

Factoring polynomials over p -adic fields

David G. Cantor
and
Daniel M. Gordon

Center for Communications Research

July 4, 2000

Introduction

Factoring polynomials over \mathbb{Q}_p is an important problem in number theory.

For example, it is used to factor rational primes p over number fields.

Until recently, algorithms used in practice took exponential time in worst case.

Chistov gave a polynomial-time algorithm, which would be difficult to implement.

Recent Developments

In the past few years, three groups have independently come up with new, practical algorithms.

Bernd Souvignier and C. R. Leedham-Green wrote a new algorithm for MAGMA.

PARI now uses an improved version of the Round 4 algorithm due to Ford, Pauli and Roblot.

Pauli showed it has expected time

$$O\left(n^3 \log n(n \log n + \text{ord } \Delta_F)\right)$$

field operations.

Recent Developments (cont'd)

Our algorithm runs in random time

$$O\left(n^{4+\epsilon} \log^{2+\epsilon} |\Delta_F| \log^{1+\epsilon} p^k\right)$$

bit operations.

An extension of the algorithm works over other local fields.

Over the field $\mathbf{F}_q((X))$ of Laurent series, it can be used to resolve singularities of plane curves.

Basic idea

Try to factor $F(X)$ using “easy” methods (Newton diagram, factor modulo p).

If that fails, find an $A(X)$ such that we can factor

$$R(Y) = \text{Res}_X(F(X), Y - A(X)).$$

This will lead to a factorization of $F(X)$.

All the new algorithms use this idea; the difference is in how $A(X)$ is found.

Some Lemmas

Lemma 1 *Suppose that $F(X)$ and $A(X)$ are polynomials in $K[X]$, with $F(X)$ monic of degree n . Put*

$$R(Y) = \text{Res}_X(F(X), Y - A(X)).$$

Then

- 1. $R(Y)$ is a monic polynomial of degree n*
- 2. $F(X)$ divides $R(A(X))$.*

Lemma 2 *Suppose $R(Y) = R_1(Y)R_2(Y)$ is a factorization of $R(Y)$. Then*

$$F(X) = F_1(X)F_2(X),$$

where

$$F_i(X) = \gcd(F(X), R_i(A(X))).$$

Lemma 3 *If $R(Y)$ is irreducible over K , then $F(X)$ is also irreducible over K .*

Lemma 4 *Suppose L is a finite algebraic extension of K , and that $G(X)$ is an irreducible, monic factor of $F(X)$ in $L[X]$.*

Put $H(X) = \text{Norm}_{L/K} G(X)$.

Then $\gcd(F(X), H(X))$ is an irreducible factor of $F(X)$ in $K[X]$.

Newton Diagrams

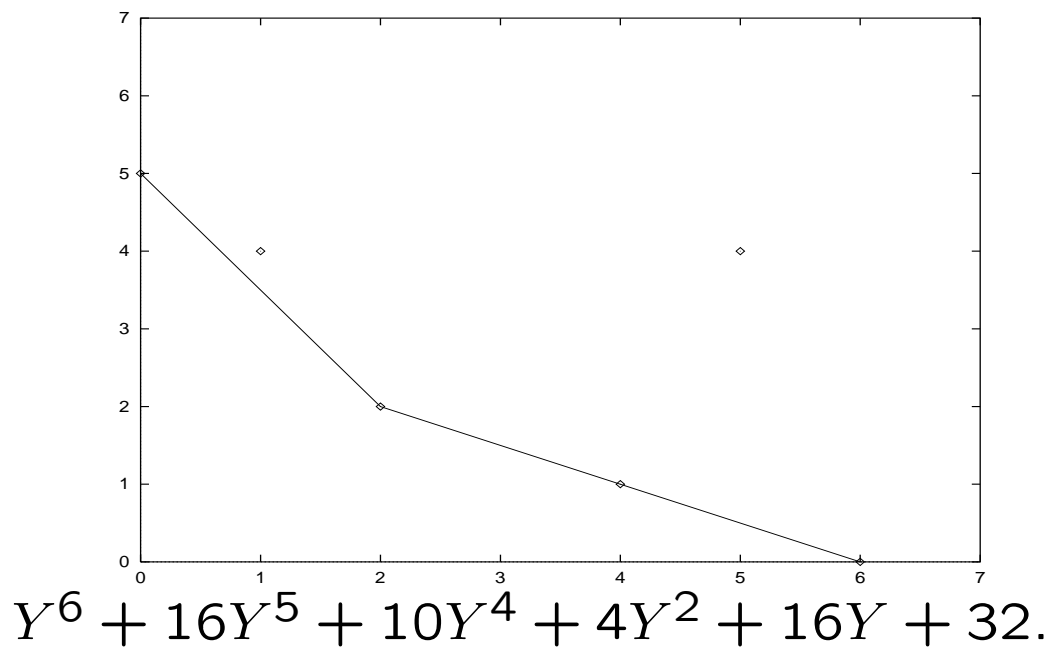
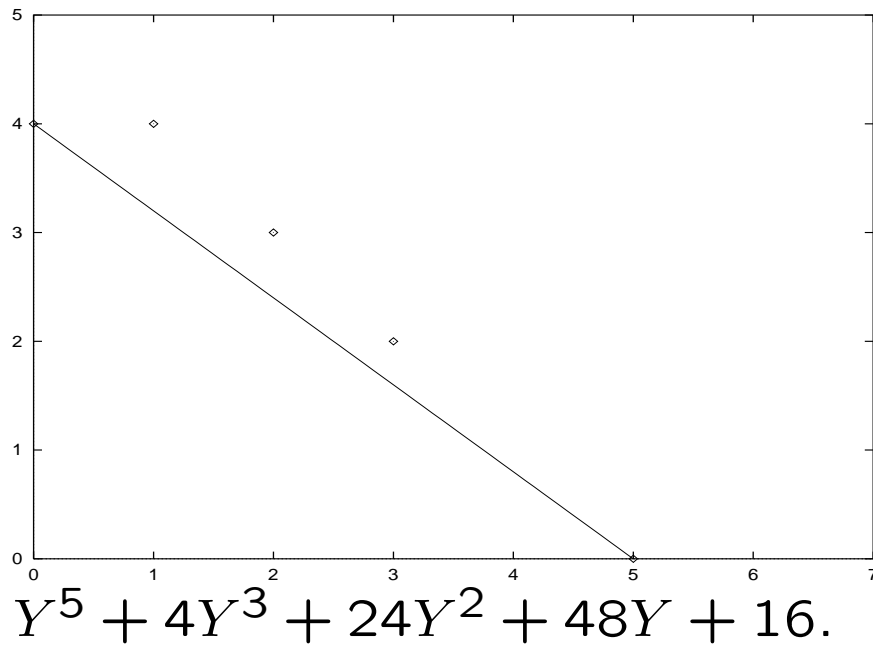
The Newton diagram of

$$R(Y) = \sum_{i=0}^n a_i Y^i$$

is the lower boundary of the convex hull of points $(i, \text{ord } a_i) \in \mathbb{R}^2$, for each nonzero term $a_i Y^i$ of $R(Y)$.

We call $R(Y)$ *pure* if $a_0 \neq 0$, $n \geq 1$, and the Newton diagram of $R(Y)$ is a straight line.

Newton Diagrams (cont'd)



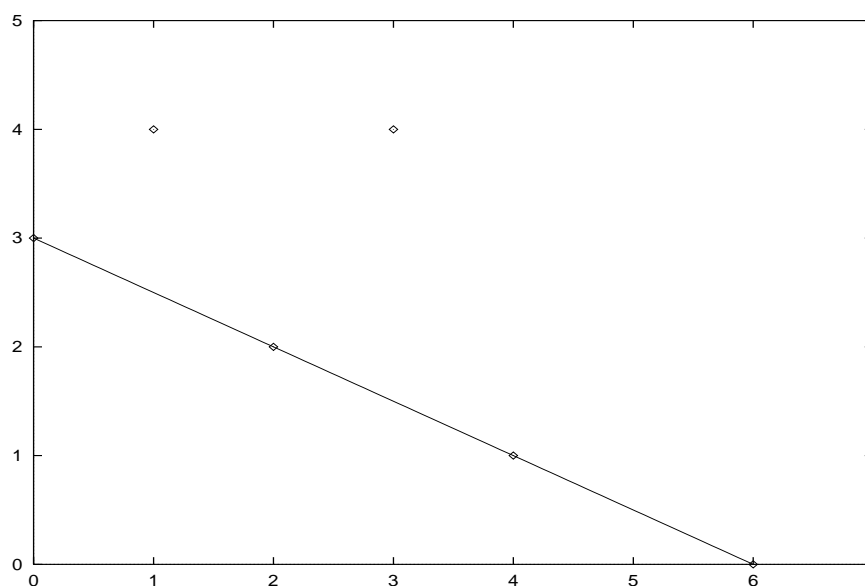
Newton Diagrams (cont'd)

Lemma 5 *If $R(Y)$ is not pure, then $R(Y)$ factors into two non-constant polynomials in $K[Y]$.*

Lemma 6 (Generalized Eisenstein criterion)
Suppose $R(Y)$ is pure, and its Newton diagram has slope k/n , where $\gcd(k, n) = 1$. Then $R(Y)$ is irreducible.

$$R^*(Y)$$

Send $Y \longrightarrow p^s Y^{1/r}$ to flatten, compress Newton diagram.



$$R(Y) = Y^6 + 2Y^4 + 16Y^3 + 4Y^2 + 48Y + 24.$$

$$R^*(Y) = Y^3 + Y^2 + Y + 1 \in \overline{K}.$$

Factors of $R^*(Y)$ over \overline{K} will give factors of $F(X)$.

Hensel Factor

Input $(K, F(X), A(X))$.

1. Compute $R(Y) = \text{Res}_X(F(X), Y - A(X))$.

2. **Hensel Factor** *succeeds* if

(a) The polynomial $R(Y)$ is not pure.

(b) The polynomial $R(Y)$ is pure and $R^*(Y)$ factors over $\overline{K}[X]$.

(c) The polynomial $R(Y)$ is pure and $R^*(Y)$ is the e th power of $\alpha(Y)$ of degree ≥ 2 .

(d) The polynomial $R(Y)$ is pure and the slope of its Newton diagram is k/n where $(k, n) = 1$.

The p-adic Factor Algorithm

Input: $(K, F(X))$.

Start with $A(X) = X$.

Apply **Hensel Factor** to $(K, F(X), A(X))$.

If it succeeds, great. Otherwise,

$$R(Y) = (Y^r - \alpha p^s)^m + \dots$$

We have $\text{ord } A(x) = s/r$ for each root x of $F(X)$.

The p-adic Factor Algorithm (cont'd)

Find a new $A(X)$ for which either:

1. **Hensel Factor** successfully factors $R(Y)$,
2. $\text{ord } A(x)$ is increased,
3. $\text{deg } A(X)$ is increased.

Since $\text{deg } A(X) < n$, and

$$\text{ord } A(x) \leq \text{deg } A(X) \cdot \text{ord } \Delta_F,$$

the algorithm will terminate in a bounded number of steps.

Example

Factor

$$F(X) = (X - 4)^2(X^2 - 2) + 2^{100}$$

over \mathbb{Q}_2 .

$F(X)$ is not pure, so we find factors

$$G_1 = (X^2 - 2) + (2^{101} + 2^{105} + \dots)X + (2^{99} + 2^{102} + \dots)$$

and

$$G_2 = (X - 4)^2 + (2^{101} + 2^{102} + \dots)X + (2^{99} + 2^{100} + \dots)$$

Hensel Factor immediately proves $G_1(X)$ is irreducible.

$A(X) = X - 4$ gives a proof that $G_2(X)$ is irreducible.

Example

Factor

$$F(X) = (X^2 - 2 - 2^{20})(X^2 - 2 + 2^{20})$$

over \mathbb{Q}_2 .

$F^*(X) = (X - 1)^2$, so **Hensel Factor** fails.

$A(X) = X^2/2$ gives

$$R(Y) = Y^4 - 4Y^3 + (6 - 2^{39})Y^2 + (-4 + 2^{40})Y + (1 - 2^{39} + 2^{76}).$$

and **Hensel Factor** fails.

$A(X) = X^2 - 2$ gives

$$R(Y) = Y^4 - 2^{41}Y^2 + 2^{80}.$$

and **Hensel Factor** fails again, with
ord $A(x) = 20$.

Example (cont'd)

Next we get $A(X) = X^2 - 2 - 2^{20}$, and

$$R(Y) = Y^4 + 2^{22}Y^3 + 2^{42}Y^2.$$

The factor of Y^2 gives

$$G_1(X) = X^2 - 2 - 2^{20}.$$

Complexity Problems

We need to guarantee that we can use finite precision and get the right answer.

In the computation

$$F_i(X) = \gcd(F(X), R_i(A(X))),$$

we do not know the R_i exactly, and so terms in the computation that appear to be zero may not be.

It is difficult to estimate the accuracy of $R_i(Y)$ that is needed, so we use another method, which involves solving linear equations.

Precision bound

We need to show that finite approximations of $R_i(Y)$ give correct factorizations.

Theorem 1 $O(|\Delta_f|)$ p -adic digits of precision suffice.

If $R(Y)$ is irreducible and

$$\|R_0(Y) - R(Y)\| < \min(1, |\Delta_{R_0}|),$$

then $R_0(Y)$ is irreducible.

If $|R(Y) - B_0(Y)C_0(Y)| < \min(1, |\Delta_{R_0}|)$, then we may find a factorization using Hensel's Lemma.

Main Result

Theorem 2 *Let K be an extension of degree k of \mathbb{Q}_p , and $F(X) \in K[X]$ have degree n . Algorithm **p-adic Factor** will factor $F(X)$ in random time*

$$O\left(n^{4+\epsilon} \log^{2+\epsilon} |\Delta_F| \log^{1+\epsilon} p^k\right)$$

bit operations.

The algorithm is nondeterministic only because of the mod p factoring step.