A method to solve cyclotomic norm equations $f * \bar{f}$

Nick Howgrave-Graham[1] and Mike Szydlo[2]

[1] NTRU Cryptosystems, Burlington, MA, USA
nhowgravegraham@ntru.com
[2] RSA Laboratories, Burlington, MA, USA
mszydlo@rsasecurity.com

Abstract. We present a technique to recover $f \in \mathbb{Q}(\zeta_p)$ where ζ_p is a primitive p_{th} root of unity for a prime p, given its norm $g = f * \bar{f}$ in the totally real field $\mathbb{Q}(\zeta_p + \zeta_p^{-1})$. The classical method of solving this problem involves finding generators of principal ideals by enumerating the whole class group associated with $\mathbb{Q}(\zeta_p)$, but this approach quickly becomes infeasible as p increases. The apparent hardness of this problem has led several authors to suggest the problem as one suitable for cryptography. We describe a technique which avoids enumerating the class group, and instead recovers f by factoring N_f, the absolute norm of f, (for example with a subexponential sieve algorithm), and then running the Gentry-Szydlo polynomial time algorithm for a number of candidates. The algorithm has been tested with an implementation in PARI.

1 Introduction

We present an algorithm to solve degree two norm equations corresponding to the field extention $\mathbb{Q}(\zeta_p)$ / $\mathbb{Q}(\zeta_p + \zeta_p^{-1})$, where ζ_p is a p_{th} primitive root of unity.

The previously best known technique for solving this problem involves finding a principal ideal generator of the ideal of (f) by enumerating representatives of the whole class group, and then applying index calculus techniques to obtain a generator of (f). This approach is explained in a little more detail in section 2, but we note it becomes very expensive as p increases (even the class group is on the order of p^p).

In [6, 12, 13] the authors explicitly assume the hardness of the cyclotomic norm equation problem to build cryptographic applications. It has been observed by several sources [5, 8, 10] that their constructions can easily be modified to ones which are not reliant on this particular assumption, for example by adding some kind of perturbation or noise factor. Such enhancements are interesting avenues for further cryptographic research, but in this paper we concentrate on the purely mathematical problem of solving the norm equation.

Our work is motivated by a concrete question concerning polynomial arithmetic in the ring $R = \mathbb{Z}[X]/(X^p - 1)$. To describe our problem, we define, for each polynomial $f \in R$, its *reversal*, f_{rev}, to be the polynomial $f(X^{-1})$. An

element in R which is equal to its own reversal is called a *palindrome*. Given a palindromic element in R of the form $g = f * f_{rev}$, the task is to find such an f.

Our algorithm builds on an earlier work [9] which is able to recover an element $f \in \mathbb{Z}[X]/(X^p - 1)$, up to multiplication by $\pm X^k$ from the principal ideal (f), and from the quantity $f * f_{rev}$.

This problem of "factoring" $f * f_{rev}$ is easily seen to be related to a norm equation as follows. Elements of R naturally map to the quotient $\mathbb{Z}[X]/(1 + X + \ldots X^{p-1})$, the ring of integers in the p_{th} cyclotomic field $\mathbb{Q}(\zeta_p)$. Under this map, a polynomial and its reversal map to Galois conjugates in the field extension $\mathbb{Q}(\zeta_p)$ / $\mathbb{Q}(\zeta_p + \zeta_p^{-1})$, and the product of the conjugates is the norm in $\mathbb{Q}(\zeta_p + \zeta_p^{-1})$. Solving such a norm equation yields a cyclotomic integer, and knowledge of $f(1)$ is sufficient to determine f as an element of $\mathbb{Z}[X]/(X^p - 1)$. The integer $f(1)$, in turn, is known up to sign since $f * f_{rev}(1) = f^2(1)$.

Some authors call g, the *autocorrelation* of f, and work more generally in the ring $\mathbb{R}[X]/(X^p - 1)$. In [6, 7] instances of the problem are considered where the coefficients of f are taken to all be in $\{0, 1\}$, and the problem of recovering an f from the product $f * f_{rev}$ is called *bit retrieval*.

In this paper, unless otherwise noted, polynomials are elements of the ring R. When it is clear, we use the same symbol for the projection to the ring of cyclotomic integers.

2 The classical approach to solving this problem

The general principal ideal problem is, given an ideal $\mathcal{I}$ which is known to be principal, to find a generator of $\mathcal{I}$.

The standard technique for solving the principal ideal problem involves constructing a huge "factor base" consisting of ideals with small norm, representing every element of the class group.

The collection of a similarly large number of relations (which are "smooth" over the factor base), will yield a generator for the target ideal. See [3], section 6.5.5 for more details of this approach.

The class number of cyclotomic fields is on the order of p^p, see [4] for a more detailed analysis of this distribution. As a result of this, the factor base approach becomes infeasible very quickly, and had lead various people to suggest it as a basis for cryptographic schemes, see [2, 6, 12, 13].

3 Statement of the problem

Problem 1 *Let p be prime and let $f \in R = \mathbb{Z}[X]/(X^p - 1)$. Let $g = f * f_{rev}$. Given g, determine such an f.*

Clearly, the problem does not have a unique solution; if f is a solution, then so is $\pm X^k f$. To understand precisely the ambiguity of f, let us consider several ways in which we might enlarge the set of acceptable solutions. Below, f is considered as a cyclotomic integer, and $\bar{f}$ denotes complex conjugation.

Variant 1 *Find fractional ideals F such that $F * \bar{F} = (g)$.*

For an ideal F which is a solution, the ideal FA will also be a solution if $A\bar{A} = (1)$. Each ideal of the form $B/\bar{B}$ has this property, so there will be infinitely many solutions.

Variant 2 *Find integral ideals F such that $F * \bar{F} = (g)$.*

To restrict solutions to integral ideals, let $F_1^{a_1}.....F_k^{a_k}$ be the prime factorization of (f). Then for any $0 \leq b_i \leq a_i$, the ideal

$$F_1^{b_1}\bar{F_1}^{(a_1-b_1)} \cdots F_k^{b_k}\bar{F_k}^{(a_k-b_k)}$$

will be a solution. So there will be exactly $\prod(a_i + 1)$ solutions in total.

Variant 3 *Find cyclotomic field elements f such that $f * \bar{f} = g$.*

To restrict the solutions of variant 1, which are ideals, to those which are principally generated. If a field element f is a solution, we see that the element fa will also be a solution, provided that $a\bar{a} = 1$. Such a, are simply the elements of the form $a = b/\bar{b}$, and so if there is one solution, there will also be infinitely many solutions.

Variant 4 *Find cyclotomic integers f such that $f * \bar{f} = g$.*

This is in fact the flavor of the problem we are most interested in. Restricting solutions to cyclotomic integers, we see that the generator f of each solution ideal of variant 2 will yield a solution for this fourth variant. For a solution f, the multiples fu which are also solutions correspond to the values of u which are $2p_{th}$ roots of unity.

Thus, there are $2p$ times the number of principal ideal solutions. In particular we obtain $2p\prod(a_i + 1)$ as a bound on the total. In the case that (f) is a prime ideal, there are exactly $2p$ solutions.

4 Algorithm overview

At a high level, there are two components to the algorithm. The first step computes candidate ideals F for the desired ideal (f). The second step consists of recovering f from the ideal (f), and the element $g = f * f_{rev}$. The second step can be accomplished with the result of [9], reviewed below in section 5.1, or in some cases, by a simplified version of it.

We illustrate the algorithm first in a somewhat special case, then proceed to show how it can be extended. In this first case, we are going to assume that the prime p is congruent to 3 (mod 4), and additionally that the norm of f, down to $\mathbb{Z}$ does not have any repeated factors. Our first assumption implies that $\mathbb{Q}(\sqrt{-p})$ is a subfield of the p_{th} cyclotomic field, so that there is a diagram of field extentions.

$$\begin{array}{ccc} \mathbb{Q}(\zeta) & \leftarrow & \mathbb{Q}(\zeta + \zeta^{-1}) \\ \uparrow & & \uparrow \\ \mathbb{Q}(\sqrt{-p}) & \leftarrow & \mathbb{Q} \end{array}$$

The essence of the first step of the algorithm is the determination of a set of potential ideals for (f). As in the classical approach to our problem, this may be

accomplished by factoring $f * f_{rev}$ into prime ideals. The candidate ideals will be the ideals of R which have the same norm as f, and also contain $(f * f_{rev})$.

To factor an ideal in the ring cyclotomic integers, one appeals to the classical prime decomposition theory [1], using the prime factorization of the absolute norm. Algorithmically, beginning with $f*f_{rev}$ in $\mathbb{Q}(\zeta+\zeta^{-1})$, it is easy to compute the norm N_f down to $\mathbb{Q}$. Our assumption that each rational prime q divides this norm with multiplicity at most 1, implies that q splits completely in $\mathbb{Q}(\zeta)$. Thus, it makes the use of arithmetic in the quadratic subfield even simpler.

There are two primes above each such q in $\mathbb{Q}(\sqrt{-p})$. Among the various possible products, $\mathcal{I}$, of these ideals in $\mathbb{Q}(\sqrt{-p})$ are the norms of the solutions f. By lifting each such $\mathcal{I}$ to $\mathbb{Q}(\zeta)$, and computing the greatest common divisor with $(f * f_{rev})$, one obtains an ideal F, with the property that $F * F^{\sigma} = (f * f_{rev})$. This can be considered a solution to the problem variant 2, above, and will be deemed a candidate ideal.

The main algorithmic difficultly of determining the potential ideals in $\mathbb{Q}(\sqrt{-p})$ is the integer factorization of N_f in $\mathbb{Z}$. Once this has been accomplished the above procedure yields a short list of candidate ideals.

Once one is in possession of both (f) and $f * f_{rev}$, the algorithm [9] can be used to find f up to a root of 1. We provide further details in the next section.

5 Algorithm details for $p = 3 \bmod 4$

In this section we provide further details of each step.

Norm Calculation The initial step consists in calculating the norm of f down to $\mathbb{Q}$. This norm is the product $\prod f(\zeta_i)$ over all p_{th} primitive roots of 1. Grouping the factors into conjugate pairs, we see that this can be efficiently computed from the $(p-1)/2$ conjugates of $(f * f_{rev})(\zeta_i)$. Equivalently, the norm of f is the square root of the norm of $f * f_{rev}$ in $\mathbb{Z}$. Let N_f denote this integer.

Prime Factorization Next, the prime factors of N_f must be determined. Unless N_f turns out to be prime, or the factorization easy, this step is the most computationally difficult part of the algorithm, and prohibitive if p is large. Depending on size, a standard elliptic curve or sieving algorithm may be successful. Let $q_1^{a_1} \dots q_k^{a_k}$ denote the prime factors of N_f.

Ideals in $\mathbb{Q}(\sqrt{-p})$: Each prime in $\mathbb{Z}$ either splits or is inert in $\mathbb{Q}(\sqrt{-p})$. By making the assumption that N_f has no repeated factors, we ensure that each prime q_i, splits. The two factors are then $(q) = (q, r+\frac{1+\sqrt{-p}}{2})$, and $(q, r-\frac{1+\sqrt{-p}}{2})$.

A candidate ideal $\mathcal{I}$ in $\mathbb{Q}(\sqrt{-p})$ is computed by multiplying together exactly one ideal above each q_i. The ideal arithmetic in this step is particularly efficient, given that we are only working in a quadratic field.

Testing Principality

A necessary condition that the ideal $\mathcal{I}$ be the norm of a solution f, is that $\mathcal{I}$ is a principal ideal. If the class number of $\mathbb{Q}(\sqrt{-p})$ is not large, the ideal $\mathcal{I}$ can be efficiently tested for principality as in [3]. This may be feasible, since the class numbers of quadratic imaginary fields seem to grow at a reasonable rate. If it is concluded that $\mathcal{I}$ is not principal, the subsequent steps of our algorithm can

not succeed, so $\mathcal{I}$ may be discarded. We remark that this step is optional. If all of the potential ideals $\mathcal{I}$ were to be processed in parallel, eventually a solution would be found.

GCD computation

In this step we determine the candidate ideals F of $\mathbb{Q}(\zeta)$. Another consequence of assuming that N_f has no repeated factors is the fact that f and f_{rev} are relatively prime. Additionally, f_{rev} is prime to the other $(p-3)/2$ Galois conjugates of f dividing $\mathcal{I}$.

Now consider both $\mathcal{I}$ and $(f*f_{rev})$ as ideals of $\mathbb{Q}(\zeta)$, and compute the greatest common divisor ideal. This is efficiently accomplished by forming the $\mathbb{Z}$ span of generators of the two $\mathbb{Z}$-modules [3]. The resulting ideal is an ideal F, with the property that $F * \bar{F} = (g)$, and is thus a solution to variant 2 of our problem. Only if F is principal will it correspond to an element f such that $f * f_{rev} = g$.

Recovering f

Given the candidate ideal $\mathcal{I}$ and the product $f * f_{rev}$, apply the algorithm, reviewed below in section 5.1. Note that if $\mathcal{I}$ does not correspond to a solution, this process must fail. One method of dealing with this detail is to process all possible candidate ideals F in parallel.

5.1 Review of Gentry-Szydlo Algorithm

In this section we review the algorithm of [9], which recovers an element from $Z[X]/(X^p - 1)$ given the ideal it generates, and the product $f * f_{rev}$.

A simplified version of the algorithm is as follows. We will express ideals as $\mathbb{Z}$-modules over the power basis $\{1, X, X^2, \ldots\}$. For a p element vector f, the *circulant* matrix $Cir(f)$ is the matrix of all rotations of f, and the *columns* of $Cir(f)$ generate the principal ideal f in $\mathbb{Q}(\zeta)$.

First, note that if F denotes $Cir(f)$, then for any other basis H of (f), $H = FU$ for some unimodular matrix. Next, we combine this information with $D = Cir(ff_{rev})$ by forming the product

$$\begin{aligned} G &= H^t D^{-1} H \\ &= U^t F^t (F f^t)^{-1} F U \\ &= U^t U. \end{aligned}$$

This last matrix can be viewed as the Gram matrix of an auxiliary lattice. The auxiliary lattice might be called a *hypercubic* lattice [19] since it is a rotation of the trivial lattice $\mathbb{Z}^p$. Moreover the paper [19] suggests that it may be easier to reduce hypercubic lattices than general lattices of a similar dimension.

This lattice, expressed via the Gram matrix G, may be reduced with LLL, or one of its variants, producing a unimodular matrix V such that $V^t G V$ is the more reduced lattice. If very successful (perhaps because of the easier problem the hypercubic lattices pose) the lattice reduction might reduce the Gram matrix right down to the Identity matrix (e.g $V = U^{-1}$ would). In such a case,

$V^t U^t U V = Id$, implies that $W = UV$ is a signed permutation matrix, and multiplying the matrix FU by V yields FW, whose columns are all signed rotations of the sought vector, f.

Thus, provided that the lattice reduction algorithm manages to fully reduce G to the identity, f can be recovered. However, the variants of LLL, which are guaranteed to produce a shortest vector, are exponential time algorithms. In practice, LLL, and its higher block-size variants [17] often produce much shorter vectors than the available bounds.

The approach taken in the more complicated algorithm in [9] provides a strategy that will guarantee that we find the shortest vector in polynomial time, even though LLL only returns a *multiple* of the shortest vector in polynomial time. The technical trick is to attempt to reduce the ideal F^{R-1}, for a prime R congruent to 1 mod p, in such a way that an element $f^{R-1}\alpha$ is produced where α has small L_2 norm. The congruence $f^{R-1}\alpha \cong \alpha \bmod R$ can be used to find α when it is small compared to R. A final calculation finds f from f^{R-1}.

6 General case

The assumptions made in the previous section, were convenient but not essential. We now explain how to extend the algorithm to the general case, where we do not assume that $p = 3 \bmod 4$ or that the norm does not have repeated prime factors.

We suppose that the cyclotomic norm, N_f, factors as $q_1^{a_1}....q_k^{a_k}$, and wish to determine a list of candidate ideals F for (f).

Fix a prime q. Then the factorization of q into prime ideals in $\mathbb{Q}(\zeta_p)$ may be efficiently computed by factoring the polynomial $1 + X + X^2 + \ldots + X^{p-1}$ over $\mathbb{F}_q$ using either the Berlekamp or Cantor-Zassenhaus algorithms (see [3], section 3.4 for more details). (The number of prime ideals above q depends on the order of $q \bmod p$.). For each prime ideal Q above q, the exact exponent of Q dividing $(f * \bar{f})$ may be computed via ideal divisibility tests. Thus, given the factorization of the norm N_f, the factorization of $(f * \bar{f})$ into ideals may be efficiently computed.

Now consider a prime ideal Q, and let r be the largest power of Q dividing g. For each Q, let $\bar{Q}$ be the complex conjugate ideal. In the case that $\bar{Q} = Q$, the power of Q dividing (f), is clearly $r/2$. When $\bar{Q} \neq Q$, then we know that the maximal power of Q, a_Q say, that divides (f) must be in $\{0, \ldots, r\}$, and moreover $a_{\bar{Q}} + a_Q = r$. This leaves $r + 1$ possible candidates to enumerate over.

Considering all prime ideal pairs $Q_i \neq \bar{Q}_i$ such that $Q^{r_i} | (f * \bar{f})$, we see that the total number of candidate ideals for (F) is $\prod(r_i + 1)$.

Once the list of candidate ideals F has been established, the algorithm may proceed as described above. The only difference is that one does not have the simplicity and efficiency gain obtained by working in the quadratic imaginary subfield.

7 Experiments

The norm, N_f, of f down to $\mathbb{Z}$, can be calculated as $1/f(1)$ times the determinant of an associated cyclic matrix, and thus can be bounded by $|f|^p$, where $|\cdot|$ denotes the Euclidean norm of the entries of f. In experiments both the number of factors and the multiplicity of these factors followed what one would expect from the Prime Number Theorem, which leads us to conjecture that the prime factorizations of the norms are distributed in a similar fashion to randomly sampled numbers with a similar bit length.

In obtaining practical results, we implemented just the $p = 3 \bmod 4$ case in PARI.

7.1 Recovering f when $p = 3 \bmod 4$

For small values of p, congruent to 3 mod 4, the entire algorithm, as described above was implemented in PARI [11]. This symbolic and computational calculator has the advantage of being able to apply the LLL algorithm to a lattice expressed via a Gram matrix, which the (otherwise flexible) NTL [18] LLL implementation does not allow for.

The arithmetic in $\mathbb{Q}(\sqrt{-p})$ was used to create candidate ideals. Our experiment also implemented the simplified version of the Gentry-Szydlo algorithm described above, which was sufficient for the primes tested. That is, the original LLL algorithm was able to reduce the Gram matrix to the identity, in the cases that yielded a solution.

Result were (quickly) obtained with p varying from 19 to 71, with the fortunate outcome that for these smallish primes, the simplified Gentry-Szydlo approach was always sufficient to completely reduce the lattice[1]. As expected, the results reflected the fact that not all candidate ideals must yield a solution for f, e.g. there were many instances where the norm had exactly two prime factors, and only two of the four candidates produced a valid f.

8 Conclusions

We have shown how the difficulty of factoring $f * \bar{f}$ into elements reduces to the problem of factoring an integer N_f, and the problem of applying the (polynomial time) Gentry-Szydlo algorithm to a number of candidates. Under the assumption that the factorization of the norms N_f are distributed in a similar fashion to average numbers of a similar size, then the bottleneck of the algorithm occurs in the integer factorization stage (since the number of candidates is exponential in the number of prime divisors of N_f but this is logarithmic in p).

In particular this paper shows that f such that $|f|$ is small, where $|\cdot|$ denotes the Euclidean norm of the entries of f, are particularly weak, since the integer N_f is particularly small in these cases.

[1] This is consistent with the conjecture that hypercubic lattice may indeed be an easier class of lattice to reduce.

We remark that if one is trying to build a cryptosystem on the basis of the hardness of factoring $f * \bar{f}$, then p should be chosen such that it is highly unlikely that the norm N_f can be factored in a reasonable time.

9 Acknowledgments

The authors wish to thank Veit Elser, Craig Gentry and Jeff Hoffstein for useful correspondence.

References

1. Z. I. Borevich and I. R. Shafarevich *Number Theory*, Academic Press, 1966.
2. J. Buchmann, M. Maurer and B. Möller, *Cryptography based on number fields with large regulator*. Journal de Théorie des Nombres de Bordeaux, 2000, pp. 293–307
3. H. Cohen, *A Course in Computational Algebraic Number Theory*, Graduate Texts in Mathematics, 138. Springer, 1993.
4. H. Cohen and H. Lenstra, *Heuristics on class groups of number fields*, Number Theory, Lecture Notes in Mathematics, vol. 1068, Springer-Verlag, 1983, pp. 33–62.
5. V. Elser, *Private Communication.*
6. V. Elser, *Bit retrieval: intractability and application to digital watermarking*, `http://arxiv.org/abs/math.NT/0309387`
7. V. Elser, *Phase retrieval challenges*, `http://www.cecm.sfu.ca/~veit/`
8. C. Gentry, *Private Communication.*
9. C. Gentry, M. Szydlo, *Cryptanalysis of the Revised NTRU signature scheme*, in Proc. of Eurocrypt '02, LNCS 2332, pages 299–320. Springer-Verlag, 2002.
10. J. Hoffstein, *Private Communication.*
11. PARI, `http://pari.math.u`bordeaux.fr/-
12. J. Hoffstein, D. Lieman, J.H. Silverman, *Polynomial Rings and Efficient Public Key Authentication*, in Proc. International Workshop on Cryptographic Techniques and E-Commerce (CrypTEC '99), Hong Kong, (M. Blum and C.H. Lee, eds.), City University of Hong Kong Press.
13. J. Hoffstein, J.H. Silverman, *Polynomial Rings and Efficient Public Key Authentication II*, in Proceedings of a Conference on Cryptography and Number Theory (CCNT '99), Birkhauser.
14. A.K. Lenstra, H.W. Lenstra Jr., L. Lovász, *Factoring Polynomials with Rational Coefficients*, Mathematische Ann. 261 (1982), 513–534.
15. D. Micciancio, *The Shortest Vector in a Lattice is Hard to Approximate to within Some Constant*, in Proc. 39th Symposium on Foundations of Computer Science, 1998, 92–98.
16. P. Nguyen and J. Stern, *Lattice Reduction in Cryptology: An Update*, in Proc. of Algorithm Number Theory (ANTS IV), LNCS 1838, pages 85–112. Springer-Verlag, 2000.
17. C.-P. Schnorr, *A Hierarchy of Polynomial Time Lattice Basis Reduction Algorithms*, Theoretical Computer Science 53 (1987), 201–224.
18. Shoup, V., NTL: A Library for Doing Number Theory. Available at `http://www.shoup.net/ntl/`.
19. Szydlo, Michael, *Hypercubic Lattice Reduction*, Eurocrypt '03
20. L. Washington, *Introduction to Cyclotomic Fields*, Graduate Texts in Mathematics 83, 1982.