (by abuse of notation we will refer to both of them as E). $G = E(\mathbb{Q}_p)/E_0(\mathbb{Q}_p)$ which is a cyclic group of order $\#G = v_p(\Delta) = v_p(q) = -v_p(j(E))$. Furthermore $0 < v_p(\Delta) < \infty$ and hence step (1) is well-defined.

Recall that $E(\mathbb{Q}_p) = \mathbb{Q}_p^*/q^{\mathbb{Z}}$ [18, Theorem V.3.1(d)], $E_0(\mathbb{Q}_p) \cong \mathbb{Z}_p^*$ [18, Page 432]. We have $\mathbb{Z}_p^*[p] = 1$ (since $x^p - 1$ has only the trivial root of unity in \mathbb{Q}_p) and hence $\#E_0(\mathbb{Q}_p)[p] = 1$.

Now by the snake lemma, the short exact sequence

$$0 \rightarrow \mathbb{Z}_p^* \rightarrow \mathbb{Q}_p^*/q^{\mathbb{Z}} \xrightarrow{v_p} G \rightarrow 0$$

gives us the following long exact sequence

$$0 \rightarrow \mathbb{Q}_p^*/q^{\mathbb{Z}}[p] \xrightarrow{v_p} G[p] \xrightarrow{\delta} \mathbb{Z}_p^*/\mathbb{Z}_p^{*p} \rightarrow (\mathbb{Q}_p^*/q^{\mathbb{Z}})/(\mathbb{Q}_p^*/q^{\mathbb{Z}})^p \rightarrow G/pG \rightarrow 0$$

$G = \mathbb{Z}/v_p(q)\mathbb{Z}$ and $G[p]$ is generated by $\frac{v_p(q)}{p}$. Now $\mathbb{Z}_p^* = \mu_{p-1} \cdot (1 + p\mathbb{Z}_p)$, where μ_{p-1} are the $p-1$ st roots of unity in \mathbb{Z}_p^* . This tells us that $\mathbb{Z}_p^*/\mathbb{Z}_p^{*p} \cong (1 + p\mathbb{Z}_p)/(1 + p^2\mathbb{Z}_p) \cong \mathbb{Z}/p\mathbb{Z}$.

Observe that $\delta = 0 \Leftrightarrow \mathbb{Q}_p^*/q^{\mathbb{Z}}[p] \cong G[p]$, therefore the question is how do we determine whether $\delta = 0$. In the case that $G[p] = 0$, that is, when $p \nmid v_p(q)$ then $\#E(\mathbb{Q}_p)[p] = 1$ and the correctness of step (2) of the algorithm follows. Now let us consider the case when $p \mid v_p(q)$. By the definition of the connecting homomorphism δ , we have $\delta(\frac{v_p(q)}{p}) = p^{\frac{v_p(q)}{p}}/q^{\mathbb{Z}} + \mathbb{Z}_p^{*p}$, where $p^{v_p(q)}/q^{\mathbb{Z}} \in \mathbb{Z}_p^*$. If $p^{v_p(q)}/q \in \mathbb{Z}_p^{*p}$ then $\delta = 0$ which would imply $\#\mathbb{Q}_p^*/q^{\mathbb{Z}}[p] = p$, otherwise $G[p] \cong \mathbb{Z}_p^*/\mathbb{Z}_p^{*p}$ and $\#\mathbb{Q}_p^*/q^{\mathbb{Z}}[p] = 1$.

To check whether $p^{v_p(q)}/q \in \mathbb{Z}_p^{*p}$, firstly we will need $v_p(q)$ which is equal to $v_p(\Delta)$. The q parameter is obtained by working with the $j(E)$, the j -invariant of E [18, Lemma V.5.1]. Specifically $q \equiv j(E)^{-1} \pmod{p^{2 \cdot v_p(\Delta)}}$ and hence $p^{v_p(q)}/q \equiv j(E) \pmod{p^{2 \cdot v_p(\Delta)}}$. We want to ascertain whether the unit $p^{v_p(q)}/q \in \mathbb{Z}_p^*$ is actually in $\mathbb{Z}_p^{*p} \cong (1 + p^2\mathbb{Z}_p)$ and therefore it follows that we need to compute the unit to 2 digits of p -adic precision since $v_p(q) > p > 2$.

To decide if $s \in \mathbb{Z}_p^*$ is in fact an element of \mathbb{Z}_p^{*p} we do the following: Suppose $s = s_0 + s_1p + \dots$, then we can express it as a product of a $p-1$ st-root of unity (say $u = u_0 + u_1p + \dots$, which is obtained by Hensel lifting s_0 to a root of $x^{p-1} - 1$ in \mathbb{Z}_p^*) and a 1-unit ($t = t_0 + t_1p + \dots$). Now working

modulo p^2 , we can decide whether an element of \mathbb{Z}_p^* is in $\mathbb{Z}_p^{*p} = 1 + p^2\mathbb{Z}_p$ ($\Leftrightarrow t_1 = 0$). \square

Given Δ , the time to compute $v_p(\Delta)$ is $\mathcal{O}(\log \Delta \log p)$ [20, Theorem 9.17]. Step (4) of the algorithm where we compute the $p - 1^{\text{st}}$ -root of unity using Hensel lifting to 2 p -adic digits will cost $\mathcal{O}(\log^2 p)$ (where we work modulo p^2 to compute $s_0^{p-1} - 1$ since p divides it).

4.3. The complete algorithm

Algorithm 8: An elliptic curve E over \mathbb{Q} given by a Weierstrass equation $y^2 = x^3 + ax + b$, with $a, b \in \mathbb{Z}$.

Input. We are given the coefficients of E and a prime $p > 2$.

Output. TRUE if $\#E(\mathbb{Q}_p)[p] = p$ and FALSE if $\#E(\mathbb{Q}_p)[p] = 1$.

- (1) Compute the minimal Weierstrass equation of E at p using Tate's algorithm.
- (2) If E has additive reduction, $p = 3$ and $\#G[3] = 3$ then using f_3 and either some initialization and Hensel lifting of the singular point or p -adic factorization algorithm [15] determine whether $\#E(\mathbb{Q}_3)[3] = 1$ or 3 and return FALSE, TRUE respectively.
- (3) If E has split multiplicative reduction then return output of algorithm 6.
- (4) Return output of algorithm 2.

The time complexity of Step 1 - Tate's algorithm - is analyzed in §9. If Hensel lifting will cost more than softly quadratic time, we resort to a general purpose p -adic factorization algorithm whose expected number of bit operations is $\mathcal{O}(\log^2 \Delta)$, fixing $p = 3$. It follows from our discussion that the time complexity above algorithm is polynomial in $\log p$ and $\log \Delta$ and this completes the proof of theorem 1.

Computing $E(\mathbb{Q}_p)[p]$ efficiently seems to be an interesting problem. In certain scenarios we can use Hensel lifting with f_p and the appropriate initialization, which is dictated by the discriminant of f_p , otherwise we can resort to p -adic polynomial factorization algorithms. When p is small and fixed this method can be used to compute $E(\mathbb{Q}_p)[p]$ in softly quadratic time, a subroutine employed in our p -adic torsion rational computation procedure which is described over the following two sections.

We would like to remark that working with the Tate curve to explicitly and efficiently compute torsion points on an elliptic curve with split

multiplicative reduction seems to be an interesting problem due to the power series used to express the coefficients and coordinates of the Tate curve [18, Theorem V.3.1].

5. Computing $E(\mathbb{Q})[2]$

In this section we will look at determining the 2-torsion part of $E(\mathbb{Q})$. This procedure will come into play when we compute $E(\mathbb{Q})_{tors}$ in the following section. Also similar techniques can be pursued to compute $E(\mathbb{Q})[2^i]$ working with f_{2^i} division polynomials, where $i = 2, 3$.

Given an elliptic curve over \mathbb{Q} and a prime $l > 2$, we use Tate's algorithm to obtain a minimal Weierstrass equation at l . We begin by listing conditions to avoid singular torsion contribution instances referred to as the S2T (Singular 2-Torsion) scenarios.

Lemma 9: $E_0(\mathbb{Q}_l)[2] = E(\mathbb{Q}_l)[2]$ provided either

- (1) E has good reduction at l or
- (2) E has split multiplicative reduction and $\gcd(v_l(\Delta), 2) = 1$ or
- (3) E has additive reduction and $G[2] = 0$ or
- (4) E has non-split multiplicative reduction $G[2] = 0$ or
- (5) $\#E_0(\mathbb{Q}_l)[2] = 4$.

Proof: The proof follows for case (i) since $E = E_0$ and $\overline{E}_{ns} = \overline{E}$. Applying the extended snake lemma gives us

$$\begin{aligned} 0 \rightarrow E_0(\mathbb{Q}_l)[2] \rightarrow E(\mathbb{Q}_l)[2] \rightarrow G[2] \rightarrow E_0(\mathbb{Q}_l)/2E_0(\mathbb{Q}_l) \rightarrow E(\mathbb{Q}_l)/2E(\mathbb{Q}_l) \\ \rightarrow G/2G \rightarrow 0 \end{aligned}$$

Either of the cases (ii) – (iv) of the lemma gives us $G[2] = 0$ and therefore $E_0(\mathbb{Q}_l)[2] \cong E(\mathbb{Q}_l)[2]$ For case (5), it follows from the fact that $\#E(\overline{\mathbb{Q}}_l)[2] = 4$. \square

Algorithm 10: Let E be an elliptic curve over \mathbb{Q} given by a Weierstrass equation and a prime $l > 2$. Suppose we want to compute the $E(\mathbb{Q}_l)[2]$ to $O(\log \Delta)$ digits of l -adic accuracy.

Input. We are given the coefficients of E and l .

Output. $\langle T_i \rangle$, where $T \in E(\mathbb{Q}_p)[2]$ and $i = 1, 2$.

- (1) Compute the minimal Weierstrass equation of E at l using Tate's algorithm.
- (2) $n \leftarrow \#\overline{E}_{ns}(\mathbb{F}_l)$.

- (3) If $\gcd(n, 2) = 1$ and !S2T then Return $\langle O \rangle$.
- (4) Generate non-trivial elements in $\overline{E}_{ns}(\mathbb{F}_l)[2]$, say \overline{P}_i .
- (5) Lift \overline{P}_i to $P_i \in E_0(\mathbb{Q}_l)[2]$ using $f = x^3 + ax + b$.
- (6) If !S2T then Return $\langle P_i \rangle$ and $\langle O \rangle$.
- (7) If S2T then we lift \overline{Q} , the singular point on $\overline{E}(\mathbb{F}_l)$ to $Q_j \in E(\mathbb{Q}_l)[2] \setminus E_0(\mathbb{Q}_l)[2]$ using $x^3 + ax + b$. Return $\langle P_i \rangle, \langle Q_j \rangle$ and $\langle O \rangle$.

Lemma 11: *The above algorithm works as desired.*

Proof: We use the fact that the 2-torsion points have 0 as their y -coordinates. Since $E_1(\mathbb{Q}_l)[2] = 0$ and $\mathbb{Z}_l/2\mathbb{Z}_l = 0$, the appropriate application of the snake lemma sequence tells us that $E_0(\mathbb{Q}_l)[2] \cong \overline{E}_{ns}(\mathbb{F}_l)[2]$.

If we are not in the S2T case, by lemma 9 we have $E_0(\mathbb{Q}_l)[2] \cong E(\mathbb{Q}_l)[2]$. Hence if $\gcd(n, 2) = 1$, there are no non-trivial $E(\mathbb{Q}_l)[2]$ points. On the other hand if $\gcd(n, 2) \neq 1$, we might obtain 1 or 3 non-trivial $E(\mathbb{Q}_l)[2]$ points by lifting $\overline{E}_{ns}(\mathbb{F}_l)[2]$ points.

Now if we are in the S2T case, there might be a contribution of 0, 1 or 2, $E(\mathbb{Q}_l)[2]$ points from the singular point.

The above algorithm is essentially a root-finding algorithm for the polynomial f . We use Hensel lifting when we can lift points efficiently provided $v_l(\Delta)$ is bounded (see §9). Otherwise we use an l -adic polynomial factorization algorithm [15]. Fixing the prime l the time complexity in the former case is $\mathcal{O}(\log \Delta)$ deterministic time and the latter approach takes $\mathcal{O}(\log^2 \Delta)$ expected time. Therefore a clever choice of the prime makes the algorithm compute faster. \square

6. The p -adic algorithm

Given an elliptic curve E over \mathbb{Q} this procedure computes $E(\mathbb{Q})_{tors}$ p -adically. We first compute the 2-torsion part l -adically, where $l = 3, 5$ or 7 as described in the previous section. In order to compute m -torsion, where m is the power of a prime $p > 2$, we view E as a curve over \mathbb{Q}_p and using the appropriate factor of the m -division polynomial compute $E(\mathbb{Q}_p)[m]$ points to desired precision and then check to see if they are rational. The integers m we have to consider are implied by Mazur's theorem.

If p is the prime of good reduction and $p^i \mid m$, then we can lift a \mathbb{F}_p -rational root of f_p^i using Hensel's lemma with $k = i$ because of the following lemma due to Satoh [16, Lemma 2.7]: Let E be an ordinary elliptic curve over \mathbb{Q}_p given by a minimal Weierstrass equation $y^2 = x^3 + ax + b$ that has good reduction at p , where $p > 2$ and $ab \neq 0$. Suppose $E(\mathbb{Q}_p^{ur})[p^i] \neq O$.

$P \in E(\mathbb{Q}_p^{ur})[p^i]$ is a non-trivial point, then $v_p(\partial\Psi_{p^i}(x(P))) = i$.

The above lemma will play a role in the algorithm when p happens to be 3, 5 or 7 and we compute 3, 9, 5 and 7 torsion points respectively.

The lemma deals with ordinary elliptic curves and hence before we apply the lemma we should rule out the supersingular case. The following is a test for supersingularity [1, Page 46]: Let E be an elliptic curve over \mathbb{Q}_p that had good reduction at p . E is supersingular if and only if

- $p \geq 5$ and $\#E(\mathbb{F}_p) = p + 1$.
- $p = 2, 3$ and $\#E(\mathbb{F}_p) = 1, p + 1$ or $2p + 1$.

Also if E is supersingular then $E(\mathbb{Q}_p)[p] = O$ and hence $E(\mathbb{Q})[p] = O$.

In the following algorithm we compute a non-zero element of $E(\mathbb{Q}_p)[p]$ by solving the p -division polynomial to $O(\log_p C)$ p -adic precision. We compute the p -adic expansions of the points using either Hensel lifting or p -adic polynomial factorization algorithms (see §9).

Algorithm 12:

Input. An elliptic curve E in the form $y^2 = x^3 + ax + b$, with $a, b \in \mathbb{Z}$.

Output. $\langle T, t \rangle_i$, where $T \in E(\mathbb{Q})[t]$ and $i = 1, 2$ and these points are the generators of $E(\mathbb{Q})_{tors}$.

- (1) Compute $E(\mathbb{Q})[2]$. Use a small prime ($l = 3, 5, 7$) to compute $E(\mathbb{Q}_l)[2]$.
 $r \leftarrow \#E(\mathbb{Q})[2] - 1$. Let R_1, \dots, R_r be the non-trivial 2-torsion points.
- (2) If $r = 0$ then
 - (a) For $p = 3, 5, 7$ do the following:
 - i. Compute $\#E(\mathbb{Q}_p)[p]$.
 - ii. If $\#E(\mathbb{Q}_p)[p] = 1$ then goto start of the loop and iterate with next prime.
 - iii. Compute a point Q using f_p .
 - iv. If $p = 5, 7$ then Return $\langle Q, p \rangle$.
 - v. If $p = 3$, then try to compute S , a non-trivial 9-torsion point using f_9 . If successful then
 - Return $\langle S, 9 \rangle$.
 - else Return $\langle Q, 3 \rangle$.
 - (b) Return $\langle O, 1 \rangle$.
- (3) If $r = 1$ then
 - (a) For $p = 3, 5$ do the following:
 - i. Compute $\#E(\mathbb{Q}_p)[p]$.

- ii. If $\#E(\mathbb{Q}_p)[p] = 1$ then goto start of the loop and iterate with next prime.
 - iii. Compute a point Q using f_p .
 - iv. $U \leftarrow R_1 + Q$.
 - v. If $p = 5$ then Return $\langle U, 10 \rangle$.
 - vi. If $p = 3$ then try and compute V , a non-trivial 4-point using f_4 .
If successful then
 - Return $\langle Q + V, 12 \rangle$.
 - else Return $\langle U, 6 \rangle$.
 - (b) Try and compute a non-trivial W , a non-trivial 4-torsion point using f_4 . If successful then
 - try and compute Z , a non-trivial 8-torsion point using f_8 . If successful then
 - Return $\langle Z, 8 \rangle$.
 - else Return $\langle W, 4 \rangle$.
 - (c) Return $\langle R_1, 2 \rangle$.
- (4) If $r = 3$ then
- (a) For $p = 3$ do the following:
 - i. Compute $\#E(\mathbb{Q}_p)[p]$.
 - ii. If $\#E(\mathbb{Q}_p)[p] = 1$ then exit loop.
 - iii. Compute a point Q using f_p .
 - A. $U \leftarrow R_1 + Q$.
 - B. Return $\langle U, 6 \rangle$ and $\langle R_2, 2 \rangle$.
 - (b) For R_1, R_2, R_3 do the following:
 - i. Try and compute W , a non-trivial 4-torsion point using f_4 . If successful
 - then try and compute Z , a non-trivial 8-torsion point using f_8 . If successful then
 - Return $\langle Z, 8 \rangle$ and $\langle R_1, 2 \rangle$.
 - else Return $\langle W, 4 \rangle$ and $\langle R_1, 2 \rangle$.
 - (c) Return $\langle R_1, 2 \rangle$ and $\langle R_2, 2 \rangle$.

Theorem 13: The above algorithm works as desired.

Proof: Observe that if $\#E(\mathbb{Q})[2] = 4$ then we are in case (ii) of Mazur's classification. Conversely suppose case (ii) holds then assuming $\#E(\mathbb{Q})[2] =$

16 *Burhanuddin and Huang*

1, 2 leads us to contradictions. Therefore we first compute points in $E(\mathbb{Q}_l)[2]$ and check to see if we get 1, 2, 4 points in $E(\mathbb{Q})[2]$ (equivalently $x^3 + ax + b$ has 0, 1, 3 roots in \mathbb{Q}).

The algorithm works since the map $E(\mathbb{Q})[p] \rightarrow E(\mathbb{Q}_p)[p]$ is injective and for $p > 2$, $\#E(\mathbb{Q}_p)[p] = 1, p$. The rest of the algorithm proceeds in a case-by-case fashion aided by implications of Mazur's result. \square

7. Discriminant of the division polynomial

In this section we prove a few facts about the discriminant of f_m , in particular Lemma 16, which makes l -adic torsion computation efficient.

7.1. Prime Support of $\Delta(f_m)$

Lemma 14: *Let $m = 2k + 1 > 1$ be an integer.*

- (1) $f \mid f'_m$
- (2) $2^2 \mid f'_m$
- (3) $m \mid f'_m$
- (4) $m^{d-1} \mid \Delta(f_m)$

Proof:

- (1) We will prove this by induction on m . $f'_3 = 12f$ and the base case holds. Suppose $f \mid f'_i$ for all odd $i < m$. Let us assume k is odd (in the even case a similar argument applies). We have $\Psi_{2k+1} = f_{2k+1} = f_{k+2}f_k^3 - (f_{k-1}\Psi_2)(f_{k+1}\Psi_2)^3 = f_{k+2}f_k^3 - f_{k-1}f_{k+1}^3\Psi_2^4 = f_{k+2}f_k^3 - 2^4 \cdot f_{k-1}f_{k+1}^3f^2$. Now $f'_{2k+1} = f'_{k+2}f_k^3 + 3 \cdot f_{k+2}f_k^2f'_k - 2^4 \cdot (f_{k-1}f_{k+1}^3)'f^2 - 2^5 \cdot f_{k-1}f_{k+1}^3ff'$. f divides f'_{k+2} and f'_k by the inductive assumption and hence f divides each of the terms in f'_{2k+1} . In particular f exactly divides f'_{2k+1} , otherwise f_{2k+1} would have repeated roots.
- (2) The argument is similar to the one above for part 1.
- (3) By [2, Theorem 1 and Corollary 1] we have $m \mid (\Psi_m^2)'$ for any m and $p \nmid \Psi_m^2$ for any odd prime p . Suppose $m = \prod_i p_i$, where p_i are odd primes. From the above, $m \mid \Psi_m \Psi'_m$ which implies $p_i \mid \Psi_m \Psi'_m$. Also $p_i \nmid \Psi_m$. Therefore $p_i \mid \Psi'_m$ and therefore $m \mid \Psi'_m$.
- (4) The statement follows from part 3, $lc(f_m) = m$ and the matrix definition of the discriminant, where the coefficients of f'_m are repeated on $\frac{m^2-1}{2}$ rows. \square

Lemma 15: *Let $m = 2k > 2$ be an integer.*

- (1) $k|\Delta(f_m)$
- (2) $2^2|f'_m$
- (3) $m|\Delta(f_m)$

Proof:

- (1) Recall that $lc(f_m) = k$ and the coefficient of x^{d-1} in f_m is 0. Consider the matrix associated to $R(f_m, f'_m)$. The first column of this matrix has k at entry (1, 1) and $k\frac{m^2-4}{2}$ at $(\frac{m^2-4}{2}, 1)$ and 0 elsewhere. The second column of this matrix has k at entry (2, 2) and $k\frac{m^2-4}{2}$ at $(\frac{m^2-4}{2} + 1, 2)$ and 0 elsewhere. Hence $k^2|R(f_m, f'_m)$ and $k|\Delta(f_m)$ by the definition of discriminant.
- (2) $2^2|f'_4$ and the base case holds. Suppose $2^2|f'_i$ for all even $i < m$. Now $f'_{2k} = (f_{k+2}f_{k-1}^2 - f_{k-2}f_{k+1}^2)f'_k + (f'_{k+2}f_{k-1}^2 + f_{k+2} \cdot 2 \cdot f_{k-1}f'_{k-1} - f'_{k-2}f_{k+1}^2 - f_{k-2} \cdot 2 \cdot f_{k+1}f'_{k+1})f'_k$. Let us assume that k is odd (similar analysis for the even case). From the previous lemma we know that 2^2 divides f'_k, f'_{k+2}, f'_{k-2} . By the inductive hypothesis 2^2 also divides f'_{k-1}, f'_{k+1} . Hence $2^2|f'_m$.
- (3) Follows from part 1 and 2. □

Lemma 16: *Prime support of $\Delta(f_m) = \text{prime support of } m \cup \text{prime support of } \Delta$, where $m > 2$ is an integer.*

Proof: Lemmas 14 and 15 state that 2 and m divide $\Delta(f_m)$. Hence it suffices to consider the primes which are relatively prime to m and prove that the prime support of $\Delta(f_m)$ equals the set of primes where E has bad reduction over \mathbb{Q} .

Consider E to be an elliptic curve over \mathbb{Q}_p . We will take a minimal Weierstrass equation denoted again by E and let its discriminant be Δ . Let L be a finite extension of \mathbb{Q}_p . Let $\mathfrak{p} = (\pi)$ be the prime over p in the ring of integers of L , $\mathbb{F}_{\mathfrak{p}}$ the residue field and e the ramification index.

Let $x = u^2x' + r, y = u^3y' + su^2x' + t$ be a change of coordinates giving a minimal Weierstrass equation for E/L denoted by E' . The discriminant Δ' for E' satisfies $\Delta' = u^{-12}\Delta$ and hence $v_{\mathfrak{p}}(\Delta') = -12v_{\mathfrak{p}}(u) + v_{\mathfrak{p}}(\Delta)$.

Let x_i and $x'_i, 1 \leq i \leq d$ be the roots of the m -division polynomial associated to E and E' respectively. For $i \neq j$, we have $v_{\mathfrak{p}}(x'_i - x'_j) = v_{\mathfrak{p}}(\frac{x_i - r}{u^2} - \frac{x_j - r}{u^2}) = v_{\mathfrak{p}}(\frac{x_i - x_j}{u^2})$ and hence $v_{\mathfrak{p}}(x_i - x_j) = 2v_{\mathfrak{p}}(u) + v_{\mathfrak{p}}(x'_i - x'_j)$.

Now let us consider the valuation of the discriminant of the m -division polynomial associated to E over \mathbb{Q}_p :

$$v_p(\Delta(f_m)) = (2d - 2)v_p(lc(f_m)) + 2 \sum_{1 \leq i < j \leq d} (2v_p(u) + v_p(x'_i - x'_j))$$

Suppose $(m, p) = 1$. Observe that $v_p(\cdot) = ev_p(\cdot)$. Since $v_p(m) = 0$, we have $v_p(lc(f_m)) = v_p(m) = 0$ when m is odd and when m is even, we have $v_p(lc(f_m)) = v_p(m/2) = 0$.

Case 1. Let E be an elliptic curve with potential good reduction over \mathbb{Q}_p . Let $L = \mathbb{Q}_p(E(\overline{\mathbb{Q}_p})[m])$ be a finite extension of \mathbb{Q}_p over which E has good reduction [18, Proposition IV.10.3] which means $v_p(\Delta') = 0$ and therefore $v_p(u) = v_p(\Delta)/12$.

Moreover the reduction modulo \mathfrak{p} map $E(L)[m] \rightarrow \overline{E}(\mathbb{F}_p)[m]$ is injective [17, Proposition VII.3.1 b] and hence $v_p(x'_i - x'_j) = 0$ for all $i \neq j$ and hence $v_p(\Delta(f_m)) = d(d-1)/6 \cdot v_p(\Delta)$ which implies that

$$v_p(\Delta(f_m)) = \frac{d(d-1)}{6} \cdot v_p(\Delta)$$

Hence if p is a prime of good reduction for E/\mathbb{Q} then $p \nmid \Delta(f_m)$ and if p is a prime of bad (additive) reduction then $p \mid \Delta(f_m)$.

Case 2. Let E be an elliptic curve with potential multiplicative reduction over \mathbb{Q}_p . Let $L \supset \mathbb{Q}_p(E(\overline{\mathbb{Q}_p})[m])$ be a finite extension of \mathbb{Q}_p over which E has (split) multiplicative reduction which means $v_p(\Delta') > 0$ and $v_p(c'_4) = 0$. We know that $v_p(c'_4) = v_p(u^{-4}c_4)$ therefore $v_p(u) = v_p(c_4)/4$. Also $j = c_4^3/\Delta$ and this implies $v_p(c_4) = \frac{1}{3} \cdot (v_p(\Delta) + v_p(j))$.

$$v_p(\Delta(f_m)) = \frac{d(d-1)}{6} \cdot (v_p(\Delta) + v_p(j)) + \sum_{1 \leq i < j \leq d} 2v_p(x'_i - x'_j)$$

Now $v_p(\Delta) + v_p(j) = 0$ or > 0 depending on whether E has multiplicative or additive reduction over \mathbb{Q}_p . Also in either case there exist i, j such that $v_p(x'_i - x'_j) > 0$ (x -coordinates of points which reduce to the singular point). Hence $v_p(\Delta(f_m)) > 0$ which implies $v_p(\Delta(f_m)) > 0$.

Therefore if p is a prime of bad (additive or multiplicative) reduction for E over \mathbb{Q} then $p \mid \Delta(f_m)$. \square

7.2. Discriminant formula

While investigating the discriminant of the m -division polynomials we stumbled upon a precise formula which expresses $\Delta(f_m)$ in terms of m

and the discriminant of the elliptic curve E . Based on symbolically computing discriminants of f_m , the m -division polynomials for $3 \leq m \leq 12$, we arrived at the following:

$$\Delta(f_m) = \begin{cases} (-1)^{\frac{m-1}{2}} m^{d-1} \Delta^{\frac{d(d-1)}{6}} & m \text{ odd,} \\ 2^4 m^{d-4} \Delta^{\frac{d(d-1)}{6}} & m \text{ even.} \end{cases}$$

The above formula in the odd case turns out to be equivalent to a lemma of Stark [19], which he proved using a complex-analytic approach, in particular Kronecker's second limit formula. Stark's result deals with an elliptic curve E given in Weierstrass normal form by $y^2 = 4x^3 - g_2x - g_3$, where g_2 and g_3 are rational. This equation can be parameterized by the Weierstrass \wp -function, $x = \wp(w)$, $y = \wp'(w)$ whose period lattice is written in the form $\Omega = \omega_0[1, z]$, where z is in \mathfrak{h} , the upper half plane. Suppose N is odd and v_1 and v_2 run through a set of representatives of $N^{-1}\Omega \bmod \Omega$. It is shown that

$$\prod_{v_1 \neq 0, v_2 \neq 0, v_1 \pm v_2 \neq 0} [\wp(v_1) - \wp(v_2)] = \pm N^{-2(N^2-3)} \Delta(E)^{(N^2-1)(N^2-3)/6}. \quad (1)$$

Note that the above equation is a product over of differences of torsion points, which are distinct and not inverses of each other.

Now we will establish the equivalence between our formula and that of Stark. Let m be an odd number and the roots of f_m be x_i , $1 \leq i \leq d$, then by the definition of the discriminant of a polynomial as stated in the §2,

$$\Delta(f_m) = lc(f_m)^{2d-2} \prod_{1 \leq i < j \leq d} (x_i - x_j)^2$$

where $lc(f_m) = m$ and $d = \deg f_m = \frac{m^2-1}{2}$. Using our formula modulo sign, we have

$$\prod_{i \neq j} (x_i - x_j) = \prod_{1 \leq i < j \leq d} (x_i - x_j)^2 = \frac{\Delta(f_m)}{m^{2d-2}} = m^{-\frac{m^2-3}{2}} \Delta^{\frac{(m^2-1)(m^2-3)}{24}}. \quad (2)$$

The above equation is a product of differences of distinct x -coordinates of the torsion points. Now comparing Eq. 1 with Eq. 2 and taking $m = N$ we see that they are equivalent up to fourth root, which is explained by the difference in the products.

8. The l -adic algorithm

As stated in the introduction our algorithm works as follows: given an elliptic curve E over \mathbb{Q} we view it as a curve over \mathbb{Q}_l and using the appropriate

20 *Burhanuddin and Huang*

factor of the m -division polynomial we compute the \mathbb{Q}_l -rational m -torsion points to desired precision and then check to see if they are in fact in $E(\mathbb{Q})$. The prime l is chosen such that E over \mathbb{Q}_l has good reduction. The integers m we have to consider are dictated by Mazur's theorem.

Algorithm 17: Input. An elliptic curve E in the form $y^2 = x^3 + ax + b$, with $a, b \in \mathbb{Z}$.

Output. $\langle T, t \rangle_i$, where $T \in E(\mathbb{Q})[t]$ and $i = 1, 2$ and these points are the generators of $E(\mathbb{Q})_{tors}$.

- (1) Pick a prime $l > 7$ such that $l \nmid \Delta$.
- (2) Compute $E(\mathbb{Q}_l)[2]$ using f .
- (3) $r \leftarrow \#E(\mathbb{Q})[2] - 1$. Let R_1, \dots, R_r be the non-trivial 2-torsion points.
- (4) If $r = 0$ then
 - (a) For $p = 3, 5, 7$ do the following:
 - i. If $p \nmid \#\overline{E}(\mathbb{F}_l)$ then goto start of the loop and iterate with next prime.
 - ii. Compute $Q \in E(\mathbb{Q}_l)[p] \setminus \{O\}$ using f_p .
 - iii. If $Q \notin E(\mathbb{Q})$ then goto start of the loop and iterate with next prime.
 - iv. If $p = 5, 7$ then Return $\langle Q, p \rangle$.
 - v. If $3^2 \mid \#\overline{E}(\mathbb{F}_l)$ then
 - Compute $S \in E(\mathbb{Q}_l)[9] \setminus E(\mathbb{Q}_l)[3]$ using f_9 . If $S \notin E(\mathbb{Q})$ then Return $\langle Q, 3 \rangle$ else Return $\langle S, 9 \rangle$.
 - else Return $\langle Q, 3 \rangle$.
 - (b) Return $\langle O, 1 \rangle$.
- (5) If $r = 1$ then
 - (a) For $p = 3, 5$ do the following:
 - i. If $p \nmid \#\overline{E}(\mathbb{F}_l)$ then goto start of the loop and iterate with next prime.
 - ii. Compute $Q \in E(\mathbb{Q}_l)[p] \setminus \{O\}$ using f_p .
 - iii. If $Q \notin E(\mathbb{Q})$ then goto start of the loop and iterate with next prime.
 - iv. $U \leftarrow R_1 + Q$.
 - v. If $p = 5$ then Return $\langle U, 10 \rangle$.
 - vi. If $4 \nmid \#\overline{E}(\mathbb{F}_l)$ then Return $\langle U, 6 \rangle$.
 - vii. Compute $V \in E(\mathbb{Q}_l)[4] \setminus E(\mathbb{Q}_l)[2]$ using f_4 .
 - viii. If $V \in E(\mathbb{Q})$ then

- Return $\langle V + Q, 12 \rangle$.
 - else Return $\langle U, 6 \rangle$.
- (b) If $4 \nmid \#\overline{E}(\mathbb{F}_l)$ then Return $\langle R_1, 2 \rangle$.
- (c) Compute $W \in E(\mathbb{Q}_l)[4] \setminus E(\mathbb{Q}_l)[2]$ using f_4 .
- (d) If $W \in E(\mathbb{Q})$ then
- If $8 \nmid \#\overline{E}(\mathbb{F}_l)$ then Return $\langle W, 4 \rangle$.
 - Compute $Z \in E(\mathbb{Q}_l)[8] \setminus E(\mathbb{Q}_l)[4]$ using f_8 .
 - If $Z \in E(\mathbb{Q})$ then
 - Return $\langle Z, 8 \rangle$.
 - else Return $\langle W, 4 \rangle$.
- (e) Return $\langle R_1, 2 \rangle$.
- (6) If $r = 3$ then
- (a) Do the following:
- i. If $3 \nmid \#\overline{E}(\mathbb{F}_l)$ then exit loop.
 - ii. Compute a point $Q \in E(\mathbb{Q}_l)[3] \setminus \{O\}$ using f_3 .
 - iii. If $Q \in E(\mathbb{Q})$ then
 - A. $U \leftarrow R_1 + Q$.
 - B. Return $\langle U, 6 \rangle$ and $\langle R_2, 2 \rangle$.
- (b) Do the following:
- i. If $8 \nmid \#\overline{E}(\mathbb{F}_l)$ then Return $\langle R_1, 2 \rangle$ and $\langle R_2, 2 \rangle$.
 - ii. Compute $W \in E(\mathbb{Q}_l)[4] \setminus E(\mathbb{Q}_l)[2]$ using f_4 .
 - iii. If $W \in E(\mathbb{Q})$ then
 - If $16 \nmid \#\overline{E}(\mathbb{F}_l)$ then Return $\langle W, 4 \rangle$ and $\langle R_2, 2 \rangle$.
 - Compute $Z \in E(\mathbb{Q}_l)[8] \setminus E(\mathbb{Q}_l)[4]$ using f_8 .
 - If $Z \in E(\mathbb{Q})$ then
 - Return $\langle Z, 8 \rangle$ and $\langle R_2, 2 \rangle$.
 - else Return $\langle W, 4 \rangle$ and $\langle R_2, 2 \rangle$.
- (c) Return $\langle R_1, 2 \rangle$ and $\langle R_2, 2 \rangle$.

Theorem 18: The algorithm works as desired.

Proof: Our choice of prime $l > 7$ implies that E has good reduction at l and hence the reduction map $E(\mathbb{Q}_l)[m] \rightarrow \overline{E}(\mathbb{F}_l)[m]$ is injective for all m we consider ($m = 2, 3, 4, 5, 7, 8, 9$) since $(m, l) = 1$. Suppose we want to compute the roots of the m -division polynomial. If $m \mid \#\overline{E}(\mathbb{F}_l)$ then the roots of f_m are distinct modulo l and therefore we can lift an \mathbb{F}_l -rational

root of f_m to \mathbb{Q}_l using Hensel's lemma with $k = 0$. Finally since we know that the magnitude of the rational torsion points, we check to see if the \mathbb{Q}_l -rational p -torsion computed to appropriate precision are in fact rational torsion. \square

9. Time complexity analysis of the algorithms

Let $M(N)$ denote the bit operations required to multiply two numbers of size N . We will assume that a fast integer multiplication algorithm like Schönhage-Strassen [20, Theorem 8.24] is used in which case $M(N) = O(N \log N \log \log N) = \mathcal{O}(N)$ (in soft-Oh notation). An integer of size N can be expressed p -adically using a recursive procedure called radix conversion [20, Theorem 9.17] in $O(M(N \log p) \log N)$ time. Due to the quadratic convergence of the Hensel lifting, if the desired precision is N then we can perform lifting in $O(M(N \log p))$ bit operations [20, Theorem 9.26] (assuming the degree of the polynomial is constant).

Given an elliptic curve with integral coefficients and discriminant Δ , we use Tate's algorithm [4, Chapter 3.2] to compute the minimal Weierstrass equation at a prime p , the type of reduction at p and the local index $\#G = E(Q_p)/E_0(Q_p)$ (data which is used in p -adic algorithm). Let $\gamma = \max\{\Delta, p\}$. Then time complexity of Tate's algorithm is $\mathcal{O}(\log \gamma)$ (plus the time compute the number of roots of certain quadratic and cubic congruences modulo p).

The primes we work with are very small in size compared to $\log \Delta$ - constant length in the p -adic algorithm and $\mathcal{O}(\log \log \Delta)$ in the l -adic procedure (see below) - hence in our context the running time of Tate's procedure is $\mathcal{O}(\log \Delta)$.

9.1. The p -adic algorithm

We recall that the discriminant of a polynomial gives us an upper bound on when the roots of the polynomial separate and in practice Hensel lifting might be efficiently usable at an "earlier" stage. The discriminant of the division polynomial depends on the discriminant of the elliptic curve and we shall see that p -adic information regarding the latter will lead to faster algorithms. We note that in certain scenarios, $v_p(\Delta)$ can be read from a table [18, Page 365].

Since the primes p we work with are small $p = 3, 5, 7$ the time to compute $\#E(\mathbb{F}_p)$ (when p is a prime of good reduction) is negligible. Once we find an approximate root we use Hensel's lifting to compute a \mathbb{Q}_p -rational

root up to $O(\log_p C)$ accuracy and the time complexity of this operation is $O(M((\log_p C) \log p)) = \mathcal{O}(\log C) = \mathcal{O}(\log \Delta)$ bit operations. Hence the idea is to find an approximate root of the division polynomial in $\mathcal{O}(\log \Delta)$ such that the overall time complexity of lifting is still softly linear. In our context, if for $p = 3, 5$ and 7 , $v_p(\Delta) = \text{constant}$ or $O(\log \log \Delta)$ then $v_p(\Delta(f_p)) = \text{constant}$ or $O(\log \log \Delta)$ and we end up with the lifting operation being softly linear. In particular from the formula for the discriminant we know that $v_p(\Delta(f_p)) = \frac{p^2-3}{2} + \frac{(p^2-3)(p^2-1)}{24}v_p(\Delta)$. Hence if p is a prime of good reduction the roots of f_p separate at a level which is independent of Δ . Moreover in this situation computation to find an approximate root is reduced drastically due to Satoh's lemma. When $m = 2, 4, 8$, we compute l -adically with f, f_4, f_8 respectively where l is either $3, 5$ or 7 . Reasoning similar to the above tells us that if $v_l(\Delta) = \text{constant}$ or $O(\log \log \Delta)$ then lifting costs softly linear time.

On the other hand without the above bounds on the valuation of the discriminant, Hensel lifting may be expensive and in this case we resort to factoring f_m using p -adic and l -adic (when $m = 2, 4, 8$) polynomial factorization algorithms [15] whose time complexity after fixing the prime and the degrees of the polynomials is expected softly quadratic in $\log \Delta$. In this scenario instead of naive lifting techniques we choose polynomial factorization algorithms because the former could take exponential time in $\log \Delta$.

Therefore in the worst case the overall time complexity of computing $E(\mathbb{Q})_{tors}$ is $\mathcal{O}(\log^2 \Delta)$ deterministic or expected depending on the flow of control through the algorithm.

9.2. The l -adic algorithm

To deterministically find a prime $l > 7$ which does not divide Δ takes $\mathcal{O}(\log^2 \Delta)$ [7]. We compute with f and m -division polynomials, where $m = 3, 4, 5, 7, 8, 9$. and the size of the coefficients of these polynomials are bounded by $O(\log C) = O(\log \Delta)$ [7]. Hence by resorting to a randomized algorithm [20, Corollary 18.12 (ii)], the expected running time to pick a good prime is $\mathcal{O}(\log \Delta)$. In either case the magnitude of this prime is $\mathcal{O}(\log \Delta)$ and hence the time to compute $\#\overline{E}(\mathbb{F}_l)$ is negligible and to naively find an \mathbb{F}_l -rational root of the division polynomial is $O(l)$ (or expected $\mathcal{O}(\log l)$ [20, Corollary 14.16]) operations in \mathbb{F}_l . Once we find an approximate root we use Hensel lifting (with $k = 0$) to a compute \mathbb{Q}_l -rational root up to $O(\log_l C)$ accuracy and the time complexity of this operation is $O(M((\log_l C) \log l)) = \mathcal{O}(\log C) = \mathcal{O}(\log \Delta)$ bit operations.

Therefore the finding the good prime routine dictates the overall running time of the l -adic algorithm, which is $\mathcal{O}(\log^2 \Delta)$ deterministic time or an expected running time of $\mathcal{O}(\log \Delta)$. The idea is that if we can find a good prime which is small in magnitude quickly in $\mathcal{O}(\log \Delta)$ time then the complexity of computing $E(\mathbb{Q})_{tors}$ is softly linear.

10. Conclusions

We have presented two algorithms to compute elliptic curve rational torsion both of which run in worst case softly quadratic time (deterministic or expected in the p -adic approach depending on the branch of execution and deterministic in the l -adic algorithm). We believe the l -adic algorithm is better because it is more elegant and is faster in theory as the randomized version of the l -adic algorithm runs in expected softly linear time. Our choice of using a p -adic polynomial factorization algorithm in the p -adic algorithm was a philosophical one as we wanted to compute p -adically as much as possible. Therefore in practice a hybrid algorithm, using a combination of p -adic and l -adic methods might work faster.

To extend the ideas in this paper to devise efficient algorithms to compute elliptic curve torsion over number fields, analogues to the theorems of Nagell-Lutz and Mazur might prove useful. [17, Exercise VIII.8.11] generalizes the former, while structural results à la Mazur's result exist for number field extensions of degrees 2, 3 [9], [8] respectively and for higher extension degrees there are the bounds of Osterlé and Parent [14]. We note that these results are not essential in our context as we can instead obtain an upper bound for the size of the torsion group of the given elliptic curve by computing the size of group of the elliptic curve over the residue field for a few primes. This endeavor of computing $E(K)_{tors}$ will require working with extensions of \mathbb{Q} and \mathbb{Q}_p , their associated rings of integers and residue fields. Hence arithmetic in these objects will have to be analyzed to develop efficient algorithms.

References

1. Blake I.F., et al. *Elliptic curves in cryptography*. London Mathematical Society Lecture Note Series, 265. Cambridge University Press, Cambridge, 1999.
2. Cassels, J. W. S. *A note on the division values of $\wp(u)$* . Proc. Cambridge Philos. Soc., 45 (1949), 167–172.
3. Cohen, H. *A course in computational algebraic number theory*. Graduate Texts in Mathematics, 138. Springer-Verlag, Berlin, 2000.

4. Cremona, J. *Algorithms for Modular Elliptic Curves*. Available at <http://www.maths.nott.ac.uk/personal/jec/book/fulltext/index.html>
5. Doud D. *A procedure to calculate torsion of elliptic curves over \mathbb{Q}* . Manuscripta Math. 95 (1998), no. 4, 463–469.
6. Fouquet, M., et al. *An extension of Satoh's algorithm and its implementation*, J. Ramanujan Math. Soc., 15 (2000), no. 4, 281–318.
7. Garcia-Selfa, I. et al. *Computing the rational torsion of an elliptic curve using Tate normal form*. J. Number Theory 96 (2002), no. 1, 76–88.
8. Jeon, D., et al. *On the torsion of elliptic curves over cubic number fields*. Acta Arith., 113 (2004), no .3, 291–301.
9. Kamienny, S., Mazur B. *Rational torsion of prime order in elliptic curves over number fields*. With an appendix by A. Granville. Columbia University Number Theory Seminar (New York, 1992). Astérisque No. 228 (1995), 3, 81–100.
10. Koblitz, N. *A course in number theory and cryptography*. Graduate Texts in Mathematics, 114. Springer-Verlag, New York, 1994.
11. Mazur, B. *Modular curves and the Eisenstein ideal*. Inst. Hautes Études Sci. Publ. Math. 47 (1977), 33–186.
12. Milne, J. S. *Class Field Theory*. Available at <http://jmilne.org/math/>
13. Milne, J. S. *Elliptic Curves*. Available at <http://jmilne.org/math/>
14. Parent, P. *Bornes effectives pour la torsion des courbes elliptiques sur les corps de nombres*. J. Reine Angew. Math. 506 (1999), 85–116.
15. Pauli, S. *Factoring polynomials over local fields*. J. Symbolic Comput. 32 (2001), no. 5, 533–547.
16. Satoh, T. *The canonical lift of an ordinary elliptic curve over a finite field and its point counting*. J. Ramanujan Math. Soc. 15 (2000), no. 4, 247–270.
17. Silverman, J. H. *The arithmetic of elliptic curves*. Graduate Texts in Mathematics, 106. Springer-Verlag, New York, 1986.
18. Silverman, J. H. *Advanced topics in the arithmetic of elliptic curves*. Graduate Texts in Mathematics, 151. Springer-Verlag, New York, 1994.
19. Stark, H. M. *The Theorem of Coates-Wiles revisited*. Number theory related to Fermat's last theorem (Cambridge, Mass., 1981), pp. 349–362, Progr. Math., 26, Birkhäuser, Boston, Mass., 1982.
20. von zur Gathen, J., Gerhard J. *Modern Computer Algebra*. Cambridge University Press, Cambridge, 2003.