Math 445 Number Theory

December 6, 2004

Factoring integers using elliptic curves: the Elliptic Curve Method

The idea: use elliptic curves to factor large integers. It uses the group operation on $\mathcal{C}_f(\mathbb{Q})$, and is based on the fact that for a finite group G, with order n, every element $g \in G$ satisfies $n \cdot g = 0$.

Starting point: the Pollard $(p-1)$-test. If N is a (large) integer, with prime factor p, then by Fermat, $(a,p)=1$ implies $p|a^{p-1}-1$, and so the g.c.d. $(a^{p-1}-1,N)>1$. If we guess that $p-1$ consists of a product of fairly small primes, we can test (a^n-1,N) for n a (large) product of fairly small numbers, to arrange $1<(a^n,N)<N$, giving us a proper factor of N. In practice, we start with a randomly chosen a, and a sequence of fairly small numbers r_n, like $r_n=n$. We then form the sequence $a_1=a$, $a_2=a_1^{r_1}=a^{r_1}$, $a_3=a_2^{r_2}=a^{r_1r_2}$, and inductively, $a_{i+1}=a_i^{r_i}=a^{r_1\cdots r_i}$, and compute $g_i=(a_i-1,N)$. Since $a_i-1|a_{i+1}-1$ for every i, so $g_i|g_{i+1}$ for every i, we compute the g.c.d.'s only occasionally (since we expect to get $g_i=1$ for awhile). The process will stop, since for any prime divisor p of N, $p-1$ will divide $r_1\cdots r_n=1\cdot 2\cdots n$ for some n, so $g_n>1$. It might be that $g_n=N$, though, and so the test fails; we then restart with a different a. Typically we must wait until i is around the smallest of the largest prime factors of the $p-1$, where p ranges among all of the prime factors of N. The problem: this could be fairly large!

For the ECM, the basic idea is to take the machinery we have developed for computing on elliptic curves, and do all of the calculations mod p, for some (unknown!) prime dividing N. In practice, this really means we do the calculations mod N. Using the formulas for addition we have from above, we can create an addition formula for points in what we choose to call $\mathcal{C}_f(\mathbb{Z}_p)$. The formulas involve division; mod p, we use multiplication by the inverse (which we find by the Euclidean algorithm). We still need to know that this form of addition on $\mathcal{C}_f(\mathbb{Z}_p)$ gives us a group; this can be verified directly from the formulas (including associativity!).

$$A+B=\left(\frac{m^2-b}{a}-a_1-b_1,-(a_2+m(\frac{m^2-b}{a}-2a_1-b_1)\right)\text{ , where } m=\frac{b_2-a_2}{b_1-a_1}$$

$$2A=\left(\frac{M^2-b}{a}-2a_1,-(a_2+m(\frac{M^2-b}{a}-3a_1)\right)\text{ , where } M=\frac{3a_1^2+2aa_1+b}{2a_2}$$

To implement the ECM to find a factor of an integer N, we pick an elliptic curve $\mathcal{C}_f(\mathbb{Z}_p)$, for $f(x,y)=y^2-(x^2+ax+b)$, by choosing values for a and b, and a point A on the curve. [Usually we work the other way around; pick a point, such as $A=(1,1)$, and choose the values of a and b accordingly.] $\mathcal{C}_f(\mathbb{Z}_p)$ is a group of some finite (but unknown) order; the idea is that we expect that for some choices of a and b, it has order a product of small primes, and so a calculation like the one in the Pollard $(p-1)$-test will quickly succeed. But this is where the fun starts!

We compute high multiples $r_1\cdots r_nA$ of the point A; as we did long ago, we write $r_1\cdots r_n=2^{i_1}+\cdot+2^{i_k}$ and compute $2^{i_j}A$ by repeated doubling, and then adding together the $2^{i_j}A$ together. We want to compute mod p, but we can't; we don't know p ! Instead we compute mod N (while pretending we are computing in $\mathcal{C}_f(\mathbb{Z}_p)$). But this will not always work; not every integer has an inverse mod N. So we might eventually fail to be able to compute a step. But this is a good thing! We will fail, because the quantity we need to invert, b_1-a_1, is not relatively prime to N, i.e., $(b_1-a_1,N)>1$ (or, when doubling, $((2a_2),N)>1$). Unless this is a multiple of N we have found what we want; a proper factor of N!

In point of fact, this is what the method is designed to do; we don't want to find the order of A in $\mathcal{C}_f(\mathbb{Z}_p)$, since the order of this group really has no relation to N. It can, in fact, be any number between $p+1-2\sqrt{p}$ and $p+1+2\sqrt{p}$. What we really want to do is to discover that we can't compute the order, because the formulas break down and finds a factor of N, before the computation finishes. The point is that by varying the curve, we should be able to stumble across an f for which $\mathcal{C}_f(\mathbb{Z}_p)$ will yield a computation that breaks down. We typically keep the size of $r_1\cdots r_n$ around $\sqrt{N}$, so it is at least the expected size of $\mathcal{C}_f(\mathbb{Z}_p)$ for p the smallest prime dividing N, and vary the function f.