On the Deterministic Complexity of Factoring Polynomials over Finite Fields

Victor Shoup

Computer Sciences Technical Report #782

July 1988

On the Deterministic Complexity of Factoring Polynomials over Finite Fields

Victor Shoup*
Computer Sciences Department
University of Wisconsin–Madison
Madison, WI 53706

July 21, 1988

1. Summary

In this paper, we present some new deterministic algorithms for factoring polynomials over finite fields that are asymptotically faster than many commonly known deterministic factoring algorithms.

First, some notation. Let $\log x = \log_2 x$, and $L(x) = \log x \log\log x$. To suppress logarithmic factors, we will occasionally use the expression x^ϵ to denote a fixed, but unspecified, polynomial in $\log x$.

Let p be a prime number. Our main result is a deterministic algorithm that factors polynomials in $\mathbf{Z}_p[X]$ of degree n using $O(p^{1/2+\epsilon}n^{2+\epsilon})$ operations $(+, -, \times, /)$ in $\mathbf{Z}_p$. This improves upon the running time, with respect to both n and p, of many previously known deterministic factoring algorithms.

Using Berlekamp's deterministic algorithm (Berlekamp [1968, ch. 6]) in conjunction with fast algorithms for polynomial arithmetic, polynomials over $\mathbf{Z}_p$ of degree n can be factored deterministically with $O(n^3 + pn^{2+\epsilon})$ operations in $\mathbf{Z}_p$. Our algorithm improves upon Berlekamp's method by a factor of nearly n by avoiding a null space computation, and by a factor of nearly $p^{1/2}$ by proving that a brute-force search through all of $\mathbf{Z}_p$ is not necessary.

Camion [1983a] discusses some factoring methods, but the worst case running time of Camion's deterministic factoring algorithm is no better than that of Berlekamp's.

Ben-Or [1981] and Cantor and Zassenhaus [1981] give *probabilistic* algorithms that use $O((\log p)n^{2+\epsilon})$ *expected* operations in $\mathbf{Z}_p$. The use of randomness in these algorithm is essential in improving upon the running time, with respect to n as well as p, of Berlekamp's deterministic

* This research was supported by NSF grants DCR-8504485 and DCR-8552596

algorithm. At least for small values of p, the running time our algorithm is competitive with that of these probabilistic algorithms, even though ours avoids the use of randomness.

Other work on factoring polynomials over finite fields includes Berlekamp [1970], Camion [1983b], Lazard [1982], McEliece [1969], Rabin [1980], and Ronyai [1987].

In addition to a general factoring algorithm, we present two algorithms that are of use in special situations. Suppose we are given a polynomial over $\mathbf{Z}_p$ of degree n that is the product of r distinct monic irreducible polynomials of degree d ($n = dr$). Then we can extract a single irreducible factor of this polynomial with $O((\log p)d^{1+\epsilon}n^{1+\epsilon} + p^{1/2+\epsilon}n^{1+\epsilon})$ operations in $\mathbf{Z}_p$, and we can extract *all* of the irreducible factors with $O(p^{1/2+\epsilon}d^{1+\epsilon}n^{1+\epsilon})$ operations in $\mathbf{Z}_p$.

2. New Factoring Methods

We will make use of the following results concerning the complexity of performing arithmetic operations on polynomials.

Lemma 1. *Let R be a commutative ring with unity such that either $2^{-1} \in R$ or $3^{-1} \in R$. Let F be a finite field.*

(1) Multiplication of two polynomials in $R[X]$ of degree $\leq n$ can be performed using $O(nL(n))$ operations $(+, -, \times$ only) in R.

(2) Let $\alpha_1, \ldots, \alpha_n \in R$. Then the coefficients of $(X - \alpha_1) \cdots (X - \alpha_n) \in R[X]$ can be computed using $O(nL(n)(\log n))$ operations $(+, -, \times$ only) in R.

(3) Let f and g be polynomials in $F[X]$ of degree $\leq n$, $g \neq 0$. Then $f \bmod g$ can be computed using $O(nL(n))$ operations in F.

(4) Let $f, g_1, \ldots, g_k$ be polynomials in $F[X]$ such that $\deg f \leq n$ and $\deg g_1 + \cdots + \deg g_k \leq n$. Then $f \bmod g_1, \ldots, f \bmod g_k$ can be computed using $O(nL(n)(\log k))$ operations in F.

(5) Let f and g be polynomials in $F[X]$ of degree $\leq n$. Then the greatest common divisor of f and g can be computed using $O(nL(n)(\log n))$ operations in F.

(1) follows from Schönhage [1977]. See also Cantor and Kaltofen [1987] for a more general multiplication algorithm that works over *any* ring R. (2) follows from (1) by a divide and conquer method (see Borodin and Munro [1975, p. 100]). (3) follows from (1) by a Newton iteration scheme (see Borodin and Munro [1975, p. 95]). (4) follows from (3) by a divide and conquer method (see Borodin and Munro [1975, p. 100]). (5) follows from (1) by an algorithm described in Aho, Hopcroft and Ullman [1974, pp. 303-308].

We will also make use of the following number theoretic lemma.

Lemma 2. *Let p be an odd prime, and let $a, b \in \mathbf{Z}_p$, such that $a \neq b$ and*

$$\chi(ab) = \chi((a+1)(b+1)) = \cdots = \chi((a+M)(b+M)) = 1,$$

where χ is the quadratic character on $\mathbf{Z}_p$. Then $M < p^{1/2} \log p$.

Proof. Let $k = \lceil \frac{1}{2} \log p \rceil$. Let N be the number of solutions $(x, y_0, \ldots, y_{k-1})$ in $\mathbf{Z}_p^{k+1}$ to the system of equations

$$(x+a+i)(x+b+i) = y_i^2 \qquad (i = 0, \ldots, k-1).$$

We will first show that

$$N \leq p + p^{1/2}(2^k(k-1)+1). \tag{1}$$

Now, for fixed $c \in \mathbf{Z}_p$ the number of $y \in \mathbf{Z}_p$ satisfying the equation $y^2 = c$ is precisely $1 + \chi(c)$. Therefore,

$$\begin{aligned} N &= \sum_{x \in \mathbf{Z}_p} \prod_{i=0}^{k-1} (1 + \chi((x+a+i)(x+b+i))) \\ &= \sum_{0 \leq e_0, \ldots, e_{k-1} \leq 1} \quad \sum_{x \in \mathbf{Z}_p} \chi\left(\prod_{i=0}^{k-1} (x+a+i)^{e_i} (x+b+i)^{e_i} \right). \end{aligned}$$

In this last expression, the term corresponding to $e_0 = \cdots = e_{k-1} = 0$ is p. To bound the other terms, we use the result that for any polynomial $h \in \mathbf{Z}_p[X]$ that has m distinct roots and is not a constant multiple of a perfect square, we have

$$\left| \sum_{x \in \mathbf{Z}_p} \chi(h(x)) \right| \leq (m-1)p^{1/2} \tag{2}$$

(see Schmidt [1976, p. 43]).

Now let $e_0, \ldots, e_{k-1}$ be fixed, and suppose that $l > 0$ of the e_i's are nonzero. Put $h(X) = \prod_{i=0}^{k-1} (X+a+i)^{e_i} (X+b+i)^{e_i}$.

To make use of (2), we must show that h is not a perfect square. Suppose that it were. Then for distinct $i_1, \ldots, i_l$ between 0 and $k-1$, we would have

$$a + i_1 = b + i_2, \quad a + i_2 = b + i_3, \quad \ldots, \quad a + i_{l-1} = b + i_l, \quad a + i_l = b + i_1.$$

Summing, we have $la + \sum_\nu i_\nu = lb + \sum_\nu i_\nu$. But this implies that $la = lb$, and since $0 < l < p$, we can cancel, obtaining $a = b$, a contradiction. Therefore, h is not a perfect square, and (2) holds.

We have

$$N \le p + p^{1/2} \sum_{l=1}^{k} \binom{k}{l} (2l-1)$$

$$= p + p^{1/2}(2^k(k-1)+1).$$

This proves (1).

Now, the number of $x \in \mathbf{Z}_p$ such that

$$\chi((x+a+i)(x+b+i)) = 1 \qquad (i = 0, \ldots, k-1)$$

is at most $N/2^k$. The worst possible case is when all such x are bunched together near zero. So we have $M < N/2^k + k$. By (1), we have $M < p/2^k + p^{1/2}(k - 1 + 2^{-k}) + k$. Since $k = \lceil \frac{1}{2} \log p \rceil$, we have $M < p^{1/2} + \frac{1}{2} p^{1/2} \log p + \frac{1}{2} \log p + 2$. The right hand side of this inequality is asymptotic to $\frac{1}{2} p^{1/2} \log p$, and is less than $p^{1/2} \log p$ for $p > 16$. For $p < 16$, $p^{1/2} \log p > p$, and so the lemma is trivially true. ■

Let $f \in \mathbf{Z}_p[X]$ be a polynomial of degree n that we wish to factor. As in Ben-Or [1981] and Cantor and Zassenhaus [1981], we first perform distinct degree factorization. That is, we construct polynomials $f^{(1)}, \ldots, f^{(n)}$ where $f^{(i)}$ is the product of all distinct monic irreducible polynomials of degree i that divide f. We can do this as follows. We first compute $X^{p^i} - X \bmod f$ for $i = 1, \ldots, n$. This can be done using $O((\log p) n^2 L(n))$ operations in $\mathbf{Z}_p$. Then, for $i = 1, \ldots, n$, we do the following: put $f^{(i)} = \gcd(f, X^{p^i} - X)$; while $\gcd(f, X^{p^i} - X) \neq 1$, replace f by $f / \gcd(f, X^{p^i} - X)$. This requires $O(n^2 L(n)(\log n))$ operations in $\mathbf{Z}_p$. Thus, the total number of operations in $\mathbf{Z}_p$ required to perform distinct degree factorization is $O(n^2 L(n)(\log n + \log p))$. The correctness of the algorithm follows immediately from the fact that $X^{p^i} - X$ is the product of all distinct monic irreducible polynomials over $\mathbf{Z}_p$ of degree $j \mid i$ (Ireland and Rosen [1982, p. 84]).

In the sequel, we assume that $f \in \mathbf{Z}_p[X]$ is a polynomial of degree n, and is the product of $r > 1$ distinct monic irreducible polynomials $f_1, \ldots, f_r$, each of degree d ($n = rd$). Let $R = \mathbf{Z}_p[X]/(f)$ and let $x = X \bmod f$ be the image of X in R. Let θ_i be the natural homomorphism from R onto $R_i = \mathbf{Z}_p[X]/(f_i)$.

The well-known *Berlekamp subalgebra* B of R is defined by $B = \{\alpha \in R : \alpha^p = \alpha\}$. Equivalently, we have $B = \{\alpha \in R : \theta_i(\alpha) \in \mathbf{Z}_p \text{ for each } i = 1, \ldots, r\}$. Following Camion [1983a], we call a subset $S \subset B$ a *separating set* if for any $1 \le i < j \le r$ there exists $s \in S$ such that $\theta_i(s) \neq \theta_j(s)$.

We consider the problem of constructing a separating set S. We first discuss two approaches used by others.

Berlekamp [1968, ch. 6] uses as a separating set a basis for the Berlekamp subalgebra B. This can be done by computing a basis for null space of the matrix of the $\mathbf{Z}_p$-linear map on R given by $\alpha \mapsto \alpha^p - \alpha$. The matrix can be constructed with $O((\log p)nL(n) + n^2L(n))$ operations in $\mathbf{Z}_p$. The null space of the matrix can be computed with $O(n^3)$ operations in $\mathbf{Z}_p$ using an algorithm described in Knuth [1981, p. 425].

Another way to construct a separating set is described in Camion [1983a]. For $\alpha \in R$, let $T(\alpha) = \alpha + \alpha^p + \cdots + \alpha^{p^{d-1}}$. Camion shows that $S = \{T(x), T(x^2), \ldots, T(x^{2d-1})\}$ is a separating set. S can be constructed with $O((\log p)dnL(n) + d^2nL(n))$ operations in $\mathbf{Z}_p$.

We now describe our method of constructing a separating set. We compute $g(Y) \in R[Y]$ where $g(Y) = (Y - x)(Y - x^p)\cdots(Y - x^{p^{d-1}})$. This can be done using $O((\log p)d)$ operations in R to compute the powers of x, and $O(dL(d)(\log d))$ operations in R to compute the coefficients of $g(Y)$. This gives a total of $O((\log p)dnL(n) + dL(d)(\log d)nL(n))$ operations in $\mathbf{Z}_p$.

Suppose $g(Y) = g_0 + g_1Y + \cdots + g_{d-1}Y^{d-1} + Y^d$, where the g_i's are in R. We claim that the set of coefficients $S = \{g_0, \ldots, g_{d-1}\}$ is a separating set. To prove this, we first show that $g^{\theta_i}(Y) = f_i(Y)$. To see this, let $x_i = \theta_i(x)$. Then x_i is a root of f_i, and the other roots of f_i in R_i are $x_i^p, \ldots, x_i^{p^{d-1}}$. But $g^{\theta_i}(Y) = (Y - x_i)(Y - x_i^p)\cdots(Y - x_i^{p^{d-1}}) = f_i(Y)$. Now, to show that the coefficients of $g(Y)$ form a separating set, let $1 \le i < j \le r$ be given. Then since f_i and f_j are distinct, one of their coefficients differ. Suppose that the coefficient a of X^k in f_i is different from the coefficient b of X^k in f_j. Then $\theta_i(g_k) = a$ and $\theta_j(g_k) = b$, where a and b lie in the ground field $\mathbf{Z}_p$, and $a \neq b$.

Given a separating set S, the usual way to deterministically factor f is, more or less, as follows. We construct finer and finer factorizations $U \subset \mathbf{Z}_p[X] - \mathbf{Z}_p$ consisting of monic polynomials with $\prod_{u \in U} u = f$. Initially, we put $U = \{f\}$. Then, for $z = 0, 1, \ldots, p-2$ and $s \in S$ we replace U by $Refine(U, s + z)$, where for any polynomial g, $Refine(U, g) = \bigcup_{u \in U}\{\gcd(u, g), u/\gcd(u, g)\} - \{1\}$. If $\deg g < n$, then $Refine(U, g)$ can be computed with $O(nL(n)(\log n))$ operations in $\mathbf{Z}_p$ by first computing $g \bmod u$ for each $u \in U$, and then computing $\gcd(g \bmod u, u)$ for each $u \in U$. Thus, f can be completely factored with $O(|S|pnL(n)(\log n))$ operations in $\mathbf{Z}_p$.

For small p, the method described above is adequate; however, for large p, the running time becomes prohibitive. We now describe our method for deterministically factoring f, given a separating set S, that runs faster for large values of p (we assume $p > 2$). Initially, we put $U = \{f\}$. Then, for $z = 0, 1, \ldots$, until $|U| = r$, we execute the following procedure: for each $s \in S$, replace U by $Refine(U, (s + z)^{(p-1)/2} - 1)$, and then replace U by $Refine(U, s + z)$.

For any fixed value of z, the cost of the above procedure is $O(|S|nL(n)(\log p+\log n))$ operations in $\mathbf{Z}_p$. To determine the total cost, we must obtain a bound on the number of different values of z which are tried before a complete factorization is of f is obtained.

Suppose that for some $1 \leq i < j \leq r$, the above procedure has been applied for $z = 0, \ldots, M$, and that $f_i f_j \mid u$ for some $u \in U$. Since S is a separating set, there is an $s \in S$ such that $\theta_i(s) = a$, $\theta_j(s) = b$, where a and b are distinct elements in $\mathbf{Z}_p$. It follows that $\chi((a+z)(b+z)) = 1$ for $z = 0, \ldots, M$, where χ is the quadratic character on $\mathbf{Z}_p$. By Lemma 2, we must have $M < p^{1/2}(\log p)$. Therefore, given a separating set S, we can deterministically factor f with $O(|S|p^{1/2}(\log p)nL(n)(\log p + \log n))$ operations in $\mathbf{Z}_p$.

In some applications, only a single irreducible factor of f is required. We describe a recursive method to do this efficiently given a factoring set S. First, we find an element $s \in S$ that is not a constant. Then, for $z = 0, 1, \ldots$ we compute $\gcd(f, s+z)$ and, if p is odd, $\gcd(f, (s+z)^{(p-1)/2})$ until one of these is a nontrivial divisor g of f. Finally, we replace f by g or f/g, choosing the one of smaller degree, reduce each element of S modulo this new value of f, and invoke the method recursively. Using this method, we can extract an irreducible factor of f using $O(|S|nL(n) + p^{1/2}(\log p)nL(n)(\log p + \log n))$ operations in $\mathbf{Z}_p$.

Combining our new method of constructing a separating set S with our new method of factoring f given S, we obtain

Theorem 1. *Let f be a polynomial over $\mathbf{Z}_p$ of degree n that is the product of r distinct irreducible monic polynomials of degree d ($n = dr$). Then a single irreducible factor of f can be extracted deterministically with*

$$O(dnL(n)(\log p + L(d)(\log d)) + p^{1/2}(\log p)nL(n)(\log p + \log n))$$

operations in $\mathbf{Z}_p$, and all of the factors of f can be extracted deterministically with

$$O(dL(d)(\log d)nL(n) + p^{1/2}(\log p)dnL(n)(\log p + \log n))$$

operations in $\mathbf{Z}_p$.

Combining Theorem 1 with the distinct degree factorization algorithm, we obtain

Theorem 2. *Polynomials over $\mathbf{Z}_p$ of degree n can be deterministically factored with*

$$O(n^2 L(n)^2(\log n) + p^{1/2}(\log p)n^2 L(n)(\log p + \log n))$$

operations in $\mathbf{Z}_p$.

We conclude with a couple of open questions.

Open Question 1. *Can the bound in Lemma 2 be significantly improved?*

Of course, if $M = O((\log p)^c)$ for some constant c, then factoring polynomials could be done deterministically in polynomial time. No one has been able to prove such a bound, even assuming the Extended Riemann Hypothesis.

In Bach and Shoup [1988, Theorem 3.2], it is shown that if $k = \lceil \log p \rceil$, then the number of pairs $(a, b) \in \mathbf{Z}_p^2$ such that $\chi(ab) = \chi((a+1)(b+1)) = \cdots = \chi((a+k-1)(b+k-1)) = 1$ is at most $p(\log p)^2$. This shows that $M \leq \log p$ for the vast majority of possible choices of a and b.

Burgess [1963] has shown that for any *fixed* nonlinear polynomial $t(X)$ with integer coefficients that is the product of rational linear factors and is not a perfect square, the maximum number of consecutive integers x having the same value of $\chi(t(x))$ is $O(p^{1/4}(\log p))$. Unfortunately, the constant implicit in the big-O depends in a fairly horrible way on $t(X)$. For this reason, Burgess' result does not seem to help us in answering our question.

Berlekamp [1970, p. 732] observes that the maximum number of consecutive quadratic residues or nonresidues mod p is $O(p^{1/4}(\log p)^{3/2})$. This observation, however, does not by itself imply anything regarding the deterministic complexity of factoring, as has been mistakenly concluded by some authors.

Open Question 2. *Can we factor polynomials of degree n over $\mathrm{GF}(p^l)$ deterministically with $O(p^{1/2+\epsilon}(nl)^{2+\epsilon})$ operations in $\mathbf{Z}_p$.*

In factoring polynomials over $\mathrm{GF}(q)$, where $q = p^l$, we can easily adapt our method of constructing a separating set if we use as the Berlekamp subalgebra $B' = \{\alpha \in R : \alpha^q = \alpha\}$. This gives rise to a deterministic factoring algorithm that uses $O(qn^{2+\epsilon})$ operations in $\mathrm{GF}(q)$. However, to achieve the time bound in Open Question 2, we would like to use as the Berlekamp subalgebra $B = \{\alpha \in R : \alpha^p = \alpha\}$. Our method for constructing a separating set in this situation breaks down if any of the roots of the polynomial to be factored are conjugate over $\mathbf{Z}_p$, but not over $\mathrm{GF}(q)$.

References

A. Aho, J. Hopcroft and J. Ullman [1974]. *The Design and Analysis of Computer Algorithms*, Addison-Wesley.

E. Bach and V. Shoup [1988]. "Factoring polynomials using fewer random bits," Computer Sciences Technical Report No. 757, University of Wisconsin–Madison.

M. Ben-Or [1981]. "Probabilistic algorithms in finite fields," in *Proceedings of the 22nd Annual IEEE Symposium on Foundations of Computer Science*, pp. 394-398.

E. Berlekamp [1968]. *Algebraic Coding Theory,* McGraw-Hill.

E. Berlekamp [1970]. "Factoring polynomials over large finite fields," *Mathematics of Computation,* Vol. 24, No. 111, pp. 713-735.

A. Borodin and I. Munro [1975]. *The Computational Complexity of Algebraic and Numeric Problems,* American Elsevier.

D. Burgess [1963]. "On Dirichlet characters of polynomials," *Proceedings of the London Mathematics Society,* Vol. 3, No. 13, pp. 537-548.

P. Camion [1983a]. "Improving an algorithm for factoring polynomials over a finite field and constructing large irreducible polynomials," *IEEE Transactions on Information Theory,* Vol. 29, No. 3, pp. 378-385.

P. Camion [1983b]. "A deterministic algorithm for factorizing polynomials of $\mathbf{F}_q[X]$," *Annals of Discrete Mathematics,* Vol. 17, pp. 149-157.

D. Cantor and E. Kaltofen [1987]. "Fast multiplication of polynomials over arbitrary rings," Department of Computer Science Technical Report No. 87-35, Rensselaer Polytechnic Institute.

D. Cantor and H. Zassenhaus [1981]. "A new algorithm for factoring polynomials over finite fields," *Mathematics of Computation,* Vol. 36, No. 154, pp. 587-592.

K. Ireland and M. Rosen [1982]. *A Classical Introduction to Modern Number Theory,* Springer-Verlag.

D. Knuth [1981]. *The Art of Computer Programming, Vol. 2 (2nd edition),* Addison-Wesley.

D. Lazard [1982]. "On polynomial factorization," in *Computer Algebra,* J. Calmet, ed., Lecture Notes in Computer Science No. 144, Springer-Verlag, pp. 126-134.

R. McEliece [1969]. "Factorization of polynomials over finite fields," *Mathematics of Computation,* Vol. 23, pp. 861-867.

M. Rabin [1980]. "Probabilistic algorithms in finite fields," *SIAM Journal on Computing,* Vol. 9, No. 2, pp. 273-280.

L. Ronyai [1987]. "Factoring polynomials over finite fields," in *Proceedings of the 28th Annual IEEE Symposium on Foundations of Computer Science,* pp. 132-137.

W. Schmidt [1976]. *Equations over Finite Fields,* Springer-Verlag Lecture Notes in Mathematics No. 536.

A. Schönhage [1977]. "Schnelle Multiplikation von Polynomen über Körpern der Charakteristik 2," *Acta Informatica,* Vol. 7, pp. 395-398.