Lecture 31: Using Elliptic Curves to Factor, Part II

William Stein

Math 124 Harvard University **Fall 2001**

I constructed $N = 800610470601655221392794180058088102053408423$ by multiplying together five random (and promptly forgotten) primes p with the property that $p-1$ is not B-power-smooth for $B = 10^8$. Since N is a product of five not-too-big primes, N begs to be factored using the elliptic curve method.

1 The Elliptic Curve Method (ECM)

The following description of the algorithm is taken from Lenstra's paper [*Factoring Integers with Elliptic Curves*, Annals of Mathematics, **126**, 649–673], which you can download from the Math 124 web page.

Cohen and Lenstra

"The new method is obtained from Pollard's $(p-1)$-method by replacing the multiplicative group by the group of points on a random elliptic curve. To find a non-trivial divisor of an integer $n > 1$, one begins by selecting an elliptic curve E over $\mathbb{Z}/n\mathbb{Z}$, a point P on E with coordinates in $\mathbb{Z}/n\mathbb{Z}$, and an integer k as above [$k = \mathrm{lcm}(2, 3, \ldots, B)$]. Using the addition law of the curve, one next calculates the multiple $k \cdot P$ of P. One now hopes that there is a prime divisor p of n for which $k \cdot P$ and the neutral element $\mathcal{O}$ of the curve become the same modulo p; if E is given by a homogeneous Weierstrass equation $y^2z = x^3 + axz^2 + bz^3$, with $\mathcal{O} = (0 : 1 : 0)$, then this is equivalent to the z-coordinate of $k \cdot P$ being divisible by p. Hence one hopes to find a non-trivial factor of n by calculating the greatest common divisor of this z-coordinate with n."

If the above algorithm fails with a specific elliptic curve E, there is an option that is unavailable with Pollard's $(p-1)$-method. We may repeat the above algorithm with a different choice of E. The number of points on E over $\mathbb{Z}/p\mathbb{Z}$ is of the form $p+1-t$ for some t with $|t| < 2\sqrt{p}$, and the algorithm is likely to succeed if $p+1-t$ is B-power-smoth.

Suppose that $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ are nonzero points on an elliptic curve $y^2 = x^3 + ax + b$ and that $P \neq \pm Q$. Let $\lambda = (y_1 - y_2)/(x_1 - x_2)$ and $\nu = y_1 - \lambda x_1$. Recall that $P + Q = (x_3, y_3)$ where

$$x_3 = \lambda^2 - x_1 - x_2 \qquad \text{and} \qquad y_3 = -\lambda x_3 - \nu.$$

If we do arithmetic on an elliptic curve modulo N and at some point we can not compute λ because we can not compute the inverse modulo N of $x_1 - x_2$, then we (usually) factor N.

2 Implementation and Examples

For simplicity, we use an elliptic curve of the form

$$y^2 = x^3 + ax + 1,$$

which has the point $P = (0, 1)$ already on it.

The following tiny PARI function implements the ECM. It generates an error message along with a usually nontrivial factor of N exactly when the ECM succeeds.

```
{ECM(N, m)= local(E);
   E = ellinit([0,0,0,random(N),1]*Mod(1,N));
   print("E: y^2 = x^3 + ",lift(E[4]),"x+1,  P=[0,1]");
   ellpow(E,[0,1]*Mod(1,N),m);  \\ this fails if and only if we win!
}
```

The following two functions are also useful:

```
{lcmfirst(B) =
   local(L,i); L=1; for(i=2,B,L=lcm(L,i));
   return(L);
}
numpoints(a,p) = return(p+1 - ellap(ellinit([0,0,0,a,1]),p));
```

First we will try the program on a small integer N, then we will try it on the N at the top of this lecture. (ECM uses the random function, so the results of your run may differ from the one below.)

```
? N = 5959;           \\ This number motivated the ECM last time.
\\ Recall what happened when we tried to factor 5959 using the p-1 method.
? m = lcmfirst(20);   \\ B = 20.
? Mod(2,N)^m-1
%108 = Mod(5944, 5959)
? gcd(5944,5959)
%109 = 1               \\ bummer!
\\ Now we try the ECM:
? ECM(N,m)
E: y^2 = x^3 + 1201x+1,  P=[0,1]
%112 = [Mod(666, 5959), Mod(3229, 5959)]
? ECM(N,m)
E: y^2 = x^3 + 1913x+1,  P=[0,1]
  ***   impossible inverse modulo: Mod(101, 5959).
\\ Wonderful!! There's a factor--------/\
```

```
? factor(numpoints(1913,101))
%120 =
[2 4]                          \\ #E(Z/101) is 16-power-smooth,
[7 1]                          \\ so ECM sees 101.
? factor(numpoints(1913,59))
%119 =
[2 1]                          \\ #E(Z/59) is 29-power-smooth,
[29 1]                         \\ so ECM doesn't see 59.

\\ Here's the view from another angle:
? E = ellinit([0,0,0,1752,0]*Mod(1,5959));
? P = [0,1]*Mod(1,5959);
? ellpow(E,P,2)
%127 = [Mod(4624, 5959), Mod(1495, 5959)]
? ellpow(E,P,3)
%128 = [Mod(3435, 5959), Mod(1031, 5959)]
? ellpow(E,P,4)
%129 = [Mod(803, 5959), Mod(5856, 5959)]
? ellpow(E,P,8)
%133 = [Mod(1347, 5959), Mod(2438, 5959)]
? ellpow(E,P,m)
  ***   impossible inverse modulo: Mod(101, 5959).
```

Now we are ready to try the big integer N from the begining of the lecture.

```
? N = 8006104706016552213927941800580881020534O8423;
? B = 100;
? m = lcmfirst(B);
? ECM(N,m);
E: y^2 = x^3 + 273687051132207711452727265152539544370874547x+1,  P=[0,1]
... many tries ..
? ECM(N,m);
E: y^2 = x^3 + 174264237886300715545169749498695137077020788x+1,  P=[0,1]
? B=1000;  \\ give up and try a bigger B.
? m=lcmfirst(B);
? ECM(N,m);
E: y^2 = x^3 + 652986182461202633808585244537305097270008449x+1,  P=[0,1]
... many tries ...
? ECM(N,m);
E: y^2 = x^3 + 755060727645891482095225151281965348765197238x+1,  P=[0,1]
? B=10000; \\ try an even bigger B
? m=lcmfirst(B);
? ECM(N,m);
E: y^2 = x^3 + 722355978919416556225676818691766898771312229x+1,  P=[0,1]
? ECM(N,m);
E: y^2 = x^3 + 124781379199538996805045456359983628056546634x+1,  P=[0,1]
```

```
? ECM(N,m);
E: y^2 = x^3 + 3503107156272519792781442715947445140523646663x+1,  P=[0,1]
? ECM(N,m);
E: y^2 = x^3 + 3963850014650323091382382956262041054794730 7x+1,  P=[0,1]
  ***   impossible inverse modulo: Mod(1004320322301182911,
                              80061047060165522139279418005808810205340842 3).
```

Thus $N = N_1 \cdot N_2 = 1004320322301182911 \cdot 79716645459013454873760793$. One checks that neither N_1 nor N_2 is prime. Next we try ECM on each:

```
? N1 = 1004320322301182911; N2 = N / N1;
? ECM(N1,m);
E: y^2 = x^3 + 725771039569085210x+1,  P=[0,1]
  ***   impossible inverse modulo: Mod(1406051123, 1004320322301182911).
? ECM(N2,m);
E: y^2 = x^3 + 57336947544152211015643780 6x+1,  P=[0,1]
  ***   impossible inverse modulo: Mod(2029256729,
                                        79716645459013454873760793).
```

Now

$$N = N_{1,1} \cdot N_{1,2} \cdot N_{2,1} \cdot N_{2,2} = 1406051123 \cdot 714284357 \cdot 2029256729 \cdot 392836669307471617,$$

and one can check that $N_{1,1}$, $N_{1,2}$, $N_{2,1}$ are prime but that $N_{2,2}$ is composite. Again, we apply ECM:

```
? N22 = 392836669307471617
%173 = 392836669307471617
? ECM(N22,m)
E: y^2 = x^3 + 133284810657519512x+1,  P=[0,1]
%174 = [0]
? ECM(N22,m)
E: y^2 = x^3 + 368444010842952211x+1,  P=[0,1]
%175 = [Mod(236765299763600601, 392836669307471617),
                       Mod(63845045623767003, 392836669307471617)]
? ECM(N22,m)
E: y^2 = x^3 + 245772885854824846x+1,  P=[0,1]
%176 = [0]
? ECM(N22,m)
E: y^2 = x^3 + 33588046732320063x+1,  P=[0,1]
  ***   impossible inverse modulo: Mod(615433499, 392836669307471617).
```

This time it took a long time to factor $N_{2,2}$ because m is too large so we often get both factors. A smaller m would have worked more quickly. In any case, we discover that the prime factorization is

$$N = 1406051123 \cdot 714284357 \cdot 2029256729 \cdot 615433499 \cdot 638308883.$$