# GOOD POLYNOMIALS FOR THE NUMBER FIELD SIEVE

**Jason E. Gower, Chaogui Zhang, and Samuel S. Wagstaff, Jr.**

**Center for Education and Research in Information Assurance and Security**

## INTRODUCTION

The security of many publickey cryptosystems such as RSA relies on the difficulty of factoring large numbers. It follows that one measure of the security of such cryptosystems is the run - time complexity of the fastest known factoring algorithm. Currently, this algorithm is the number field sieve (NFS).

Most steps in NFS are well understood except the first in which one chooses a polynomial. Since the run-time of NFS is largely determined by the choice of polynomial, our best hope for reducing the run-time of NFS seems to be in devising a method for choosing a good polynomial. Our work aims to better understand what makes for a good polynomial and how to find such polynomials.

## POLYNOMIALS USED IN NFS

Let $N$ be a number that one wishes to factor. The first step of NFS is to find a polynomial $f$ of specified degree $d$ with integer coefficients and an integer $m$ such that $m$ is a root of $f$ modulo $N$. The traditional method for doing this is the base -$m$ method:

Choose an integer $m$ such that $N^{(d+1)^{-1}} \le m \le N^{1/d}$ and write $N$ to the base $m$:

$$N = c_d m^d + c_{d-1} m^{d-1} + \ldots + c_1 m + c_0$$
$$0 \le c_i < m \text{ for } i = 0, 1, \ldots, d.$$

The polynomial $f$ can now be taken to be:

$$f = c_d X^d + c_{d-1} X^{d-1} + \ldots + c_1 X + c_0$$

with root $m$ modulo $N$. Finally, we construct the homogeneous polynomials:

$$F_1(X, Y) = Y^d f(X/Y),$$
$$F_2(X, Y) = X - mY.$$

## PROPERTIES OF GOOD POLYNOMIALS

The rate determining step in NFS is the search for smooth integers; that is, integers with no large prime factors. In this case, the integers that are examined for smoothness are the values of $F_1(a, b)$ and $F_2(a, b)$, for specified values of $a$ and $b$. If we define the *yield* of $F_1$, $F_2$ to be the number of smooth values of $F_1, F_2$, for a given smoothness bound and range for $a$ and $b$, then our search for good polynomials is reduced to finding $F_1$ and $F_2$ with high *yield*.

Let $C$ be the largest coefficient of $f$ in absolute value and define the *size* of the pair $F_1, F_2$ to be $m/C$. It is well known that size affects yield. Furthermore, root properties play an important role in determining yield. In particular, we consider roots of $F_1$ modulo $N$. The pair $(a, b)$ is a *projective root* of $F_1$ modulo a prime $p$ if $p$ divides both $F_1(a, b)$. Any other pair $(a, b)$ which is a root of $F_1$ modulo $p$ is a *nonprojective root*. Finally, nonprojective roots $(a, b)$ with $p$ dividing $a$ are called *zero roots*. We would like to find those $F_1$ which have many projective and nonprojective roots.

## FINDING POLYNOMIALS WITH SMALL SIZE

First, we consider how the output of the base-$m$ method varies as we change the base $m$. To do this, suppose we are given a polynomial $f$. Replace every occurrence of $X$ in $f$ with $X - Y$:

$$f(X - Y) = c_d(X - Y)^d + \ldots + c_1(X - Y) + c_0.$$

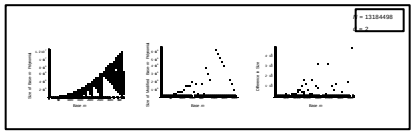
Now expand in terms of $X$:

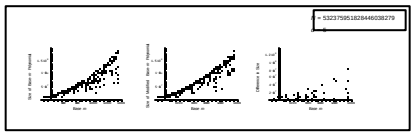
$$= c'_d X^d + \ldots + c'_1 X + c'_0,$$

where $c'_i$ is a function of the $c_i$ and $Y$.

Each solution $(x, y)$ to the equation $X = m + Y$, where $x$ and $y$ are integers, gives rise to a polynomial with root $x$ modulo $N$. The coefficients of this new polynomial are functions of the old coefficients $c_i$ and $y$. We seek a choice of $y$ that minimizes the size $(y + m)/C'$ of this new polynomial, where $x = y + m$ is the root modulo $N$ and $C'$ is the maximum of the $|c'_i|$ in absolute value. Unfortunately, the size of polynomials generated by the base-$m$ method is not easily predicted. However we note that $(y + m)^d C' = (y + m)^d |c'_d| + \ldots + |c'_1| + |c'_0|$. For fixed $c_i$, this upper bound is a differentiable function of $y$, so we may use standard Calculus techniques to minimize this function and produce a $y$ that minimizes the upper bound. Clearly then, $(y + m)^d C' = (y + m)^d (|c'_d| + \ldots + |c'_1| + |c'_0|)$ is by making the upper bound small, we can force the size of the polynomial to be small. Note that if $y = 0$ is the value that minimizes the upper bound, then we have $(y + m)^d C' = mC$. So in general the size of the new polynomial will be no greater than the original polynomial $g$ generated by the base-$m$ method. It is also important to note that $c'_d = c_d$. That is, the modified base-$m$ method has no effect on the leading coefficient of the polynomial generated with the base -$m$ method.

We refer to the procedure that we have developed above as the *modified base -m* method. Our interest is in how much smaller we can expect modified base -$m$ polynomials to be when compared to polynomials generated by the base-$m$ method. To this end, a simple Mathematica program which implements both the base-$m$ and modified base-$m$ methods was run for different values of $N$ and $d$, with $N^{d(d+1)^{-1}} < m < N^{1/d}$. What follows are some typical graphs of the difference in the size of the polynomials generated by the base-$m$ method and the modified base-$m$ method versus $m$.




## ROOT PROPERTIES

Now we turn to the problem of finding polynomials that have good root properties. Note that $F_1$ will have a projective root modulo $p$ if and only if $p$ divides the leading coefficient $c_d$ of $f$. Also note that if $a$ is a positive integer and $N\phi + 1 > m^d = Nb$, then the polynomial produced by the base-$m$ method will have leading coefficient $c_d = a$. So by suitably restricting the range for $m$, we can force the polynomials produced by the modified base-$m$ method to have a predetermined set of projective roots.

At this point we have a way of finding polynomials of small size with many projective roots. To force these polynomials to have non-projective roots will take some extra effort. Others have suggested a *rotation* of $f$ as a possible means of gaining nonprojective roots. A rotation of $f$ is simply $f + (X - m)\cdot g$ for a suitable choice of polynomial $g$. The rotation will still have the root $m$ modulo $N$ and, through a judicious choice of $g$ may also have many non-projective roots in addition to any projective roots of $f$.

We describe our method for finding $g$. We start by noting that $F_1$ will have a zero root modulo $p$ if and only if $p$ divides $c_0$, the constant term of $f$. Now we factor $c_0 = pr_1 \cdot \ldots \cdot r_k = B$, where $B$ is a measure of how much work we are willing to do. Once $j$ is found so that $r_{j+1}$ has many small prime factors we replace $f$ with $f - (X - m)\cdot j$. This new polynomial has $c_0 = r_1 \cdot \ldots \cdot r_j$ as its constant term and thus has many zero roots. Now suppose $f$ is a polynomial that already has many projective and zero roots. We can now replace $f$ by $f + (a_1 X + b_1)\cdot(X - m)$ and get a polynomial with all the projective and zero roots of $f$. Finally, we examine many values for $a$ and $b$ to see which gives the most non-zero non-projective roots. Of course, all of this work may alter the size of our polynomial, but the tradeoff may be worth the effort.

## FUTURE WORK

- Local minima of $c_d X^d + \ldots + c_1 X + c_0$ need not correspond in any way to those values that minimize $C$. We are currently looking for other functions of the $c_i$ that more sharply bound $C$ that could replace $c_d X^d + \ldots + c_1 X + c_0$ in the modified base -$m$ method.
- Experimentally we have found that as $d$ increases, the difference between the sizes of the outputs of the base -$m$ method and the modified base$m$ method decreases. This is likely due to the fact that the modified base$m$ method must minimize all $d+2$ coefficients, $c'_i$, and the base$y + m$. As we saw, forcing the polynomials to have many non-projective roots can come at the cost of larger low order coefficients but as it turns out, it is more important for the high order coefficients to be small. This suggests that we should minimize $(y + m)^{d'} |w_d c'_d| + \ldots + w_1 |c'_1| + w_0 |c'_0|$ in the modified base -$m$ method, for suitable weights $w_i$.
- Non-projective roots that are not zero roots depend upon all the coefficients of $f$, which makes them difficult to characterize. We hope to find a relationship between the rotation polynomial $g$ and the non -zero non -projective roots of the rotated polynomial that can be exploited without destroying projective and zero root properties.
- We would like to construct a norm on polynomials that measures yield. We could then conduct sieving experiments to compare polynomials of varying size and root properties. Ultimately we would like to use our methods to find good polynomials that we could then compare with other polynomials being used to factor large numbers such as RSA155.