

$X_0(2)$

January 26, 2024

1 Modular curve $X_0(2)$

The modular curve $X_0(2)$ parameterizes pairs of elliptic curves related by a cyclic isogeny of degree 2, and it is the zero set of the modular polynomial

$$\varphi_2(x, y) = -x^2y^2 + x^3 + 1488x^2y + 1488xy^2 + y^3 - 162000x^2 + 40773375xy - 162000y^2 + 8748000000x + 8748000000y - 15746400000000$$

When $N = 2$, a computation with the Hurwitz genus formula tells us the genus of this curve is 0 (see Modular curves, Hecke correspondences, and L-functions by David E. Rorlich p.42). So, there exists a rational parametrization of this curve.

Huge shoutout to [Gaurish Korpai](#) for introducing me to modular curves.

```
[ ]: var('x,y,z,t')
phi_2 = x^3 - x^2*y^2
phi_2 += 1488*x^2*y - 162000*x^2
phi_2 += 1488*x*y^2 + 40773375*x*y
phi_2 += 8748000000*x + y^3 - 162000*y^2 + 8748000000*y - 15746400000000;
phi_2
```

```
[ ]: -x^2*y^2 + x^3 + 1488*x^2*y + 1488*x*y^2 + y^3 - 162000*x^2 + 40773375*x*y -
162000*y^2 + 8748000000*x + 8748000000*y - 15746400000000
```

Now how to find it is not that easy, for the idea Viktor Vaughn's answer to the following question <https://math.stackexchange.com/questions/4164279/find-the-rational-parametrization-of-this-curve>.

1.1 Magma Code

(use <http://magma.maths.usyd.edu.au/calc/> to evaluate)

```
QQ := Rationals();
```

```
R<x,y> := PolynomialRing(QQ,2);
```

```
f := x^3 - x^2*y^2 + 1488*x^2*y - 162000*x^2 + 1488*x*y^2 + 40773375*x*y + 8748000000*x + y^3 -
```

```
C0 := Curve(AffineSpace(R),f);
```

```

C := ProjectiveClosure(C0);
P<U,V,W> := Ambient(C);
K := CanonicalDivisor(C);
mp := DivisorMap(-K);
mp;
P<X,Y,Z> := Codomain(mp);
Q:= Image(mp);
Q;
KC<xx,yy> := FunctionField(C);
KQ<u,v> := FunctionField(Q);
mpCQ := Restriction(mp, C, Q);
mpQC := map< Q -> C | [Pushforward(mpCQ, xx), Pushforward(mpCQ, yy), 1]>;
mpQC;

```

```

[ ]: var('t,u,Y_0,Y_1,Y_2')
tilde_phi_2 = Y_0*Y_1 - 720*Y_1^2 + 4352/91*Y_1*Y_2 - 4096/4095*Y_2^2
solve(tilde_phi_2.subs(Y_0=1,Y_1 = u,Y_2=t*u) == 0, u)

```

```

[ ]: [u == 4095/16/(256*t^2 - 12240*t + 184275), u == 0]

```

```

[ ]: tilde_phi_2.subs(Y_0=1,Y_1= 0,Y_2=0).is_zero()

```

```

[ ]: True

```

```

[ ]: tilde_phi_2.subs(Y_0=1,Y_1= 4095/16/(256*t^2 - 12240*t + 184275),Y_2=t*( 4095/
↪16/(256*t^2 - 12240*t + 184275))).is_zero()

```

```

[ ]: True

```

```

[ ]: tilde_phi_2.subs(Y_0= 256*t^2 - 12240*t + 184275,Y_1=4095/16,Y_2=4095*t/16)

```

```

[ ]: 0

```

```

[ ]: Y_0= 256*t^2 - 12240*t + 184275
Y_1=4095/16
Y_2=4095*t/16
X_0 = 16769025*Y_1^4 - 196560*Y_1^3*Y_2 + 768*Y_1^2*Y_2^2 - 4096/4095*Y_1*Y_2^3

```

```

X_1 = -4095*Y_1^3*Y_2 + 768*Y_1^2*Y_2^2 - 65536/1365*Y_1*Y_2^3 + 16777216/
↳16769025*Y_2^4
X_2 = Y_1^2*Y_2^2
x =X_0/X_2
y =X_1/X_2
phi_2.subs(x=x,y=y).expand()

```

[]: 0

Since the plugging in the parametrization into $\varphi_2(x, y)$ gives 0, we've found a rational parametrization of $X_0(2)$!

```

[ ]: x = x.numerator().factor()/x.denominator()
     y = y.numerator().factor()/y.denominator()

```

```

[ ]: x,y

```

```

[ ]: (-1/4095*(16*t - 4095)^3/t^2, 1/16769025*(256*t - 4095)^3/t)

```

```

[ ]: def minus_floor(x):
     return x-floor(x)
     def build_frame(s):
         X = lambda u,v: x.subs(t = u+i*v).real()
         Y = lambda u,v: x.subs(t = u+i*v).imag()
         Z = lambda u,v,s: cos(s)*y.subs(t = u+i*v).real()+sin(s)*y.subs(t = u+i*v).
↳imag()
         cf = lambda u,v: minus_floor(sin(s)*y.subs(t = u+i*v).real()+cos(s)*y.
↳subs(t = u+i*v).imag())
         cm = colormaps.hsv
         G= (X,Y,lambda u,v: Z(u,v,s))
         P = parametric_plot3d(G,(-100, 100),(-100,100),aspect_ratio=1,colors =_
↳(cf,cm))
         P = P.add_condition(lambda x,y,z: abs(z)<2*10^4)
         P = P.add_condition(lambda x,y,z: abs(x+10800/3)<10^5/6)
         P = P.add_condition(lambda x,y,z: abs(y)<10^5/5)
         return P
frames = [build_frame(t) for t in srange(0,6.28,0.08)+[6.28]]
plot = animate(frames).interactive()
plot.show(delay=5, auto_play=False, projection='orthographic', frame=false,
↳viewpoint=[[-0.5393,-0.5633,-0.626],118.13])

```

Graphics3d Object