CM COMPLEX ELLIPTIC CURVES AND ALGORITHMS COMPLEXITY

Bogdan Cânepă[1], Radu Gaba[2], Vladimir Olteanu[2]

In this paper we develop faster computer programs in order to improve the complexity orders of the algorithms of [2] by mean of which Cânepă and Gaba classified the complex elliptic curves E for which there exist subgroups (not necessarily cyclic) $C \leq (E,+)$ of order n such that the elliptic curves E/C and E are isomorphic; we also show why this isomorphism can only occur for non-singular projective curves of genus 1. We compute the complexity orders of the algorithms of [2] as well as of the new ones and provide a thoroughly comparison of the results obtained when running them on the same machine.

2000 Mathematics subject classification: Primary: 11G07, 11G15, 11Y16, Secondary: 14D22

Keywords: *elliptic curve, algorithm, non-cyclic subgroup*

1. Introduction

Let $\mathcal{H}$ be the upper half plane, $\mathcal{H} := \{z \in \mathbb{C}, \mathrm{Im}(z) > 0\}$ and let E be a complex elliptic curve and C a subgroup (not necessarily cyclic) of order $n < \infty$ of $(E,+)$. That is, C is a subgroup of order n of $E[n] := \{P \in E : [n]P = O\}$, the n-torsion subgroup of E. The group E/C has a structure of Riemann variety since C acts effectively and properly discontinuous on E, structure which is compatible with the natural projection $\pi : E \to E/C$. In addition the isogeny π is unramified of degree n: $\deg\pi = |\pi^{-1}(O)| = |C| = n$ (see [5], Theorem 3.4). Denote by $Y_0(n)$ the open modular curve defined as the quotient space $\Gamma_0(n)/\mathcal{H}$, in other words $Y_0(n)$ is the set of orbits $\{\Gamma_0(n)\tau : \tau \in \mathcal{H}\}$, where $\Gamma_0(n)$ is the "Nebentypus" congruence subgroup of level n of $SL_2(\mathbb{Z})$, acting on $\mathcal{H}$ from the left:

$$\Gamma_0(n) = \{\begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathbb{SL}_2(\mathbb{Z}) |\, c \equiv 0(\mathrm{mod} n)\}.$$

"UAIC" University of Iasi, 11 Carol 1 Blvd., 700506 Iasi Romania e-mail: bogdan_canepa@yahoo.com

Institute of Mathematics "Simion Stoilow" of the Romanian Academy, P.O. BOX 1-764 RO-014700 Bucharest e-mail: radu.gaba@imar.ro

University Politehnica of Bucharest, P.O. BOX RO-060042 Bucharest e-mail: vladimir.olteanu@cs.pub.ro

In this note we develop faster algorithms than the ones previously developped by Cânepă and Gaba in [2] and provide a thoroughly comparison of the results obtained when running them on the same machine after computing their complexity orders. We also show why it is important to study the isomorphism $E \simeq E/C$, that this isomorphism can only occur for non-singular projective curves of genus 1 and we provide a new proof by mean of Hurwitz's Theorem for the known fact that E/C is a complex elliptic curve.

2. **Preliminaries**

In [1], Cânepă and Gaba studied the complex elliptic curves E for which there exist cyclic subgroups $C \leq (E,+)$ of order n such that the elliptic curves E and E/C are isomorphic, where n is a positive integer. In [2] they extended this result and studied the complex elliptic curves E for which there exist subgroups (not necessarily cyclic) $C \leq (E,+)$ of order n such that the elliptic curves E and E/C are isomorphic, where n is a positive integer. More explicitly, in [1] Cânepă and Gaba proved the following:

Theorem 2.1. *([1], Theorem 1.1)*

Let E be a complex elliptic curve determined by the lattice $\langle 1, \tau \rangle$, $\tau \in \mathcal{H}$. Then:

i) $\exists C \leq (E,+)$ finite cyclic subgroup such that $\frac{E}{C} \simeq E \Leftrightarrow \exists u, v \in \mathbb{Q}$ such that $\tau^2 = u\tau + v$ with $\Delta = u^2 + 4v < 0$ (i.e. E admits complex multiplication);

ii) If τ satisfies the conditions of i) and $u = \frac{u_1}{u_2}, v = \frac{v_1}{v_2}, u_2 \neq 0, v_2 \neq 0, u_1, u_2, v_1, v_2 \in \mathbb{Z}, \mathrm{Gcd}(u_1, u_2) = \mathrm{Gcd}(v_1, v_2) = 1, d_2 = \mathrm{Gcd}(u_2, v_2)$, then:

$\exists C \leq (E,+)$ cyclic subgroup of order n which satisfies $\frac{E}{C} \simeq E \iff \exists (a, b') \in \mathbb{Z}^2$ with $\mathrm{Gcd}(a, b') = 1$ such that $n = \det M$, where M is the matrix

$$M = \begin{pmatrix} a & A \\ b & B \end{pmatrix}$$

and $(a, A, b, B) = \left(a, \frac{u_2 v_1}{d_2} b', \frac{u_2 v_2}{d_2} b', a + \frac{u_1 v_2}{d_2} b'\right)$;

iii) The subgroup C from ii) is $C = \langle \frac{u_{11} + u_{21}\tau}{n} \rangle$, where u_{11}, u_{21} are obtained in the following way: since $\det M = n$ and $\mathrm{Gcd}(a, A, b, B) = 1$ (one deduces easily this), the matrix M is arithmetically equivalent with the matrix:

$$M \sim \begin{pmatrix} 1 & 0 \\ 0 & n \end{pmatrix},$$

hence

$$\exists U, V \in GL_2(\mathbb{Z}) \quad \text{such that} \quad M = U \cdot \begin{pmatrix} 1 & 0 \\ 0 & n \end{pmatrix} \cdot V.$$

The elements u_{11}, u_{21} are the first column of the matrix

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix}.$$

The reason of studying the above mentioned class of Heegner points via Theorem 2.1 in [1] was, upon imposing certain conditions, to further answer the below question: "given a complex elliptic curve when can one find a cyclic subgroup of order n of E such that $(E, C) \simeq (E/C, E[n]/C)$" and Cânepă and Gaba classified in this new manner the fixed points of the action of the Fricke involution

$$w_n := \begin{pmatrix} 0 & -1 \\ n & 0 \end{pmatrix} \in GL_2(\mathbb{Q}^+)$$

on the open modular curves $Y_0(n)$ (see Theorem 2.3 of [1]). Ogg (see [6], Proposition 3) and Kenku (see [4], Theorem 2) computed this number of fixed points and, for $n > 3$, this is $\nu(n) = h(-n) + h(-4n)$ if $n \equiv 3(\mathrm{mod}4)$ and $\nu(n) = h(-4n)$ otherwise, where $h(-n)$ is the class number of primitive quadratic forms of discriminant $-n$ and $\nu(2) = \nu(3) = 2$. The reader will also obtain this number by using the second algorithm of [1]. Theorem 2.1 of [1] deals with the cyclic subgroups $C \leq (E, +)$ of order n satisfying $E \simeq \frac{E}{C}$. In [1] Cânepă and Gaba also developped the algorithm which classifies these points. In [2] they have studied this problem for the non-cyclic case as well. Consequently, in [2], Cânepă and Gaba proved the following:

Theorem 2.2. *([2], Theorem 2.1) Let E be a complex elliptic curve: Then there exists a finite subgroup C of $(E, +)$ such that $C \cong \mathbb{Z}/D_1\mathbb{Z} \times \mathbb{Z}/D_2\mathbb{Z}$, $D_1|D_2$, $D_1 \neq D_2$ and with the property that $\frac{E}{C} \simeq E$, if and only if τ satisfies the equation $\tau^2 = u\tau + v, u, v \in \mathbb{Q}, \Delta = u^2 + 4v < 0$ and there exist $(a, b') \in \mathbb{Z}^2$ with $\mathrm{Gcd}(a, b') = D_1$ such that, if we denote by a, A, b, B the numbers $(a, A, b, B) = \left(a, \frac{u_2 v_1}{d_2} b', \frac{u_2 v_2}{d_2} b', a + \frac{u_1 v_2}{d_2} b'\right)$ and by M the matrix*

$$M = \begin{pmatrix} a & A \\ b & B \end{pmatrix}$$

we have the relation $\det(M) = D_1 \cdot D_2$. *We denoted by* $u = \frac{u_1}{u_2}, v = \frac{v_1}{v_2}$, $u_2 \neq 0, v_2 \neq 0, u_1, u_2, v_1, v_2 \in \mathbb{Z}$, $\mathrm{Gcd}(u_1, u_2) = \mathrm{Gcd}(v_1, v_2) = 1, d_2 = \mathrm{Gcd}(u_2, v_2)$. *Moreover, the isomorphism $\frac{E}{C} \simeq E$ comes from a morphism of varieties: $\phi_{a,b'} : E \longrightarrow E$ which has the following properties:* $\deg(\phi_{\mathrm{a,b'}}) = D_1 \cdot D_2$, *it is a group homomorphism,* $\mathrm{Ker}(\phi) = C$ *and $\phi(z) = \lambda z$, where $\lambda = a + b\tau$.*

In [2] Cânepă and Gaba also developped the algorithm which classifies these points and implemented it in Magma. We briefly recall now this algorithm developed in [2] for the classification of the complex elliptic curves E which admit non-cyclic subgroups $C \leq (E, +)$ of order n such that the elliptic curves E and E/C are isomorphic as well as the modified algorithm which includes the cyclic case and their implementations in Magma. The complete details can be found in [2] and we keep the same notations. However, we provide below sufficient details in order to make the exposure clear enough while following [2]. It is known that the complex elliptic curves are of the form $\frac{\mathbb{C}}{L}$

for some $L = \mathbb{Z} + \mathbb{Z}\tau \subset \mathbb{C}$ where $\tau \in G = \Big\{z = x + iy \in \mathbb{C} : -\frac{1}{2} \leq x < \frac{1}{2}$ and either $|z| \geq 1$ if $x \leq 0$ or $|z| > 1$ if $x > 0\Big\}$.
If E is an elliptic curve satisfying the condition i) of Theorem 2.2, one can assume (up to an isomorphism) that E is of the form $\frac{\mathbb{C}}{L}$ with $L = \mathbb{Z} + \mathbb{Z}\tau \subset \mathbb{C}$ and $\tau \in G$. Clearly if $\tau^2 - u\tau - v = 0, u, v \in \mathbb{Q}, \Delta = u^2 + 4v < 0$ and $\tau \in G$, then one further obtains $\tau = \frac{u \pm i\sqrt{|\Delta|}}{2}$, $-1 \leq u < 1$ and $|\Delta| \geq 3$.
Now, since $\Delta = u^2 + 4v < 0$ one has that $v < 0$. Without loss of generality, one can assume $v_2 > 0, v_1 < 0$ and $u_2 > 0$. Theorem 2.2, ii) yields:

$$n = aB - bA = \Big(a + \frac{u_1 v_2}{2d} b'\Big)^2 - \frac{u_2^2 v_2^2 \Delta}{4d^2} b'^2 \quad (*) \tag{1}$$

Moreover $d = \mathrm{Gcd}(u_2, v_2)$, $\Delta = \frac{u_1^2}{u_2^2} + 4\frac{v_1}{v_2}$ and let $u_2' := u_2/d$ and $v_2' := v_2/d$. Let also $v_1 := -v_1$ and note that $u_2, v_2, v_1 > 0$ and that $\Delta \leq -3$. One multiplies $(*)$ by 4 and obtain:

$$4n = (2a + u_1 v_2' b')^2 + v_2' b'^2 (4v_1 d u_2'^2 - v_2' u_1^2) \tag{2}$$

This further leads to the inequality $4n \geq v_2' b'^2 \cdot 4v_1 d u_2'^2$ hence $n \geq v_2' b'^2 \cdot v_1 d u_2'^2$. Further denote by $\xi := 4v_1 d u_2'^2 - v_2' u_1^2$ and note that $\Delta \leq -3$ is equivalent to $\frac{u_1^2}{d^2 u_2'^2} - 4\frac{v_1}{d v_2'} \leq -3$ and furthermore to $u_1^2 v_2' - 4 d v_1 u_2'^2 \leq -3 d^2 u_2'^2 v_2'$ that is $-\xi \leq -3 d^2 u_2'^2 v_2'$, i.e. $\xi \geq 3 d^2 u_2'^2 v_2'$ $(**)$. From $(**)$ and (2) one obtains that $4n \geq v_2' b'^2 \cdot 3 d^2 u_2'^2 v_2'$ hence $4n/3 \geq v_2'^2 b'^2 \cdot d^2 u_2'^2$. Let $k := \sqrt{4n/3}$. One further obtains that u_2' runs from 1 to the integer part of k, $[k]$, v_2' from 1 to $[k/u_2']$, b' from 1 to $[k/u_2'/v_2']$ and d from 1 to $[k/u_2'/v_2'/b']$. Furthermore, $-1/2 \leq \mathrm{Re}(\tau) < 1/2$ is equivalent to $-1/2 \leq u_1/(2u_2) < 1/2$ that is $-u_2 \leq u_1 < u_2$. As a Consequencey u_1 runs from $-d u_2'$ to $d u_2' - 1$. Let $m := (2a + u_1 v_2' b')^2$. From (2) one gets that $4n + v_2'^2 b'^2 u_1^2 = m + 4 v_2' b'^2 v_1 d u_2'^2 \geq 4 v_2' b'^2 v_1 d u_2'^2$ hence $v_1 \leq \frac{4n + v_2'^2 b'^2 u_1^2}{4 v_2' b'^2 d u_2'^2}$. As a consequence, v_1 will run from 1 to $[\frac{4n + v_2'^2 b'^2 u_1^2}{4 v_2' b'^2 d u_2'^2}]$. Note that $D_1 = \mathrm{Gcd}(a, b')$ and $D_2 = n/\mathrm{Gcd}(a, b')$. Set the conditions $\mathrm{Gcd(a, b')} > 1$, $\mathrm{Gcd}(a, b') | n$, $\mathrm{Gcd}(a, b') < n/\mathrm{Gcd}(a, b')$ and $\mathrm{Gcd}(a, b')$ divides $n/\mathrm{Gcd}(a, b')$. Furthermore, $\mathrm{Gcd}(u_1, u_2) = 1$ and $\mathrm{Gcd}(v_1, v_2) = 1$. Finally, one have to make sure the condition $\tau \in G$ is entirely fulfilled by setting: ($u_1 > 0$ or $v_1 \geq v_2$) and ($u_1 \leq 0$ or $v_1 > v_2$). The first algorithm of [2] is therefore Algorithm 4.1 (see [2], page 10). Throughout the codes, the substitutions made are $b := b'$, $u_2 := u_2/d$ and $v_2 := v_2/d$, where $d = \mathrm{Gcd(u_2, v_2)}$ and b', u_2, v_2 are defined in Theorem 2.2. After modifying the first code by including the cyclic case, that is, by using the notations of Theorem 2.2 and allowing the case $D_1 = 1$ where $D_1 = \mathrm{Gcd}(a, b')$ [2] obtained the second agorithm namely Algorithm 4.2 (see [2], page 11). In the next section we will also compute their order of complexity.

3. Main Results

Let us show first why this isomorphism ($\frac{E}{C} \simeq E$) can only occur for non-singular projective curves of genus 1:

Lemma 3.1. *Let X be a non-singular projective curve of genus $g(X) \geq 2$. Then there is no non-trivial finite subgroup C of $\mathrm{Aut}(X)$ acting holomorphically and effectively on X such that $X \simeq \frac{X}{C}$.*

Proof. Assume that there exists a non-trivial finite subgroup C of $\mathrm{Aut}(X)$ acting holomorphically and effectively on X such that $X \simeq \frac{X}{C}$ and denote by $n := |C|$. Let $\pi : X \to \frac{X}{C}$ be the canonical projection. Then $\deg(\pi) = n$. By using now Hurwitz's Theorem ([5], Theorem 4.16) we obtain that $2 \cdot g(X) - 2 = (2 \cdot g(X/C) - 2) \cdot \deg(\pi) + \sum_{p \in X}(\mathrm{mult}_p \pi - 1)$. Since $\mathrm{mult}_p \pi \geq 1$ it follows that $\sum_{p \in X}(\mathrm{mult}_p \pi - 1) \geq 0$. Consequently $2 \cdot g(X) - 2 \geq (2 \cdot g(X/C) - 2) \cdot \deg(\pi)$. Denote by $w := 2 \cdot g(X) - 2$ and remark that $w \geq 2$ since $g(X) \geq 2$. Now, since $X \simeq \frac{X}{C}$ the inequality reads $w \geq w \cdot n$ hence $n \leq 1$ which is a contradiction since C was non-trivial by assumption. This completes the proof.

□

We show now in a different way, using Hurwitz's Theorem, that the quotient of a complex elliptic curve E by a subgroup of it (not necessarily cyclic) $C \leq (E, +)$ is also an elliptic curve:

Proposition 3.1. *Let E be a complex elliptic curve and $C \leq (E, +)$ a subgroup of it (not necessarily cyclic). Then $\frac{E}{C}$ is a complex elliptic curve.*

Proof. C is a (not necessarily cyclic) subgroup of order $n < \infty$ of $(E, +)$. That is, C is a subgroup of order n of $E[n] := \{P \in E : [n]P = O\}$, the n-torsion subgroup of E. As specified in the introduction it is known that since C acts effectively and properly discontinuous on E, the group E/C has a structure of Riemann variety, which is compatible with the natural projection $\pi : E \to E/C$ and that the isogeny π is unramified of degree n: $\deg\pi = |\pi^{-1}(O)| = |C| = n$ (see [5], Theorem 3.4). Since the genus of E is 1, $g(E) = 1$, E being an elliptic curve, it follows that $g(E/C) \leq 1$ hence $g(E/C) \in \{0, 1\}$. If $g(E/C)$ would be 0 then $E/C \simeq \mathbb{P}^1_{\mathbb{C}}$. From Hurwitz's Theorem ([5], Theorem 4.16) applied to $\pi : E \to \mathbb{P}^1_{\mathbb{C}}$ we obtain that $2 \cdot g(E) - 2 = (2 \cdot g(\mathbb{P}^1_{\mathbb{C}}) - 2) \cdot \deg(\pi) + \sum\nolimits'_{p \in E}(\mathrm{mult}_p \pi - 1)$ where $\sum'$ is taken over all ramification points $p \in E$. Consequently, the later becomes $2 \cdot g(E) - 2 = (2 \cdot g(\mathbb{P}^1_{\mathbb{C}}) - 2) \cdot \deg(\pi) = -2 \cdot n$ and moreover since $g(E) = 1$, the equality is equivalent to $0 = -2 \cdot n$ which is absurd. It follows that the remaining case holds namely $g(E/C) = 1$ and hence $\frac{E}{C}$ is a complex elliptic curve.

□

We compute now the complexity orders of the algorithms of [2].

Theorem 3.1. *The order of complexity of Algorithm 4.1 (C noncyclic) is $O(n^2\sqrt{n} \cdot \log^2 \mathrm{n})$.*

Proof. Since $O(\text{Floor(k)}) = O(k)$, line 3 is $O(k)$ iterations. However, since $\sum_{u2=1}^{k}(k/u2) \rightarrow \text{k} \cdot \ln(\text{k})$, lines 3 and 4 combined are $O(\text{k} \cdot \log(\text{k}))$ iterations (recall that $\sum_{u=1}^{k}(1/u) - \ln(\text{k}) \rightarrow \gamma \approx 0.57$ hence $O(\sum_{u=1}^{k}(1/u)) = O(\ln(\text{k}))$. Since Gcd is $O(\log(\text{k}))$ time, line 5 is $O(\log(\text{k}))$ iterations. Line 6 is $O(k)$ time since $(k/u_2/v_2)_{\max} = k$ (note that $k/u_2/v_2 = \frac{k}{u_2 \cdot v_2}$). Since $\sum_{b=1}^{k}(k/b) \rightarrow \text{kln(k)}$, lines 6 and 7 combined are $O(\text{k} \cdot \log(\text{k}))$ iterations. Moroever, $d_{\max} = k$ for $u_2 = 1$ hence line 8 is $O(2 \cdot k)$ iterations, that is, $O(k)$ iterations. Recall that $n = 3k^2/4$. Using the same reasoning as before, we will show that lines 8 and 9 combined cost is $O(k^2)$ iterations. For this, remark that $(v_1)_{\max} = (4 \cdot n + v_2^2 \cdot u_1^2 \cdot b^2)/(4 \cdot v_2 \cdot b^2 \cdot d \cdot u_2^2) = 3k^2/(4 \cdot v_2 \cdot b^2 \cdot d \cdot u_2^2) + v_2^2 \cdot u_1^2 \cdot b^2/(4 \cdot v_2 \cdot b^2 \cdot d \cdot u_2^2) = 3k^2/(4 \cdot v_2 \cdot b^2 \cdot d \cdot u_2^2) + v_2 \cdot u_1^2/(4 \cdot d \cdot u_2^2)$. Now, $\sum_{u1=-d \cdot u_2}^{d \cdot u_2 - 1} v_1 \leq \sum_{u1=-d \cdot u_2}^{d \cdot u_2 - 1}(v_1)_{\max} = 2d \cdot u_2 \cdot 3k^2/(4 \cdot v_2 \cdot b^2 \cdot d \cdot u_2^2) + 2d \cdot u_2 \cdot v_2 \cdot u_1^2/(4 \cdot d \cdot u_2^2) = 6k^2/(4 \cdot v_2 \cdot b^2 \cdot u_2) + v_2 \cdot u_1^2/(2 \cdot u_2)$. Note that $6k^2/(4 \cdot v_2 \cdot b^2 \cdot u_2) \leq 6k^2/4$ and $v_2 \cdot u_1^2/2 \cdot u_2 \leq k^2/2$ since $(u_1^2)_{\max} = d^2 \cdot u_2^2$ for $u_1 = -d \cdot u_2$ and since $d \leq k/(u_2 \cdot v_2 \cdot b)$ hence one obtains $v_2 \cdot u_1^2/2 \cdot u_2 \leq v_2 \cdot u_2 \cdot d^2/2 \leq k^2/(2b) \leq k^2/2$. Consequently $\sum_{u1=-d \cdot u_2}^{d \cdot u_2 - 1} v_1 \leq 6k^2/4 + k^2/2 = 2k^2$ hence lines 8 and 9 combined are $O(k^2)$ iterations. Note that $O(\text{if } \text{c}_1 \text{ then } \text{c}_2 \text{ else } \text{c}_3)$ is $O(\text{c}_1) + \text{Max}(O(\text{c}_2), O(\text{c}_3))$, which is $\text{Max}(O(\text{c}_1), O(\text{c}_2), O(\text{c}_3))$. Consequently, line 10 is $O(1)$, line 11 is $O(1)$. IsSquare function is $O(\text{sqrt})$ and since from line 11 we have that $m \leq 3k^2$, line 12 is $O(\text{sqrt}(\text{k}^2)) = O(k)$. Line 13 is $O(1)$. Line 14 is $O(1)$ since $IsEven$ is $O(1)$, line 15 is $O(1)$. Note that from line 15 one obtains $a < k \cdot \text{sqrt}(3)/2 < k$. Since $b \leq k^2$ we have that $\text{Gcd(a, b)} < \text{k}$. Finally, one obtains that line 16 is $O(\log(\text{k}))$ and that line 17 is also $O(\log(\text{k}))$. The remaining lines are $O(1)$. Summarizing, the order of complexity of the algorithm is $O(k \cdot \log(\text{k})(\log(\text{k}) + (k \cdot \log(\text{k}) \cdot k^2 \cdot (k + \log(\text{k}) + \log(\text{k}))))) = O(k \cdot \log(\text{k})(\log(\text{k}) + (k^3 \log(\text{k}) \cdot (k + \log(\text{k}))))) = O(k^4 \log^2(\text{k}) \cdot (k + \log(\text{k})))) = O(k^5 \log^2(\text{k})) = O(n^2\sqrt{n} \cdot \log^2(\sqrt{\text{n}})) = O(n^2\sqrt{n} \cdot \log^2 \text{n})$. □

Theorem 3.2. *The order of complexity of the Algorithm 4.2 (C noncyclic or cyclic) is* $O(n^2\sqrt{n} \cdot \log^2 \text{n})$.

Proof. Compared to the first algorithm, the difference is given by Line 16 in which we also allow the cases when $Gcd(a, b) = 1$. Consequently Line 16 is still $O(\log(\text{k}))$ iterations.

□

We now improve the two algorithms. We reimplement them in C++ and introduce two helper classes. The source code is available upon request. The first class, called $GCDs$, computes Gcds using dynamic programming. It holds a $(k + 1) \times (n + 1)$ matrix that stores all Gcds as they are computed. Exhaustively computing the Gcds for all pairs (i, j) where $i \leq k$ and $j \leq n$ is done in $O(k * n)$ time, and the lookup cost is $O(1)$ once a Gcd has already been computed. Hence, the amortized cost of all $GCDs :: gcd$ calls is $O(k * n) = O(k^3)$. The second class is called $Squares$, and features a single method called $sqrtIfSquare$, which returns the square root of a number if said number is

a perfect square, or -1 otherwise. The class holds a vector of size $4 * n + 1$, which is initialized with -1; then, for each i such that $i^2 < 4 * n + 1$, we populate the vector at index i^2 with i. Instantiating the $Squares$ class is done in $O(4 * n + 1) = O(k^2)$ time. All calls to $sqrtIfSquare$ are vector lookups, and hence are in $O(1)$. We obtain the following:

Theorem 3.3. *The order of complexity of the improved version of Algorithm 4.1 (case C noncyclic) is* $O(n^2 \cdot \log^2 \mathrm{n})$.

Proof. The algorithm is essentially the same as its non-optimized version, aside from the fact that Gcd and $IsSquare$ are replaced with $GCDs :: gcd$ and $Squares :: sqrtIfSquare$, which lend themselves to amortized analysis: we front-load the costs, and then treat all calls as being $O(1)$. All operations pertaining to $GCDs$ are in $O(k^3)$, and all operations pertaining to $Squares$ are in $O(k^2)$. Similarly to Algorithm 2.1, line 2 is in $O(k)$. Lines 3 and 4 are $O(\mathrm{k} \cdot \log(\mathrm{k}))$ iterations. Lines 6 and 7 are also $O(\mathrm{k} \cdot \log(\mathrm{k}))$ iterations. Lines 8 and 9 are $O(k^2)$ iterations. In summary, the order of complexity is $O(k^3) + O(k^2) + O(k) + O(\mathrm{k} \cdot \log(\mathrm{k})) \cdot \mathrm{O}(\mathrm{k} \cdot \log(\mathrm{k})) \cdot \mathrm{O}(\mathrm{k}^2) = \mathrm{O}(\mathrm{k}^4 \cdot \log^2 \mathrm{k}) = \mathrm{O}(\mathrm{n}^2 \cdot \log^2 \mathrm{n})$.

□

Similarly we obtain the Theorem 3.4:

Theorem 3.4. *The order of complexity of the improved version of Algorithm 4.2 (general case, C noncyclic or cyclic) is* $O(n^2 \cdot \log^2 \mathrm{n})$.

Proof. Algorithm 4.2 is derived from Algorithm 4.1 by deleting a check from the innermost loop, thus allowing solutions where $Gcd(a, b) = 1$. The check is in $O(1)$, hence the order of complexity will be the same as that of the optimized version of Algorithm 4.1. This completes the proof.

□

4. Examples

For large n we provide several examples of the numbers of classes of CM elliptic curves E which admit subgroups C of order n such that $E \cong E/C$ in both cases (non-cycle as well as general). The examples are gathered in Table 1 and Table 2. The value set will also contain the set provided in [2] for comparison purpose. The computations were done using Magma 2.19-9 and C++ on the same Lenovo i3-3110M laptop at 2.40 GHz and 4 GB RAM. For each n we have also recorded the CPU time it took to complete these calculations with the old codes (written in Magma) as well as with the new ones (written in C++).

5. Conclusions

This work improves the complexity orders of the algorithms of [2] by mean of the new codes implemented in C++ which can be further used in

n	the non-cyclic case	old CPU time	new CPU time
150	10	0.093s	0.001s
297	43	0.156s	0.002s
1571	0	2.465s	0.015s
2012	524	3.104s	0.028s
2017	0	2.948s	0.029s
2022	0	3.588s	0.030s
4536	3733	10.982s	0.079s
12825	2884	54.382s	0.337s
15736	6731	76.051s	0.463s

TABLE 1. number of classes of CM elliptic curves: computations for various n

n	the general case	old CPU time	new CPU time
150	348	0.102s	0.002s
297	440	0.172s	0.003s
1571	1605	3.011s	0.015s
2012	3563	3.323s	0.025s
2017	2023	3.276s	0.029s
2022	4055	4.385s	0.033s
4536	13585	11.840s	0.095s
12825	23581	58.734s	0.361s
15736	33570	81.136s	0.501s

TABLE 2. number of classes of CM elliptic curves: computations for various n

the theory of complex elliptic curves. In addition, it emphasizes the utility of the elliptic curve quotients as well as isomorphisms classes $E \simeq E/C$ while providing several applications of Hurwitz's Theorem.

REFERENCES

[1] B. Canepa, R. Gaba *On some special classes of complex elliptic curves and related algorithms*, Mathematical Reports 16(66), 4 (2014), p. 477-502.

[2] B. Canepa, R. Gaba *A generalization of a fixed point theorem for CM Elliptic Curves* , U.P.B. Sci. Bull., Series A, Vol. 81, Iss. 1, (2019), p. 3-12.

[3] D. Husemoeller, *Elliptic curves*, Graduate Texts in Mathematics, volume 111, Springer, New-York, 2004.

[4] M. A. Kenku, *Atkin-Lehner involutions and class number residuality*, Acta Arithmetica, 23 (1977), 1-9.

[5] R. Miranda, *Algebraic curves and Riemann surfaces*, Graduate Studies in Mathematics, volume 5, AMS.

[6] A. P. Ogg, *Hyperelliptic modular curves*, Bulletin de la S.M.F., tome 102 (1974), 449-462.