Visualizing Elliptic Curves and Their Tori

Galen Ballew James Duncan
University of Illinois at Chicago

Mathematical Computing Laboratory

UIC

Summary

Our research focuses on elliptic curves E over $\mathbb{Q}$ with complex multiplication (by the maximal order of an imaginary quadratic field). Viewed over $\mathbb{C}$, each E gives rise to two tori, defined by the generators ω_1 and ω_2 of the period lattice. These tori can be constructed virtually into a 3D mesh. Further, this mesh can be translated into gcode and printed using a 3D printer.

Motivation

Elliptic curves are interesting mathematical pheomena. Certain curves can be used to solve Diophantine equations, part of factoring algorithms, or used in cryptography. Elliptic curves are also a critical element in the proof of Fermat's Last Theorem. This research project is about developing an understanding of elliptic curves, their properties, and creating visualizations of them.

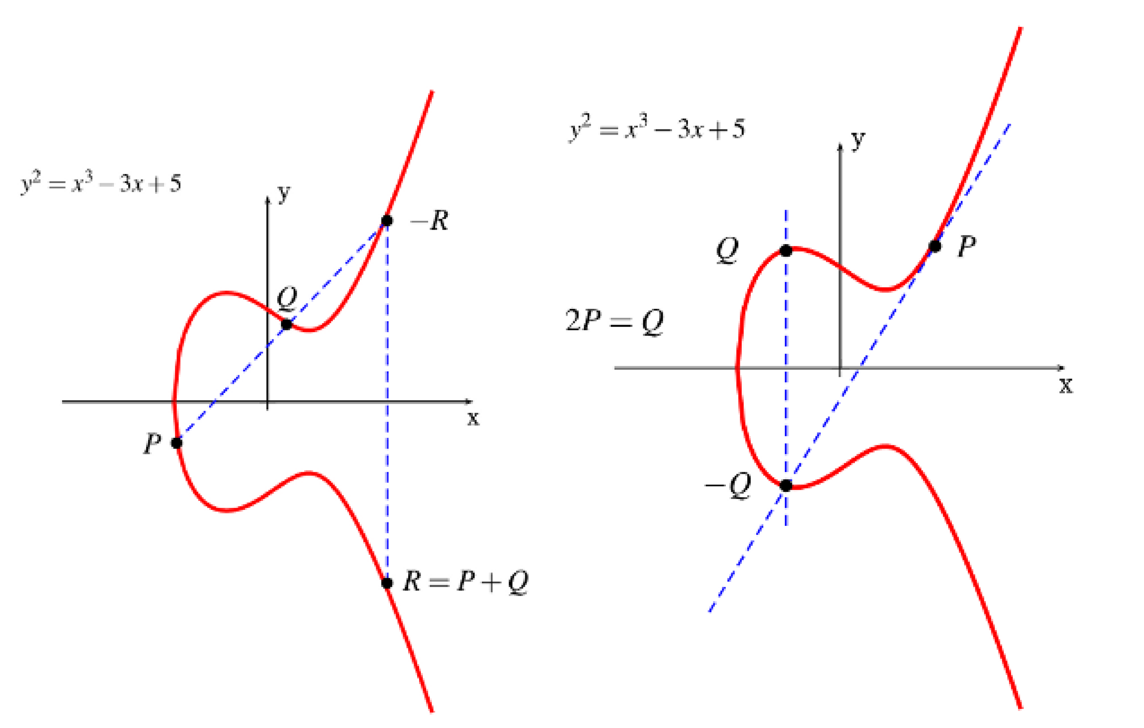
Definition

An **elliptic curve** over a field K of characteristic different than 2 and 3 is the geometric locus of an equation of the form

$$E : y^2 = x^3 + ax + b,$$

where $a, b \in K$ such that $\Delta := -16(4a^3 + 27b^2) \neq 0$, together with the projective point $\mathcal{O} = [0 : 1 : 0] \in E(K)$.

This equation is called the **Weierstrass form** of the elliptic curve. On $E(K)$, the set of points that satisfy E, we define a **group addition law via the chord-tangent method :**



E with coefficients in $\mathbb{Q}$

Let $E/\mathbb{Q}$ be an elliptic curve, and let $E(\mathbb{Q})$ be the group of points on $E/\mathbb{Q}$ with rational coefficients. Mordell's theorem states that $E(\mathbb{Q})$ is a finitely generated abelian group. Then

$$E(\mathbb{Q}) \simeq \mathbb{Z}^r \oplus E(\mathbb{Q})_{tors}$$

where $r = r(E)$ is some non-negative integer, called the **arithmetic rank** of $E/\mathbb{Q}$, and where $E(\mathbb{Q})_{tors}$ is the group of points of finite order in $E(\mathbb{Q})$, called the **torsion subgroup** of $E(\mathbb{Q})$.

E with coefficients in $\mathbb{C}$

Let $\omega_1, \omega_2 \in \mathbb{C}$ such that ω_1 and ω_2 are linearly independent over $\mathbb{R}$. We may then define the complex **lattice** Λ as

$$\Lambda = n_1\omega_1 + n_2\omega_2 : n_1, n_2 \in \mathbb{Z}.$$

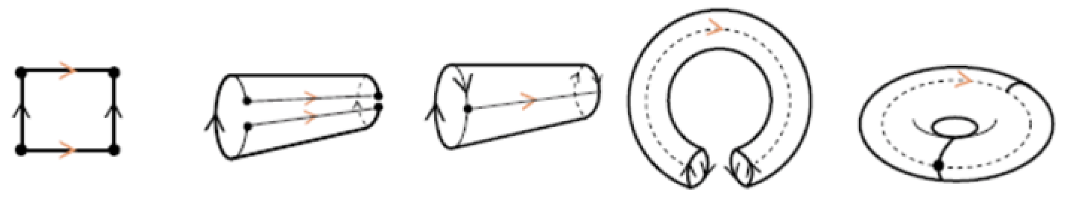
The Weierstrass $\wp$-function is doubly periodic and defined as

$$\frac{1}{u^2} + \sum_{\substack{\omega \in \Lambda \\ \omega \neq 0}} \left(\frac{1}{(u-\omega)^2} - \frac{1}{\omega^2} \right).$$

It also satifies the differential equation

$$\wp'(z)^2 = 4\wp(z)^3 - g_2\wp(z) - g_3$$

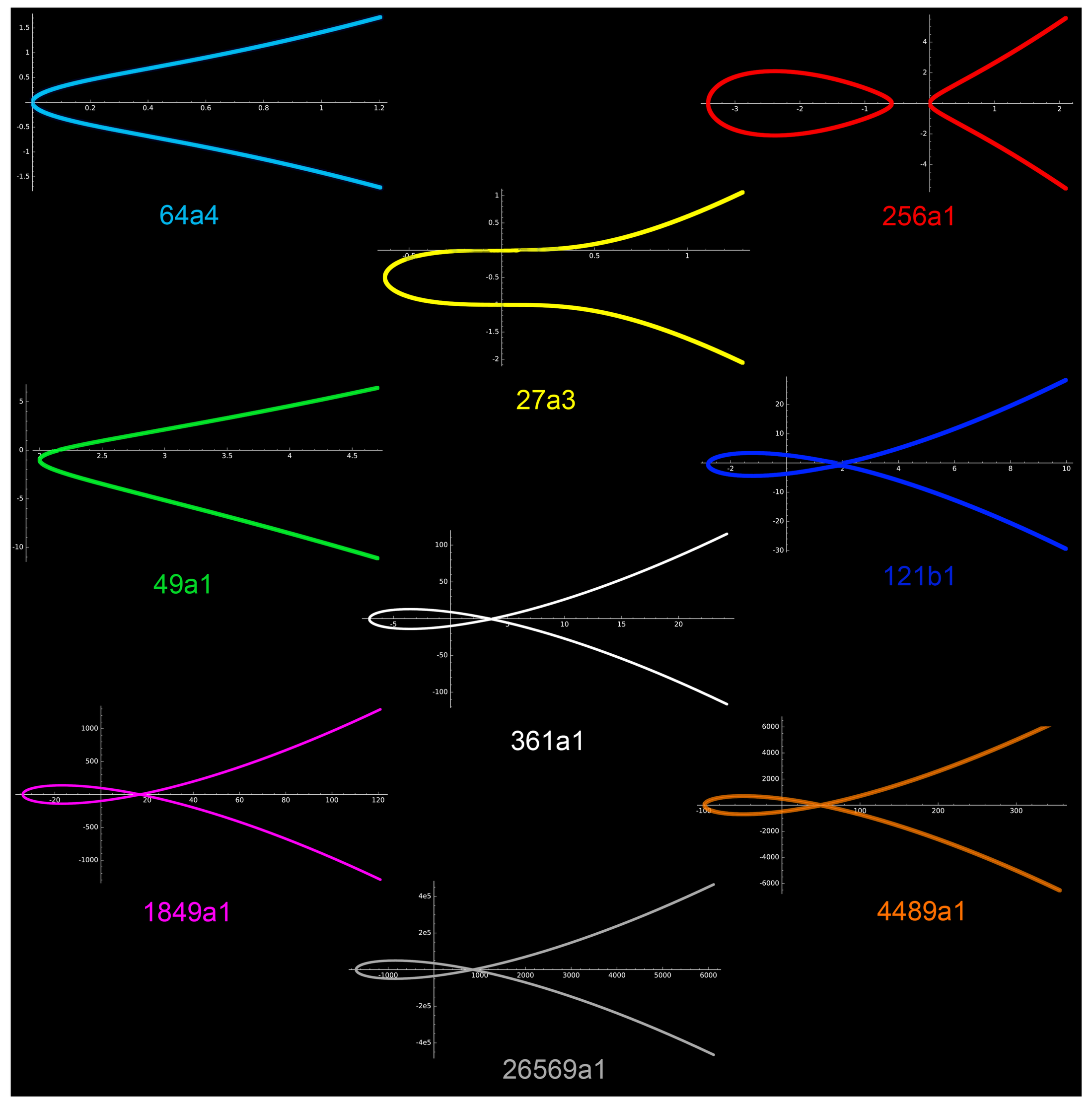
which is in the form of an elliptic curve. Π, the fundamental parallelogram of Λ, generates a torus, T^2, in the fashion below. $\mathbb{C}/\Lambda$ is topologically equivalent to T^2 and the group of poitns on T^2 is isomorphic as a group to the points on $E(\mathbb{C})$.



Complex Multiplication

Let $E/\mathbb{Q}$ be an elliptic curve with rational coefficients. We say that $E/\mathbb{Q}$ has **complex multiplication**, or CM for short, if there is an endomorphism $\phi : E/\mathbb{Q} \to E/\mathbb{Q}$ that is not a multiplication-by-n map for any integer n, so that $\mathbb{Z} \subsetneq \mathrm{End}(E)$. There are 9 curves, up to isomorphism, with CM by the maximal order of an imaginary quadratic field and they are the focus of our research.

Cremona Label	*Equation*
64a4	$y^2 = x^3 + x$
256a1	$y^2 = x^3 + 4x^2 + 2x$
27a3	$y^2 + y = x^3$
49a1	$y^2 + xy = x^3 - x^2 - 2x - 1$
121b1	$y^2 + y = x^3 - x^2 - 7x + 10$
361a1	$y^2 + y = x^3 - 38x + 90$
1849a1	$y^2 + y = x^3 - 860x^2 + 9707$
4489a1	$y^2 + y = x^3 - 7370x^2 + 243528$
26569a1	$y^2 + y = x^3 - 2174420x + 1234136692$



Sample SAGE Code

SAGE was used to calculate the period lattice of each curve, which gives the basis that defines their tori. The periods were calculated to an arbitrary precision using Gauss's Arithmetic-Geometric Mean. Further, SAGE was used to compile characteristic information about the curves, as seen in the table below.

```
def latticeHeight(lattice):
    w1, w2 = lattice.basis();
    u = vector([w1.real(),w1.imag()]);
    v = vector([w2.real(), w2.imag()]);
    cosAngle = u.dot_product(v) / (w1.abs()*w2.abs());
    sinAngle = sqrt(1 - cosAngle^2)
    h1 = w2.abs()*sinAngle
    h2 = w1.abs()*sinAngle
    h2 = w1.abs()*sinAngle
    return (h1, h2)

# returns a rhombus which we can then plot
def latticeRhombus(lattice):
    w1, w2 = lattice.basis()
    para = polygon([(0,0),
                    (w2.real(),w2.imag()),
                    (w1.real() + w2.real(), w2.imag()),
                    (w1.real(),w1.imag())])
    return para

def latticeTori(lattice):
    h1, h2 = latticeHeight(lattice)
    w1, w2 = lattice.basis()
    R1 = w1.abs() / (2*pi)
    r1 = h1 / (2*pi)
    R2 = w2.abs() / (2*pi)
    r2 = h2 / (2*pi)
    return {"Torus1": (surfaces.Torus(r1, R1), R1, r1),
            "Torus2": (surfaces.Torus(r2, R2), R2, r2)}
```

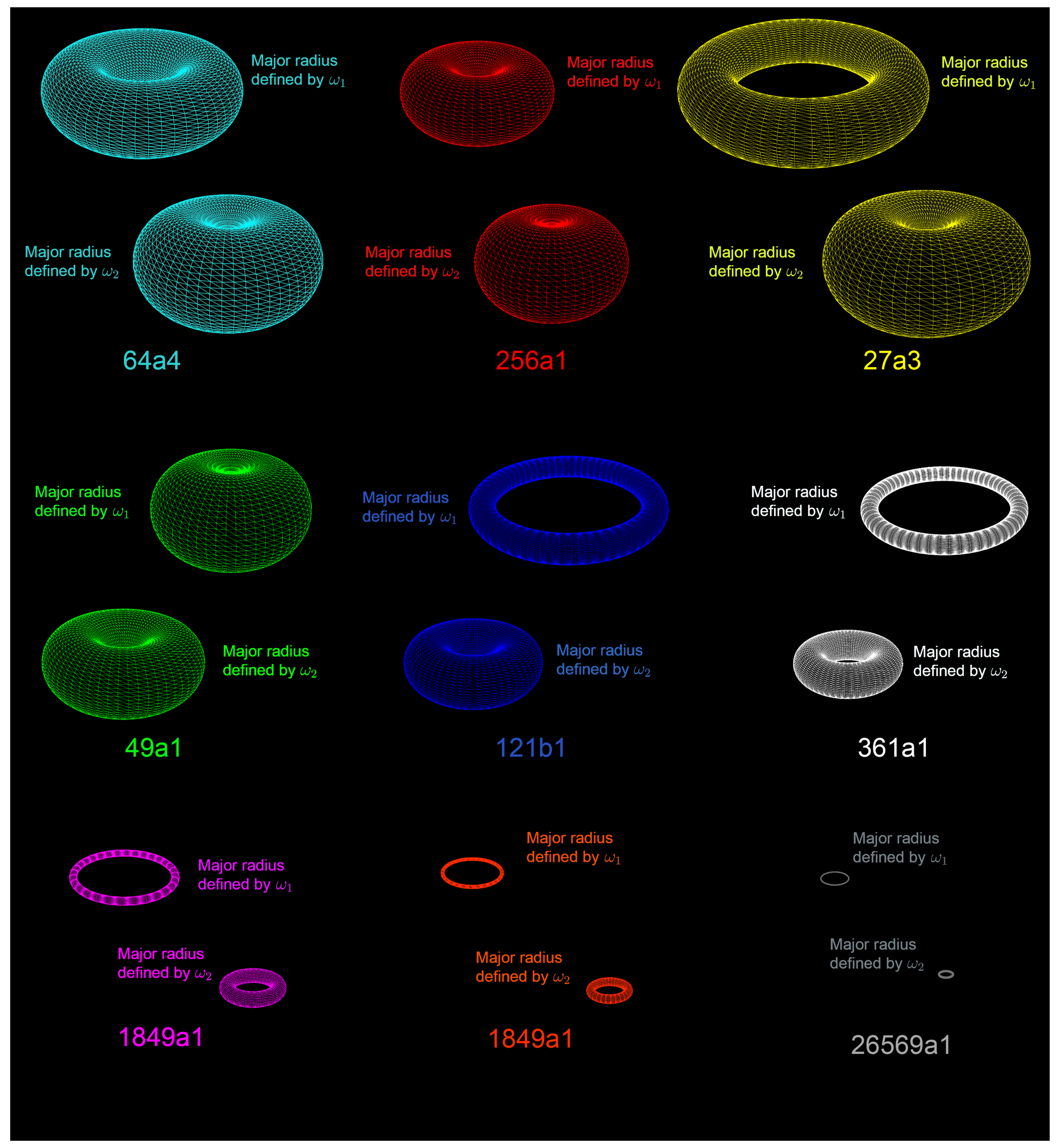
Curve Datum

Label	*Discriminant*	*Conductor*	*Torsion*	*Rank*	*CM Field*
64a4	-64	64	$\mathbb{Z}/2$	0	$\mathbb{Q}(\sqrt{-1})$
	j-invariant: 1728				
256a1	512	256	$\mathbb{Z}/2$	1	$\mathbb{Q}(\sqrt{-2})$
	j-invariant: 8000				
27a3	-27	27	$\mathbb{Z}/3$	0	$\mathbb{Q}(\sqrt{-3})$
	j-invariant: 0				
49a1	-343	49	$\mathbb{Z}/2$	0	$\mathbb{Q}(\sqrt{-7})$
	j-invariant: -3375				
121b1	-1331	121	Trivial	1	$\mathbb{Q}(\sqrt{-11})$
	j-invariant: -32768				
361a1	-6859	361	Trivial	1	$\mathbb{Q}(\sqrt{-19})$
	j-invariant: -884736				
1849a1	-79507	1849	Trivial	1	$\mathbb{Q}(\sqrt{-43})$
	j-invariant: -884736000				
4489a1	-300763	4489	Trivial	1	$\mathbb{Q}(\sqrt{-67})$
	j-invariant: -147197952000				
26569a1	-4330747	26569	Trivial	1	$\mathbb{Q}(\sqrt{-163})$
	j-invariant: -262537412640768000				

Mesh Building

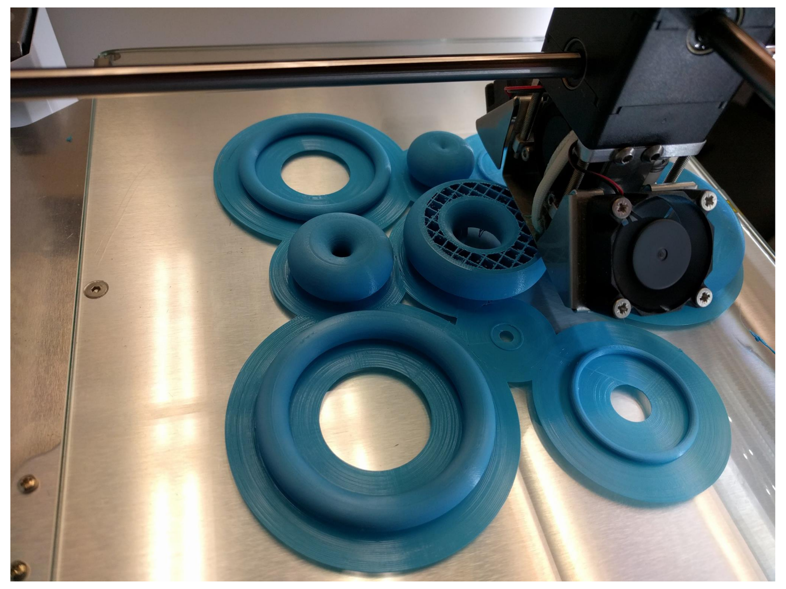
The images of the tori are wireframe renderings of their virtual 3-dimensional meshes. These were constructed in Blender and Autodesk Maya. All of the tori were built and visualized using relative scale. The meshes are the basis for the 3D prints. They were exported as .stl or .obj and imported into Cura for the Ultimaker2 3D printer.

Tori Visualization



3D Printing

The Cura software converts 3D meshes into G-code. G-code is a programming language for machine tools. It converts the 3D mesh into (X,Y,Z) coordinates for the printhead of the Ultimaker2.



Future Research

There is currently no definitive method for calculating the rank of an elliptic curve. More specifically, it is unknown whether the rank of an elliptic curve can be arbitrarily large (i.e. whether ranks are bounded or unbounded.) Currently, the largest known rank is *at least* 24, discovered by Martin and McMillen in 2000.

References

A.C. Cojocaru. *Primes, Elliptic Curves, and Cyclic Groups: A Synopsis*. (2016)

J. Siverman, J. Tate. *Rational Points on Elliptic Curves*. Undergraduate Texts in Mathematics (2015)