FFT-ECM by Division Polynomials for Factoring

Zhihong Li & Samuel S. Wagstaff, Jr.

Introduction
Factoring large numbers is very useful in cryptography. One can break certain ciphers, such as the RSA cryptosystem, if one can factor large numbers. In this poster we develop a new integer factoring algorithm similar to the ECM (elliptic curve method). The difference is that this algorithm uses division polynomials and an FFT (fast Fourier transform) to compute multiples of many points simultaneously. The ordinary ECM has little chance of factoring an RSA public key and breaking the cipher. This algorithm has much greater chance of factoring a number of the size of RSA public keys currently used.

Review of Elliptic Curves and the ordinary ECM:
An elliptic curve is the graph of an equation where are real numbers, rational numbers or integers modulo m. The set also contains a point at infinity, denoted . The point is not a point on the graph of . It is the identity of the elliptic curve group. If lies on the graph of , we define . Given two points and on the graph but not on the same vertical line, define where is the third point on the straight line through and . If the tangent line through is vertical, then and . We can prove that an elliptic curve, with group operation defined above, is a group.

In 1985, H.W. Lenstra, Jr. invented an ingenious new factoring algorithm that uses elliptic curves. It performs a calculation where is the product of all primes less than some bound raised to some suitable power. One then computes and hopes that will equal the identity of modulo , which is a prime factor of the integer we need to factor, but will not equal the identity of modulo , which is the integer we need to factor.

The FFT-ECM by Division Polynomials:
We create a new, much faster ECM using division polynomials and fast Fourier transforms. First of all, let’s define division polynomials.

Def. (of Division Polynomials)
Define division polynomials inductively as follows:

Further define

Prop:
Suppose is an elliptic curve given by with , then the multiple of is given by

The algorithm relies on the following observations:
1. Pick a random pair of integers , then for any chosen , let . This yields univariate polynomials in the variable . I proved that the degree of is at most , and

2. The ECM finds the prime factor of when computing if and only if for some

3. One may evaluate a polynomial at all terms of a geometric progression by using discrete Fourier transforms to compute the convolution of two polynomials, and this may be done swiftly by FFT. In fact, such an FFT takes only arithmetic steps without division.

Sketch of the Algorithm:
To find a prime factor of , set and computing . Pick compute and check that is non-trivial. If the gcd is non-trivial, then we are done. If the gcd is then partition the product into subproducts and compute individual gcd’s. If the gcd is 1, then we should pick up another and B. In this algorithm, we try curves in a time.

The Algorithm for Evaluation of Polynomial on Geometric Progression:
We want to evaluate the polynomial at values . is a constant (in the algorithm here, ). We transform the sums according to

Where if and only if we avoid calculating by FFTs.

Output: The values for .

Choose with least such that .

For (mod )

Zero-pad to have length .

Let .

Perform the length-C cyclic convolution by FFTs.

Return .

The convolution in the penultimate line of the algorithm may be computed with discrete Fourier Transforms as

Where * is pointwise multiplication of vectors. The DFTs are performed efficiently by FFTs.

Significance of the method:
This new factoring method we are developing will be able to find much larger prime factors of large integers than the ordinary ECM can. The ordinary ECM often finds 40-digit factors and sometimes finds 50-digit factors of large integers. Its record discovery is a 57-digit prime factor. A heuristic argument similar to that used to estimate the complexity of the ordinary ECM suggests that our new method will easily find 60-digit factors, will often find 75-digit factors, and occasionally find 90-digit factors.

Several important ciphers, such as those of RSA and of Rabin and Williams, could be broken if one could factor large integers quickly. Each discovery of a new integer factoring algorithm either increases the size of the keys that must be used in these ciphers or imposes some other restriction on them. In the case of the new factoring method proposed here, the restriction is that the composite key must not have a prime factor in the range that could be found by this method.

Our new factoring method would also be an enormous help to those who research the mathematical problems that require the explicit factorization into primes of many large numbers.